Министерство науки и высшего образования РФ
УрФУ имени первого Президента России Б.Н. Ельцина

Ю. Б. Мельников

Алгоритм

Раздел **электронного учебника**
для сопровождения лекции

e-mail:
UriiMelnikov58@gmail.com

Екатеринбург
2020

I. Инструкция к пособию **5**

II. Понятие алгоритма **14**

III. Нормальный алгорифм Маркова **38**

III.1. Нормальный алгорифм: понятия 42

III.2. Нормальный алгорифм: работа 60

IV. Машина Тьюринга **89**

Пример 1 нормального алгорифма Маркова, «гасящего» незначащие нули **137**

Пример 2 создания нормального алгорифма Маркова, вставляющего 0 между сдвоенными единицами **235**

Пример 3 нормального алгорифма Маркова, находящего середину слова 304

Задача IV.1 410

Задача IV.2 411

Пример 4 машины Тьюринга, «гасящей» незначащие нули 412

Пример 5 машины Тьюринга, вставляющей 0 между сдвоенными единицами 442

Пример 6 машины Тьюринга, находящей середину слова 610

Задача IV.3 715

Задача IV.4 716

Задача IV.5 717

Ответы и решения 718

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для корректной работы тестов следует применять компьютеры с процессором архитектуры с `Intel x-86`.

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для корректной работы тестов следует применять компьютеры с процессором архитектуры с `Intel x-86`.

Электронный учебник представляет собой систему из основного файла `0000Spisok.pdf` со ссылками на файлы `00Set.pdf`, `00Matrix.pdf`, `00AnalGeom.pdf`, `00LinAlgebra.pdf`, и файлы с тестами для обучения и самоконтроля, которые следует просматривать с помощью программы `Adobe Reader`.

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для корректной работы тестов следует применять компьютеры с процессором архитектуры с `Intel x-86`.

Электронный учебник представляет собой систему из основного файла `0000Spisok.pdf` со ссылками на файлы `00Set.pdf`, `00Matrix.pdf`, `00AnalGeom.pdf`, `00LinAlgebra.pdf`, и файлы с тестами для обучения и самоконтроля, которые следует просматривать с помощью программы `Adobe Reader`.

Кроме того, имеются гиперссылки на пособия **«Математический анализ»** и **«Элементарная математика».**

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

В презентациях, предназначенных для проведения практических занятий, имеется два вида учебных заданий: примеры, предназначенные для иллюстрации теоретического материала, демонстрации методов решения задач и т. п., и задачи, предназначенные для самостоятельного решения. Имеются гиперссылки на тесты для самообучения и самоконтроля.

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

В программе `Adobe Reader` и `Acrobat Reader` переход в полноэкранный режим и возвращение к режиму работы в окне осуществляется комбинацией клавиш Ctrl+L (т.е. одновременным нажатием клавиш «Ctrl» и «L»). Переход к следующему слайду или возвращение к предыдущему слайду осуществляется клавишами «Page Up» или «Page Down».

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для того, чтобы ***вызвать панель навигации*** в `Acrobat Reader` надо, во-первых, выйти из полноэкранного режима (например, нажатием `Esc`), и, во-вторых, нажать клавишу `F4` и выбрать на левой вертикальной панели вертикальный флажок «Закладки» 🔖.

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для перехода по гиперссылке, как обычно, следует навести указатель мыши на текст, выделенный красным (но не пурпурным) или синим цветом и нажать на левую кнопку мыши или левую кнопку тачпада (для ноутбука). «Откат», т.е. отмена предыдущей команды (например, перехода по гиперссылке) осуществляется одновременным нажатием клавиш Alt и $\leftarrow$ (в `Adobe Reader X` может не работать).

I. Инструкция к пособию

Для просмотра файлов pdf настоятельно рекомендуем использовать программу `Adobe Reader 11`, `Acrobat Reader DC` или более поздней версии. В крайнем случае можно использовать `Adobe Reader` версии 8 или 9 (но не 10).

Для перехода по гиперссылке, как обычно, следует навести указатель мыши на текст, выделенный красным (но не пурпурным) или синим цветом и нажать на левую кнопку мыши или левую кнопку тачпада (для ноутбука). «Откат», т.е. отмена предыдущей команды (например, перехода по гиперссылке) осуществляется одновременным нажатием клавиш Alt и $\leftarrow$ (в `Adobe Reader X` может не работать).

В случае, если два соседних слова выделены, допустим, синим цветом, но одно набрано обычным, а другое — полужирным шрифтом, то это означает, что переход по гиперссылкам осуществляется на различные мишени.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

Необходимость в формализации этого понятия возникает, в основном, в ситуации, когда мы хотим убедиться в том, что алгоритма не существует

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

Необходимость в формализации этого понятия возникает, в основном, в ситуации, когда мы хотим убедиться в том, что алгоритма не существует

История математики знает примеры безуспешности многовековых усилий ученых по созданию алгоритмов.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

Необходимость в формализации этого понятия возникает, в основном, в ситуации, когда мы хотим убедиться в том, что алгоритма не существует

История математики знает примеры безуспешности многовековых усилий ученых по созданию алгоритмов.

Примерами являются задача о квадратуре круга, проблема пятого постулата Евклида, отыскание корней многочлена степени, большей 4.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

Необходимость в формализации этого понятия возникает, в основном, в ситуации, когда мы хотим убедиться в том, что алгоритма не существует или когда возникают сомнения в том, алгоритм ли это.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Зачем знать точное определение алгоритма?

Если мы нашли алгоритм решения определенной задачи, то нам не понадобится точное значение того, что есть алгоритм.

Необходимость в формализации этого понятия возникает, в основном, в ситуации, когда мы хотим убедиться в том, что алгоритма не существует или когда возникают сомнения в том, алгоритм ли это.

Например, формализация понятия «граница города» требуется в условиях конфликта интересов.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

Например, исправление орфографических ошибок в тексте можно задать функцией, аргумент которой —

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

Например, исправление орфографических ошибок в тексте можно задать функцией, аргумент которой — исходный текст, а значение —

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

Например, исправление орфографических ошибок в тексте можно задать функцией, аргумент которой — исходный текст, а значение — текст, все орфографические ошибки в котором исправлены.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

В процессе построения различных алгоритмов были получены различные формализации понятия алгоритма.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

В процессе построения различных алгоритмов были получены различные формализации понятия алгоритма.

Наличие разных формализаций связано с тем, что для разных классов таких функций удобны разные подходы к понятию алгоритма.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

В процессе построения различных алгоритмов были получены различные формализации понятия алгоритма.

Главный вопрос — существует ли алгоритм, «реализующий» данную функцию, то есть вычисляющий ее значение на любом аргументе из ее области определения.

II. Понятие алгоритма

Существует несколько формализаций понятия алгоритма.

Можно считать, что всякий математический алгоритм вычисляет ту или иную функцию.

В процессе построения различных алгоритмов были получены различные формализации понятия алгоритма.

Главный вопрос — существует ли алгоритм, «реализующий» данную функцию, то есть вычисляющий ее значение на любом аргументе из ее области определения.

Например, пользуясь только циркулем и линейкой, невозможно разделить произвольный угол на три равные части (задача о трисекции угла), а на две части — можно.

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина».

(машина Тьюринга, машина Поста, машина Минского и т.п.).

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

Существуют действующие модели таких «машин», как в виде программ, имитирующих их работу, так даже и в виде электромеханических конструкций.

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

Существуют действующие модели таких «машин», как в виде программ, имитирующих их работу, так даже и в виде электромеханических конструкций.

Последние следует рассматривать, скорее, как игрушки или «рекламный трюк»,

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

Существуют действующие модели таких «машин», как в виде программ, имитирующих их работу, так даже и в виде электромеханических конструкций.

Модели в виде программ можно использовать для экспериментальной проверки правильности «функционирования» алгоритмов, с помощью которых решаются некоторые научные проблемы.

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

На самом деле вместо таких «машин» можно рассматривать математические «конструкции» традиционного типа: функции, множества и т.п. (см. Ю.Л. Ершов, Е.А. Палютин «Математическая логика»).

II. Понятие алгоритма

Оказалось, что в большинстве случаев разные формализации понятия алгоритма приводят к одному и тому же итоговому результату в том смысле, что существование алгоритма при одном понимании этого термина гарантирует его существование в другом понимании.

В формализациях понятия алгоритма часто встречается понятие «машина». При этом под «машиной» в этом случае следует понимается некое «абстрактное устройство».

На самом деле вместо таких «машин» можно рассматривать математические «конструкции» традиционного типа: функции, множества и т.п. (см. Ю.Л. Ершов, Е.А. Палютин «Математическая логика»).

Мы рассмотрим две таких формализации: **нормальный алгорифм Маркова** и так называемую **«машину Тьюринга»**.

III. Нормальный алгорифм Маркова

Алгорифм — синоним слова алгоритм.

III. Нормальный алгорифм Маркова

Алгорифм — синоним слова алгоритм.

Слово «алгоритм» представляет собой искажение имени среднеазиатского математика аль-Хорезми (9 век), латинизированное — Algorithmi.

III. Нормальный алгорифм Маркова

Алгорифм — синоним слова алгоритм.

Слово «алгоритм» представляет собой искажение имени среднеазиатского математика аль-Хорезми (9 век), латинизированное — Algorithmi.

При прочтении многих латинских слов звуки «т» и «ф» взаимозаменяемы (сравните звучание имени: Фома и Томас).

III. Нормальный алгорифм Маркова

Алгорифм — синоним слова алгоритм.

Слово «алгоритм» представляет собой искажение имени среднеазиатского математика аль-Хорезми (9 век), латинизированное — Algorithmi.

При прочтении многих латинских слов звуки «т» и «ф» взаимозаменяемы (сравните звучание имени: Фома и Томас).

Термин «алгорифм» традиционно используется при изложении теории нормальных алгоритмов.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Под длиной цепочки мы понимаем количество букв в слове.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

В этой теории пустословие носит терпимую форму:

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

В этой теории пустословие носит терпимую форму: пустых слов можно наговорить много, но они все уж очень короткие.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

В этой теории пустословие носит терпимую форму: пустых слов можно наговорить много, но они все уж очень короткие.

И все одинаковые.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

В этой теории пустословие носит терпимую форму: пустых слов можно наговорить много, но они все уж очень короткие.

И все одинаковые.

И сразу видно, что они — пустые.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Иногда пишут $a + b$.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Иногда пишут $a + b$.

Например, конкатенацией слов 103 и 2654 является слово

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Иногда пишут $a + b$.

Например, конкатенацией слов 103 и 2654 является слово 1032654.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Иногда пишут $a + b$.

Например, конкатенацией слов 103 и 2654 является слово 1032654.

В данном случае символ «+» неуместен, поскольку эти слова воспринимаются, как числа, а «+» — как операция суммирования, а не конкатенации.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Слово a называется **подсловом** слова b, если

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Слово a называется **подсловом** слова b, если b есть конкатенация слов c, a, d, т.е.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Слово a называется **подсловом** слова b, если b есть конкатенация слов c, a, d, т.е. $b = cad$.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Слово a называется **подсловом** слова b, если b есть конкатенация слов c, a, d, т.е. $b = cad$.

Здесь, как обычно, c и d могут быть пустыми словами.

III.1. Нормальный алгорифм: понятия

Алфавит — произвольное множество символов.

Слово в алфавите **A** — произвольная конечная цепочка символов алфавита **A**, длина цепочки называется **длиной слова**.

Слово может быть **пустым**, длина пустого слова равна 0.

Конкатенацией слов a и b называется слово ab.

Слово a называется **подсловом** слова b, если b есть конкатенация слов c, a, d, т.е. $b = cad$.

Нормальный алгорифм Маркова — это *упорядоченная* цепочка команд $\mathbf{Alg} = (A_1, A_2, \ldots, A_n)$, причем каждая команда A_i имеет вид (α, β, γ), где α, β слова в алфавите **A**, γ — либо 0, либо 1.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big)\,, \quad \gamma_i \in \{0;\ 1\}.$

Нормальный алгорифм перерабатывает входное слово в алфавите $\mathbf{A}$, либо получая в результате выполнения *конечного* числа шагов выходное слово в сочетании с «сигналом» нормального окончания работы алгорифма, либо останавливаясь без подачи сигнала о нормальной остановке (аварийная остановка), либо зацикливаясь — бесконечно «работая».

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово ……

Каждый шаг работы алгорифма начинается с *первой команды*.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово ……

Каждый шаг работы алгорифма начинается с *первой команды*.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово ……

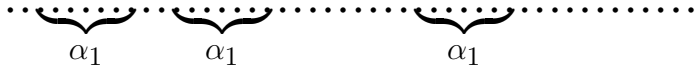
Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово

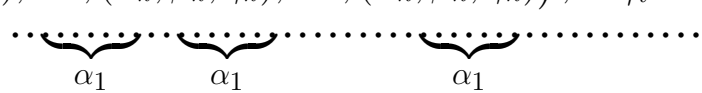


Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

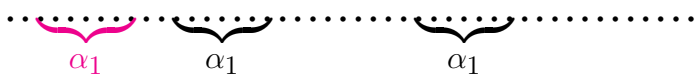
Входное слово 

Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то во входном слове сначала ищем *первое слева* вхождение подслова α_1,

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



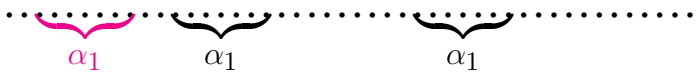
Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то во входном слове сначала ищем *первое слева* вхождение подслова α_1,

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово

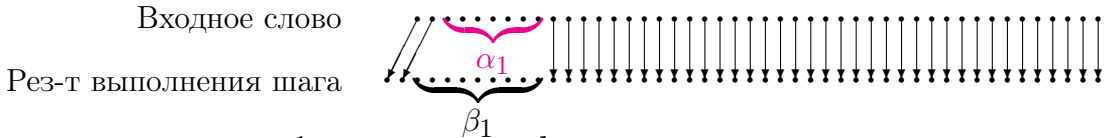


Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то во входном слове сначала ищем *первое слева* вхождение подслова α_1, и заменяем его на β_1.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

Если слово α_1 из первой команды $(\alpha_1, \beta_1, \gamma_1)$ является подсловом входного слова, то во входном слове сначала ищем *первое слева* вхождение подслова α_1, и заменяем его на β_1.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово ……

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

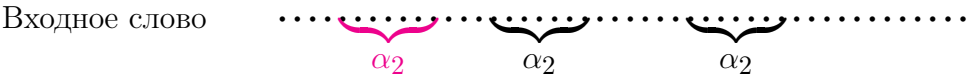
Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то во входном слове ищем *первое слева* вхождение

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



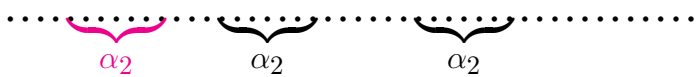
Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то во входном слове ищем *первое слева* вхождение

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



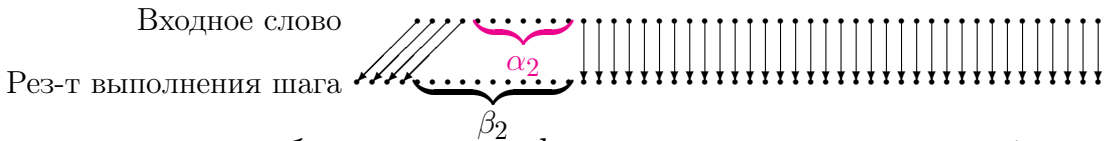
Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то во входном слове ищем *первое слева* вхождение подслова α_2, и заменяем его на β_2.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$



Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_1 не является подсловом входного слова, то переходим ко второй команде.

Если слово α_2 из второй команды $(\alpha_2, \beta_2, \gamma_2)$ является подсловом входного слова, то во входном слове ищем *первое слева* вхождение подслова α_2, и заменяем его на β_2.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

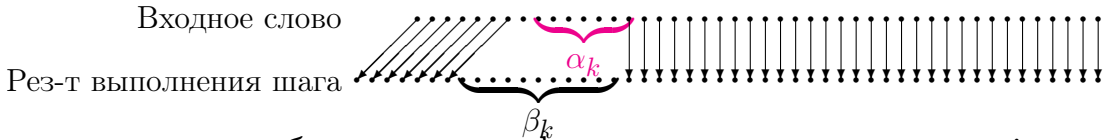
Входное слово ……

Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

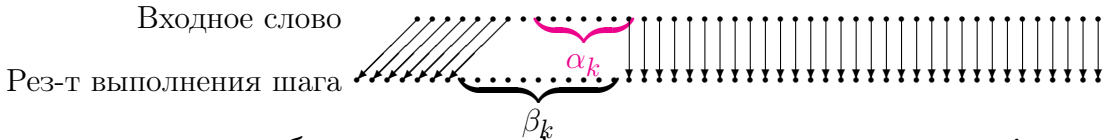


Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



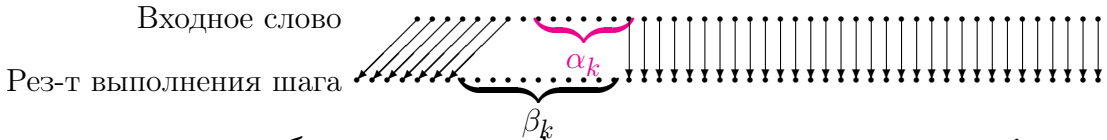
Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

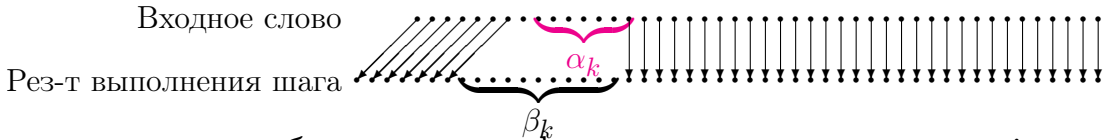
Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

Если $\gamma_k = 0$, начинается новый шаг алгорифма,

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

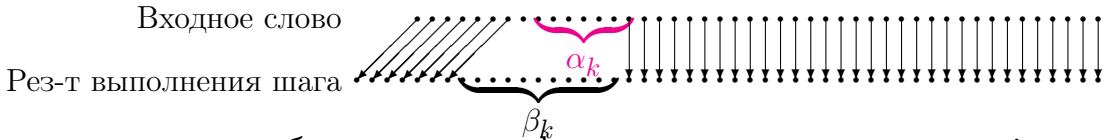
Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

Если $\gamma_k = 0$, начинается новый шаг алгорифма, начинающийся

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

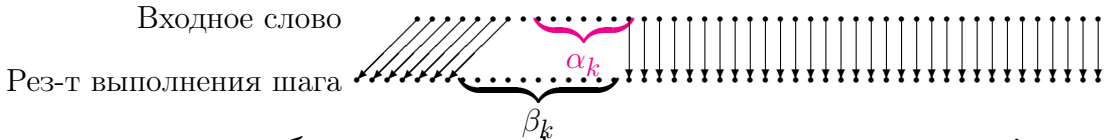
Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

Если $\gamma_k = 0$, начинается новый шаг алгорифма, начинающийся с попытки выполнить **первую команду**

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

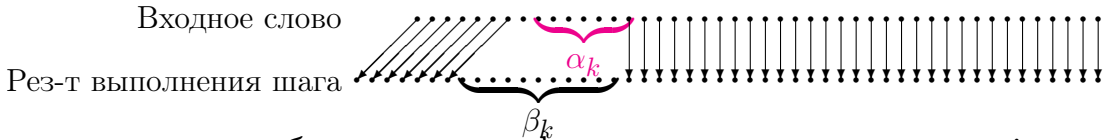
Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

Если $\gamma_k = 0$, начинается новый шаг алгорифма, начинающийся с попытки выполнить **первую команду**

III.2. Нормальный алгорифм: работа

$$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$



Каждый шаг работы алгорифма начинается с *первой команды*.

Если же α_2 не является подсловом входного слова, то переходим к третьей команде и т.п.

Если $\gamma_k = 0$, то работа алгорифма продолжается (*с первой команды!*). Если же $\gamma_k = 1$, то происходит **нормальная остановка** работы алгорифма, и получившееся слово является **выходным словом**.

Если $\gamma_k = 0$, начинается новый шаг алгорифма, начинающийся с попытки выполнить **первую команду** алгорифма.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если «сигнал» $\gamma_i = 1$ нормальной остановки не получен, но к «обрабатываемому» слову нельзя больше применить ни одну команду алгорифма (то есть ни одно из слов $\alpha_1, \alpha_2, \ldots, \alpha_n$ не является подсловом входного слова), то говорят, что входное слово **не поддается** алгорифму **Alg**.

III.2. Нормальный алгорифм: работа

$\mathbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если «сигнал» $\gamma_i = 1$ нормальной остановки не получен, но к «обрабатываемому» слову нельзя больше применить ни одну команду алгорифма (то есть ни одно из слов $\alpha_1, \alpha_2, \ldots, \alpha_n$ не является подсловом входного слова), то говорят, что входное слово **не поддается** алгорифму **Alg**.

Слово не поддается алгорифму и в случае, если «сигнал» γ_i не будет получен за конечное количество шагов, то есть в ситуации «зацикливания», бесконечного повторения алгорифма.

III.2. Нормальный алгорифм: работа

$$\textbf{Alg} = \big((\alpha_1, \beta_1, \gamma_1), (\alpha_2, \beta_2, \gamma_2), \ldots, (\alpha_k, \beta_k, \gamma_k), \ldots, (\alpha_n, \beta_n, \gamma_n)\big), \quad \gamma_i \in \{0;\ 1\}.$$

Входное слово .....

Каждый шаг работы алгорифма начинается с *первой команды*.

Если «сигнал» $\gamma_i = 1$ нормальной остановки не получен, но к «обрабатываемому» слову нельзя больше применить ни одну команду алгорифма (то есть ни одно из слов $\alpha_1, \alpha_2, \ldots, \alpha_n$ не является подсловом входного слова), то говорят, что входное слово **не поддается** алгорифму **Alg**.

Слово не поддается алгорифму и в случае, если «сигнал» γ_i не будет получен за конечное количество шагов, то есть в ситуации «зацикливания», бесконечного повторения алгорифма.

Рассмотрим пример?

IV. Машина Тьюринга

В этом разделе мы опишем еще одну формализацию алгоритма.

IV. Машина Тьюринга

В этом разделе мы опишем еще одну формализацию алгоритма.

Мы приведем сравнительно простую модель, называемую «машиной Тьюринга».

IV. Машина Тьюринга

В этом разделе мы опишем еще одну формализацию алгоритма.

Мы приведем сравнительно простую модель, называемую «машиной Тьюринга».

Под машиной Тьюринга в этой модели понимается и некое «программируемое устройство» (которое мы опишем ниже), и «программа» для этого устройства.

IV. Машина Тьюринга

В этом разделе мы опишем еще одну формализацию алгоритма.

Мы приведем сравнительно простую модель, называемую «машиной Тьюринга».

Под машиной Тьюринга в этой модели понимается и некое «программируемое устройство» (которое мы опишем ниже), и «программа» для этого устройства.

В неформальном смысле мы будем использовать термин «машина Тьюринга» в основном в смысле «программа для устройства».

IV. Машина Тьюринга

Представьте себе устройство, способное находиться в одном из состояний из некоторого списка возможных состояний (называемых **внутренними состояниями машины Тьюринга**), и снабженное одной головкой «чтения-записи».

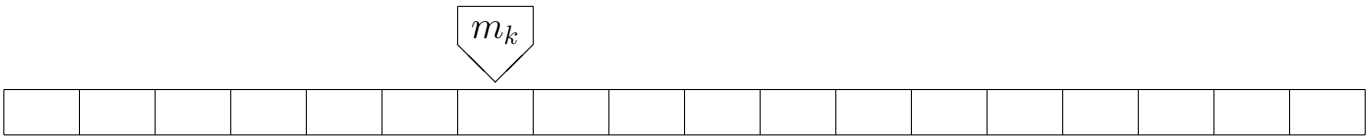
IV. Машина Тьюринга

Представьте себе устройство, способное находиться в одном из состояний из некоторого списка возможных состояний (называемых **внутренними состояниями машины Тьюринга**), и снабженное одной головкой «чтения-записи».

IV. Машина Тьюринга

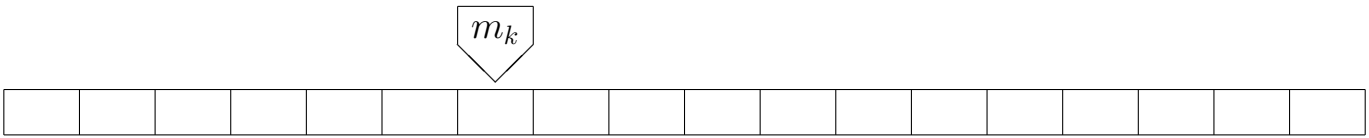
Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

IV. Машина Тьюринга



Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

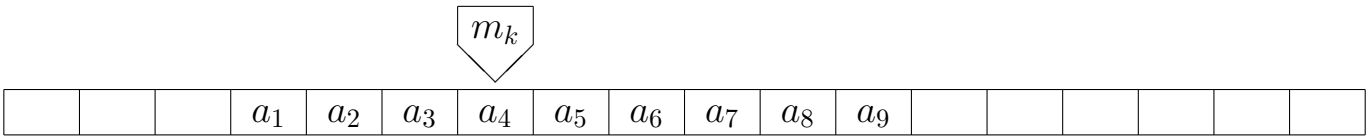
IV. Машина Тьюринга



Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

Предполагается, что каждая ячейка ленты содержит только одну «букву».

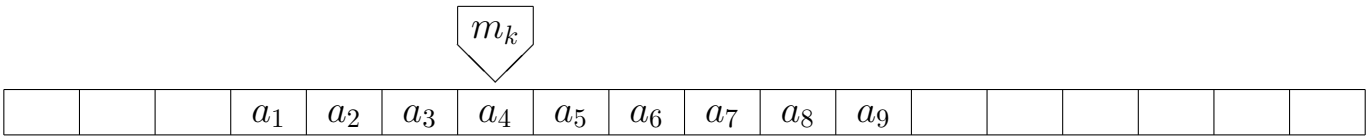
IV. Машина Тьюринга



Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

Предполагается, что каждая ячейка ленты содержит только одну «букву».

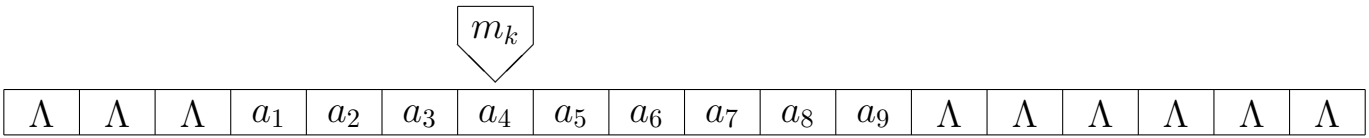
IV. Машина Тьюринга



Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

Предполагается, что каждая ячейка ленты содержит только одну «букву». Содержимое пустой ячейки обозначим символом Λ.

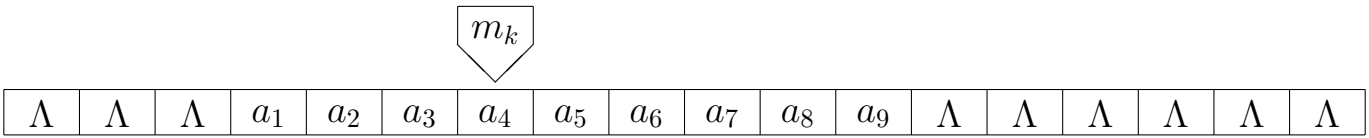
IV. Машина Тьюринга



Входное слово $a_{i_1}a_{i_2}a_{i_3}\ldots$ записано на бесконечной в обе стороны **ленте**, разделенной на **ячейки**.

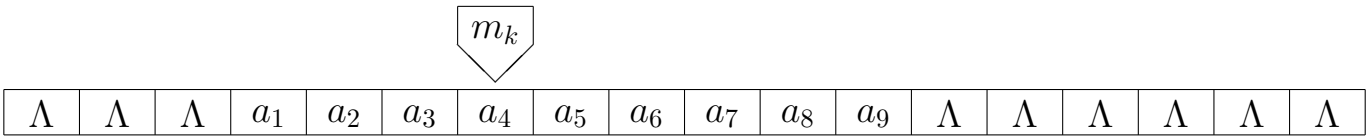
Предполагается, что каждая ячейка ленты содержит только одну «букву». Содержимое пустой ячейки обозначим символом Λ.

IV. Машина Тьюринга



Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

IV. Машина Тьюринга

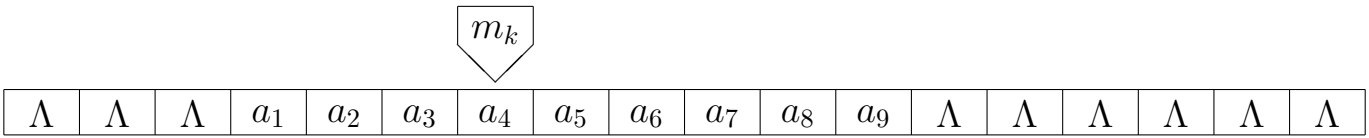


Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, \qquad \ldots\}$

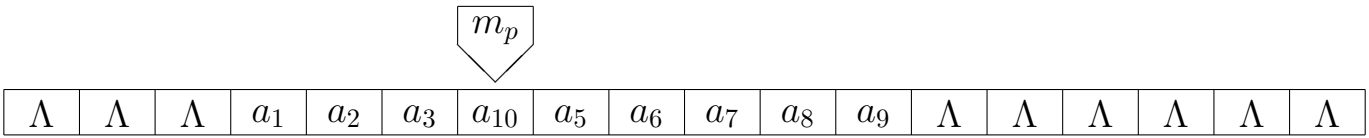


Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, \qquad \ldots\}$

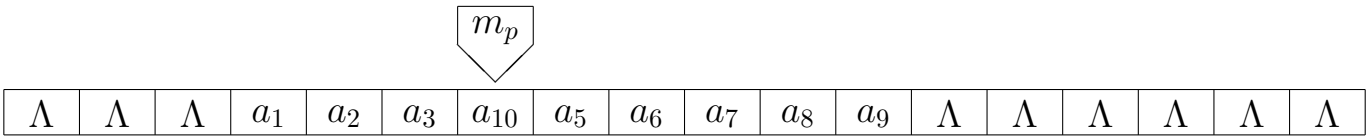


Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, \qquad \ldots\}$



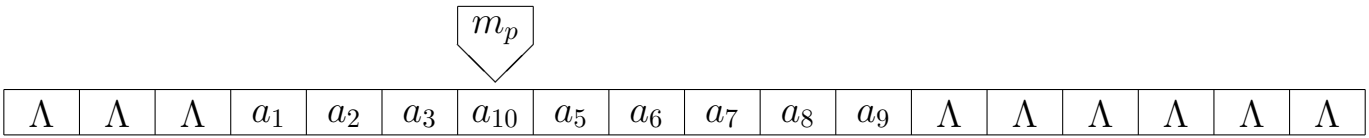
Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

После перезаписи содержимого ячейки головка способна переместиться к соседней слева или справа ячейке.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, \qquad (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



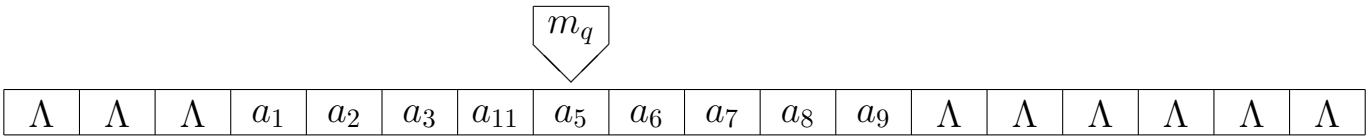
Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

После перезаписи содержимого ячейки головка способна переместиться к соседней слева или справа ячейке.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, \qquad (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



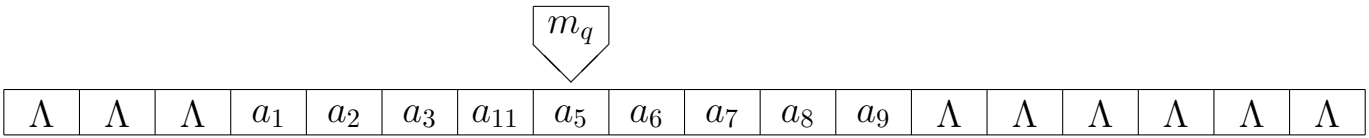
Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

После перезаписи содержимого ячейки головка способна переместиться к соседней слева или справа ячейке.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



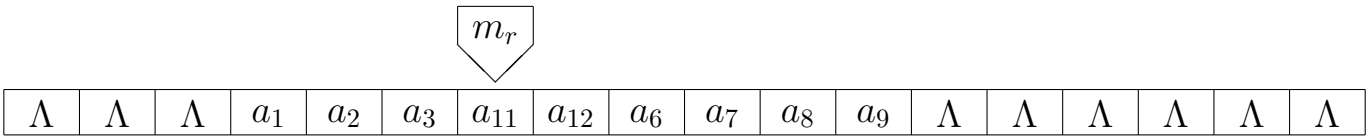
Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

После перезаписи содержимого ячейки головка способна переместиться к соседней слева или справа ячейке.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



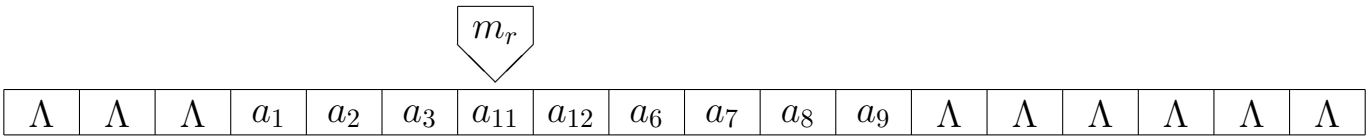
Головка устройства способна считывать содержимое ячейки и изменять, перезаписывать содержимое этой ячейки в зависимости от того, какой символ был считан в ячейке, и в каком внутреннем состоянии находилось устройство (из списка внутренних состояний машины).

При этом устройство переходит в новое внутреннее состояние (из того же списка внутренних состояний), впрочем, «новое» состояние может совпадать со «старым».

После перезаписи содержимого ячейки головка способна переместиться к соседней слева или справа ячейке.

IV. Машина Тьюринга

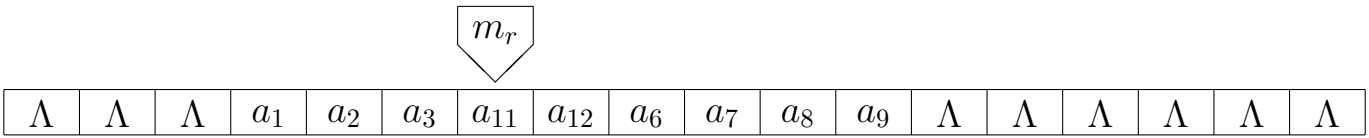
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



У машины Тьюринга есть два особых внутренних состояния: «начальное состояние» m_0 и «заключительное состояние» m_1.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

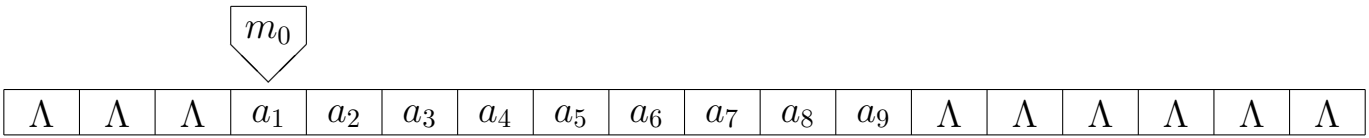


У машины Тьюринга есть два особых внутренних состояния: «начальное состояние» m_0 и «заключительное состояние» m_1.

В начальный момент времени головка машины Тьюринга находится над ячейкой с крайним левым символом входного слова в состоянии m_0.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

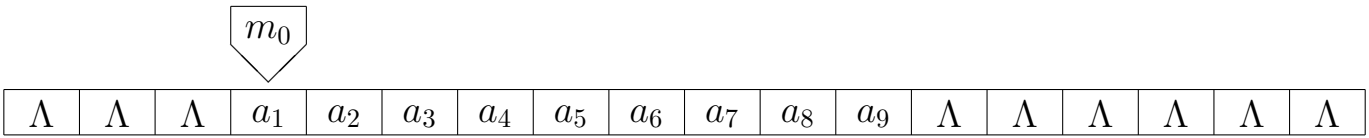


У машины Тьюринга есть два особых внутренних состояния: «начальное состояние» m_0 и «заключительное состояние» m_1.

В начальный момент времени головка машины Тьюринга находится над ячейкой с крайним левым символом входного слова в состоянии m_0.

IV. Машина Тьюринга

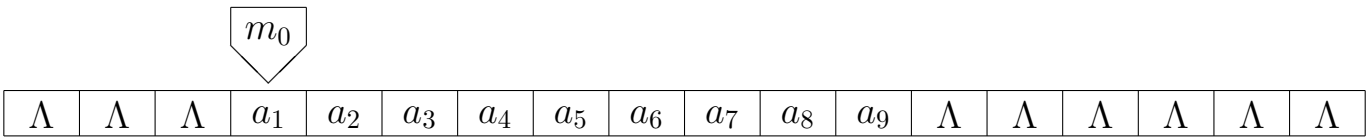
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то

IV. Машина Тьюринга

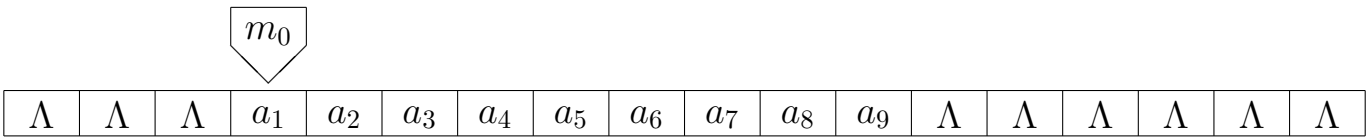
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то

IV. Машина Тьюринга

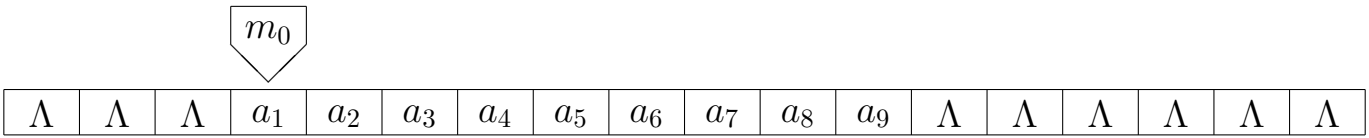
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево,

IV. Машина Тьюринга

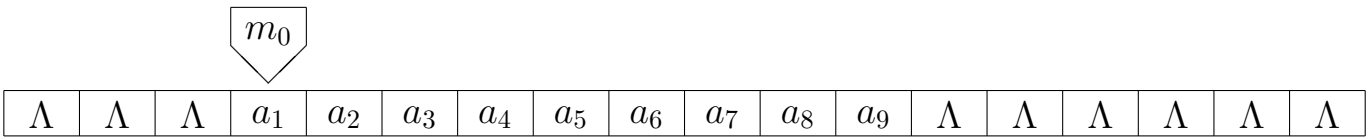
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то

IV. Машина Тьюринга

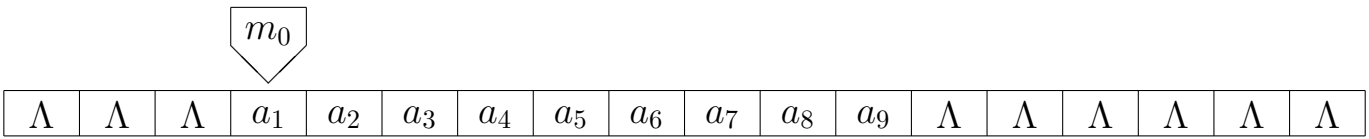
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то

IV. Машина Тьюринга

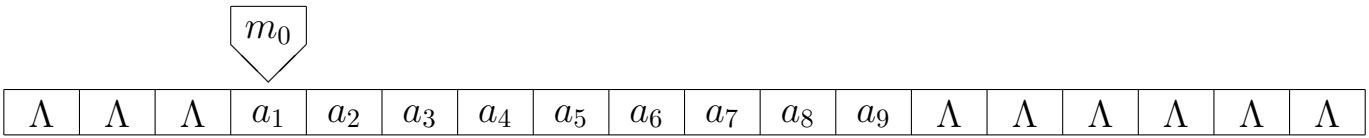
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо,

IV. Машина Тьюринга

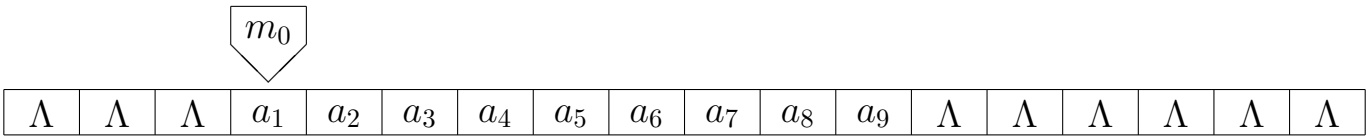
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо, если V отсутствует, то

IV. Машина Тьюринга

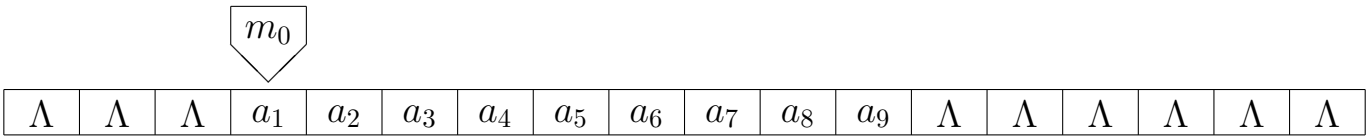
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо, если V отсутствует, то

IV. Машина Тьюринга

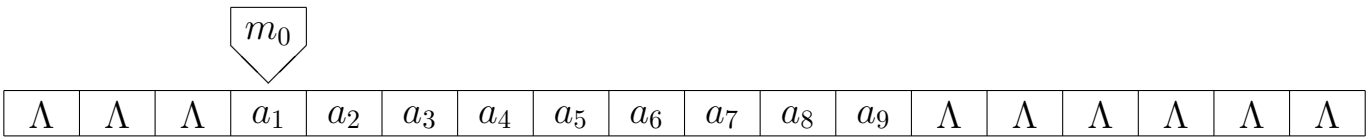
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо, если V отсутствует, то головка должна остаться над той же ячейкой.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

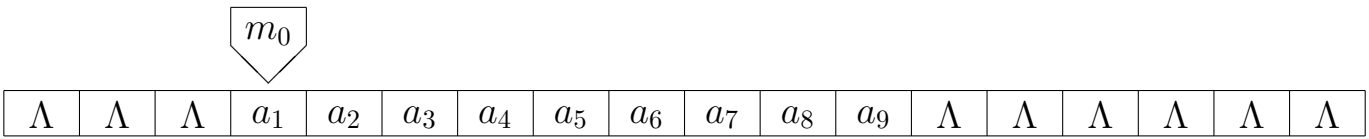


Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо, если V отсутствует, то головка должна остаться над той же ячейкой.

У каждой команды сочетание (a_u, m_p)

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

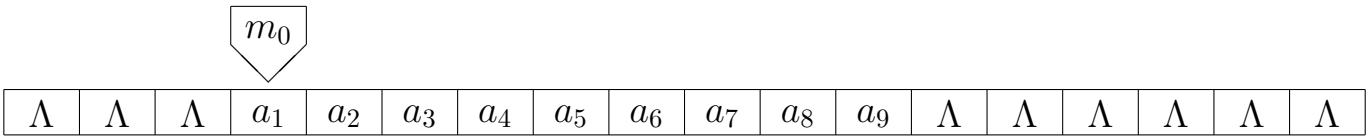


Каждая команда машины Тьюринга имеет вид $(a_u, m_p) V (a_v, m_q)$, где a_u — содержимое ячейки, обозреваемой головкой, находящейся в состоянии m_p, a_v — новое содержимое ячейки, m_q — новое состояние головки, если $V = L$, то головка должна сместиться на одну ячейку влево, если $V = R$, то головка должна сместиться на одну ячейку вправо, если V отсутствует, то головка должна остаться над той же ячейкой.

У каждой команды сочетание (a_u, m_p) является уникальным и в других командах не встречается.

IV. Машина Тьюринга

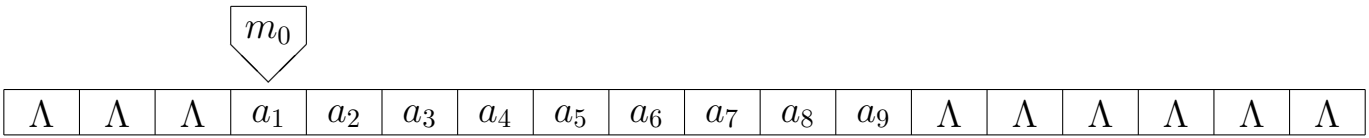
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Если головка переходит в состояние m_1, работа машины Тьюринга останавливается нормальным образом.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

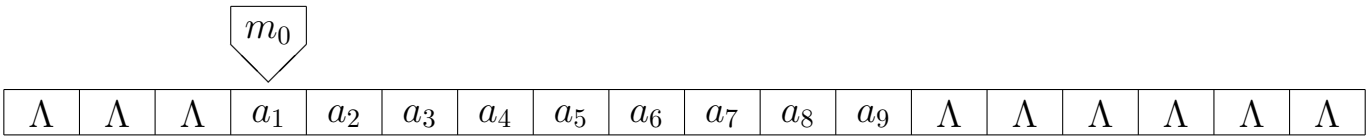


Если головка переходит в состояние m_1, работа машины Тьюринга останавливается нормальным образом.

Если сочетание содержимого ячейки a_w и состояния головки m_z не встречается ни в одной команде («зависание»), то то будем говорить, что работа машины завершена **аварийно**.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



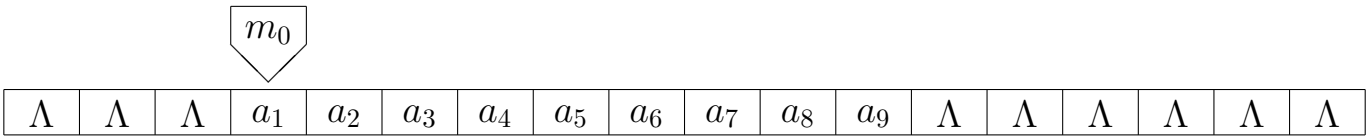
Если головка переходит в состояние m_1, работа машины Тьюринга останавливается нормальным образом.

Если сочетание содержимого ячейки a_w и состояния головки m_z не встречается ни в одной команде («зависание»), то то будем говорить, что работа машины завершена **аварийно**.

Если же в процессе работы алгоритма никогда не будет получено состояние головки m_1, то входное слово считается для машины Тьюринга **тупиковым**.

IV. Машина Тьюринга

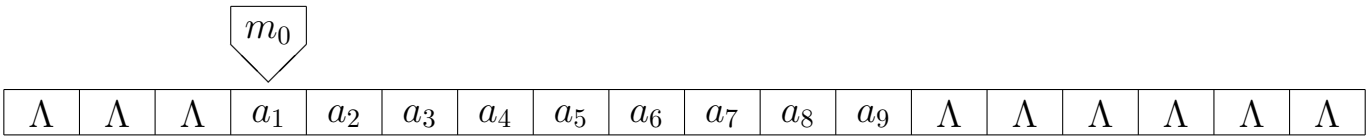
$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$



Одно из заметных отличий машины Тьюринга от нормального алгорифма Маркова состоит в *неупорядоченности* команд.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

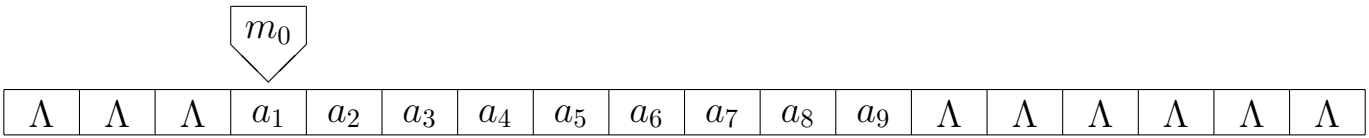


Одно из заметных отличий машины Тьюринга от нормального алгорифма Маркова состоит в *неупорядоченности* команд.

Машина Тьюринга не изменится, если в $\mathbf{T}$ поменять местами две команды.

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

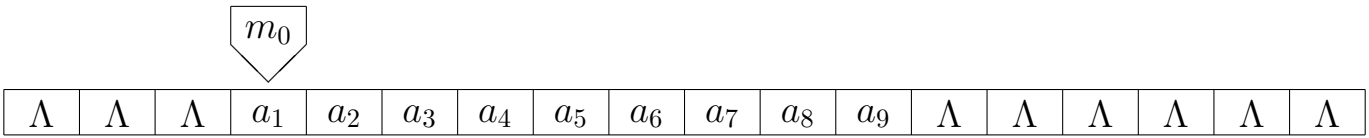


Иногда применяют табличное оформление машины Тьюринга:

	m_0	m_1	…	m_k	…	m_p	…	m_q	…	m_r	…
a_1											
…											
a_4											
a_5											
…											
a_{10}											
…											

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

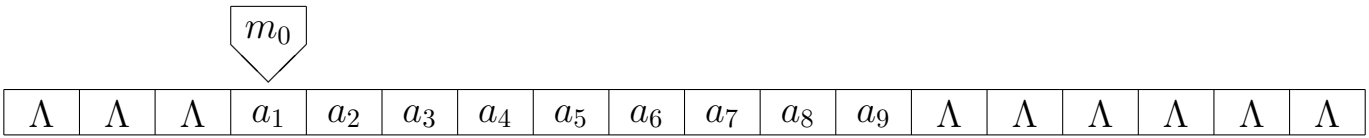


Иногда применяют табличное оформление машины Тьюринга:

	m_0	m_1	…	m_k	…	m_p	…	m_q	…	m_r	…
a_1											
…											
a_4											
a_5											
…											
a_{10}											
…											

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

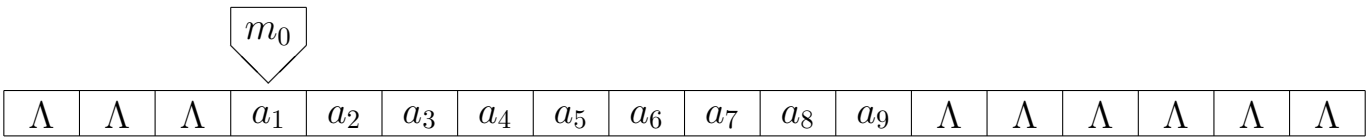


Иногда применяют табличное оформление машины Тьюринга:

	m_0 m_1 … m_k … m_p … m_q … m_r …
a_1	
…	
a_4	(a_{10}, m_p)
a_5	
…	
a_{10}	
…	

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

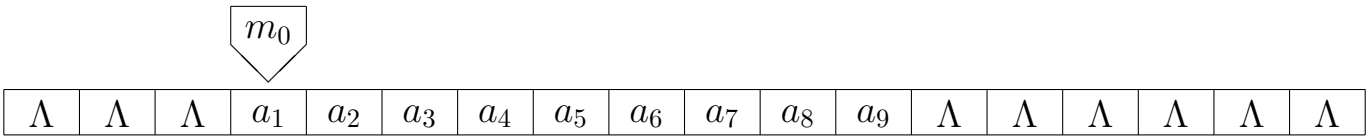


Иногда применяют табличное оформление машины Тьюринга:

	m_0 m_1 $\ldots$ m_k $\ldots$ m_p $\ldots$ m_q $\ldots$ m_r $\ldots$
a_1	
$\ldots$	
a_4	(a_{10}, m_p)
a_5	
$\ldots$	
a_{10}	
$\ldots$	

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

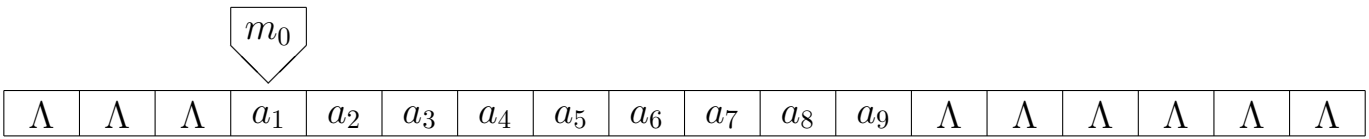


Иногда применяют табличное оформление машины Тьюринга:

	m_0 m_1 … m_k … m_p … m_q … m_r …
a_1	
…	
a_4	(a_{10}, m_p)
a_5	$L(a_{12}, m_r)$
…	
a_{10}	
…	

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

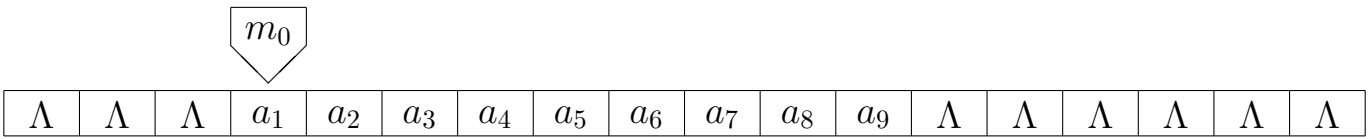


Иногда применяют табличное оформление машины Тьюринга:

	m_0 m_1 $\ldots$ m_k $\ldots$ m_p $\ldots$ m_q $\ldots$ m_r $\ldots$
a_1	
$\ldots$	
a_4	(a_{10}, m_p)
a_5	$L(a_{12}, m_r)$
$\ldots$	
a_{10}	
$\ldots$	

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L (a_{12} m_r), \ldots, (a_{10} m_p) R (a_{11} m_q), \ldots, \ldots\}$

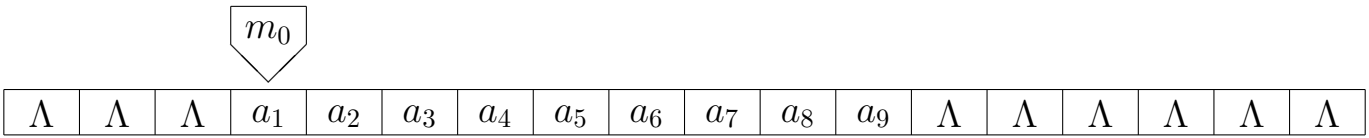


Иногда применяют табличное оформление машины Тьюринга:

	m_0 m_1 $\ldots$ m_k $\ldots$ m_p $\ldots$ m_q $\ldots$ m_r $\ldots$
a_1	
$\ldots$	
a_4	(a_{10}, m_p)
a_5	$L(a_{12}, m_r)$
$\ldots$	
a_{10}	$R(a_{11}, m_q)$
$\ldots$	

IV. Машина Тьюринга

$\mathbf{T} = \{\ldots, (a_4 m_k)(a_{10} m_p), \ldots, (a_5 m_q) L(a_{12} m_r), \ldots, (a_{10} m_p) R(a_{11} m_q), \ldots, \ldots\}$



Иногда применяют табличное оформление машины Тьюринга:

	m_0	m_1	…	m_k	…	m_p	…	m_q	…	m_r	…
a_1	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…
a_4	…	…	…	(a_{10}, m_p)	…	…	…	…	…	…	…
a_5	…	…	…	…	…	…	…	$L(a_{12}, m_r)$	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…
a_{10}	…	…	…	…	…	$R(a_{11}, m_q)$	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…

Рассмотрим пример?

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

Комментарий. Если интерпретировать входное слово, как число в двоичной записи, то есть незначащими считаются все нули, предшествующие первой «значащей цифре».

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

Комментарий. Если интерпретировать входное слово, как число в двоичной записи, то есть незначащими считаются все нули, предшествующие первой «значащей цифре».

Таким образом, первой значащей цифрой является:

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

Комментарий. Если интерпретировать входное слово, как число в двоичной записи, то есть незначащими считаются все нули, предшествующие первой «значащей цифре».

Таким образом, первой значащей цифрой является:
либо первая слева 1, если входное слово не состоит из одних нулей;

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

Комментарий. Если интерпретировать входное слово, как число в двоичной записи, то есть незначащими считаются все нули, предшествующие первой «значащей цифре».

Таким образом, первой значащей цифрой является:
либо первая слева 1, если входное слово не состоит из одних нулей;
или последняя 0, если входное слово состоит из одних нулей.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

1100101

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101$

Выполнение каждого шага начинается с попытки выполнения первой команды.

Только теперь удалось выполнить шаг алгорифма.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101$

Выполнение следующего шага тоже начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101$

Выполнение следующего шага тоже начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101$

Выполнение следующего шага тоже начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

Выполнение следующего шага тоже начинается с попытки выполнения *первой* команды!

Третий компонент команды равен 1, поэтому работа алгорифма нормально завершилась.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
000

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000$

Выполнение каждого шага начинается с попытки выполнения первой команды.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00$

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

Третий компонент команды равен 1, поэтому работа алгорифма нормально завершилась.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
00101

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101$

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101$

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$

Третий компонент команды равен 1, поэтому работа алгорифма нормально завершилась.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$

$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto$

Здесь под Λ понимается пустое слово.

Выполнение каждого шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнением этой команды завершился очередной шаг, но третий компонент команды равен 0, поэтому переходим к следующему шагу.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$
$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$
$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$
$\Lambda \mapsto c \mapsto cc \mapsto ccc$

Здесь под Λ понимается пустое слово.

Выполнение следующего шага начинается с попытки выполнения *первой* команды!

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$

$\Lambda \mapsto c \mapsto cc \mapsto ccc \mapsto cccc...$

Здесь под Λ понимается пустое слово.

Ясно, что этот процесс никогда не закончится.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$

$\Lambda \mapsto c \mapsto cc \mapsto ccc \mapsto cccc...$

Здесь под Λ понимается пустое слово.

Ясно, что этот процесс никогда не закончится.

Итак, пустое слово не поддается алгоритму.

Пример 1. *Проверьте, что алгорифм*

$$\mathbf{Alg} = ((c1, 1, 1), (c00, c0, 0), (c01, 1, 1), (c0, 0, 1), (, c, 0))$$

во входном слове в алфавите $\{0, 1\}$ *удаляет «незначащие нули». Найдите результат работы алгорифма на входных словах* **1100101**, **000**, **00101** *и* ***пустом слове***.

$1100101 \mapsto c1100101 \mapsto\bullet\ 1100101,$

$000 \mapsto c000 \mapsto c00 \mapsto c0 \mapsto\bullet\ 0,$

$00101 \mapsto c00101 \mapsto c0101 \mapsto\bullet\ 101,$

$\Lambda \mapsto c \mapsto cc \mapsto ccc \mapsto cccc...$

Здесь под Λ понимается пустое слово.

Ясно, что этот процесс никогда не закончится.

Итак, пустое слово не поддается алгоритму.

Вернемся к лекции или рассмотрим **другой пример?**

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0; 1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.

Первый алгоритм:

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.

Первый алгоритм: 1) на левом конце входного слова поставить c;

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle$ $\rangle$.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle$ [redacted] $\rangle$.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.
Для написания алгорифма применим стратегию предвкушения.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to$

Эта команда будет последней. После нее никакая команда выполняться не будет, поскольку пустое подслово есть в любом слове!

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.

Для написания алгорифма применим стратегию предвкушения.

$01110011 \to c01110011 \to 0c1110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;

2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;

3) c «перепрыгивает» через одиночную 1 «без последствий»;

4) c «перепрыгивает» через 0 «без последствий»;

5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.

Для написания алгорифма применим стратегию предвкушения.

$01110011 \to c01110011 \to 0c1110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;

2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;

3) c «перепрыгивает» через одиночную 1 «без последствий»;

4) c «перепрыгивает» через 0 «без последствий»;

5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad (c0, 0c, 0); \qquad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad (c0, 0c, 0); \qquad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad (c0, 0c, 0); \qquad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle \qquad (c0, 0c, 0); \qquad (\Lambda; c; 0) \rangle$.

Для написания алгорифма применим стратегию предвкушения.

$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;

2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;

3) c «перепрыгивает» через одиночную 1 «без последствий»;

4) c «перепрыгивает» через 0 «без последствий»;

5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); \quad (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); \quad (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0); \quad (c0,0c,0); \quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); \quad (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); \quad (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); \quad (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); (c1; 1c; 0); (c0, 0c, 0); \quad (\Lambda; c; 0) \rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0); \quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 010101001001c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0); \quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);\quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0); \quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0); \quad (\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c$
Команда $(c;\Lambda;1)$ будет после первых трех команд, иначе она их «экранирует».

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c \to \bullet\ 01010100101$

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
Для написания алгорифма применим стратегию предвкушения.
$01110011 \to c01110011 \to 0c1110011 \to 010c110011 \to 01010c10011 \to$
$\to 010101c\,0011 \to 0101010c\,011 \to 01010100c11 \to 0101010010c1 \to$
$\to 01010100101c \to \bullet\ 01010100101$
Теперь надо бы протестировать данный алгоритм «целиком», но мы оставим это вам на самостоятельную работу.

Первый алгоритм: 1) на левом конце входного слова поставить c;
2) c, «увидев справа 11» вставляет 0 между единицами и встает перед второй 1;
3) c «перепрыгивает» через одиночную 1 «без последствий»;
4) c «перепрыгивает» через 0 «без последствий»;
5) когда справа от c пустота, алгоритм «штатно завершается».

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); (c1; 1c; 0); (c0, 0c, 0); (c; \Lambda; 1); (\Lambda; c; 0) \rangle$.
$\mathcal{A}_2 = \langle \qquad\qquad\qquad\qquad \rangle$.

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); (c1; 1c; 0); (c0, 0c, 0); (c; \Lambda; 1); (\Lambda; c; 0) \rangle$.
$\mathcal{A}_2 = \langle \qquad\qquad\qquad\qquad \rangle$.

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle \qquad\qquad \rangle$.
$01110011 \to$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.

$\mathcal{A}_2 = \langle \qquad\qquad \rangle$.

$01110011 \to$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);

2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11; 10c1; 0); (c1; 1c; 0); (c0, 0c, 0); (c; \Lambda; 1); (\Lambda; c; 0) \rangle$.

$\mathcal{A}_2 = \langle \quad \rangle$.

$0\underline{11}10011 \to 0\underline{101}10011 \to$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);

2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0); \quad \rangle$.
$011110011 \to 010110011 \to$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);\quad\quad\quad

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);\quad\quad\quad

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);\quad\quad\quad

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);\quad\quad\quad

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);\quad\quad\quad

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);(\Lambda;\Lambda;1)\rangle$.
$01110011 \to 010110011 \to 0101010011 \to 01010100101 \to$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);(\Lambda;\Lambda;1)\rangle$.
$01110011 \to 010110011 \to 0101010011 \to 01010100101 \to\bullet\ 01010100101$

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);(\Lambda;\Lambda;1)\rangle$.
$01110011 \to 010110011 \to 0101010011 \to 01010100101 \to\bullet\ 01010100101$
Второй алгоритм намного проще!

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Пример 2. *Написать нормальный алгорифм Маркова, в слове в алфавите* $\{0;1\}$ *между «сдвоенными единицами» вставляющий 0, например,* $01110011 \mapsto \ldots \mapsto 01010100101$.

Решение. $\mathcal{A}_1 = \langle (c11;10c1;0);(c1;1c;0);(c0,0c,0);(c;\Lambda;1);(\Lambda;c;0)\rangle$.
$\mathcal{A}_2 = \langle (11;101;0);(\Lambda;\Lambda;1)\rangle$.
$01110011 \to 010110011 \to 0101010011 \to 01010100101 \to\bullet\ 01010100101$
Второй алгоритм намного проще!

Второй алгоритм: 1) пары 11 заменять на 101 (слева направо);
2) если 11 больше нет, остановить алгоритм.

Вернемся к лекции или рассмотрим **другой пример?**

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

Идея алгорифма:

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

1111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$\rangle$

1111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$(\Lambda, c, 0)\rangle$

1111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$(\Lambda, c, 0)\rangle$

$1111 \to c1111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$(c1, 1ce, 0)$, $(\Lambda, c, 0)\rangle$

$1111 \to c1111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:

после появления c или сдвига c вправо высылается «гонец» e в конец слова.

Гонец либо доходит до конца слова, «превращаясь» в d,

либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$(c1, 1ce, 0)$, $(\Lambda, c, 0)\rangle$

$1111 \to c1111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:

после появления c или сдвига c вправо высылается «гонец» e в конец слова.

Гонец либо доходит до конца слова, «превращаясь» в d,

либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle$

$(c1, 1ce, 0)$, $(\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce11 \to 1c1e11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (c1, 1ce, 0), (\Lambda, c, 0)\rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1111 \to c1111 \to 1ce111 \to 1c1e11 \to 1c11e1 \to 1c111e \to 1c11d1 \to 11ce1d1 \to$
$\to 11c1ed1 \to 11cd11 \to \bullet 1111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

111

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c11 \to 1ce11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1 \to 11ced1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1 \to 11ced1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0),$
$(c1, 1ce, 0), (cd, \Lambda, 1), \quad (\Lambda, c, 0)\rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1 \to 11ced1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0),$
$(c1, 1ce, 0), (cd, \Lambda, 1), \quad (\Lambda, c, 0)\rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1 \to 11ced1 \to \bullet 101$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$111 \to c111 \to 1ce11 \to 1c1e1 \to 1c11e \to 1c1d1 \to 11ced1 \to \bullet 101$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1)\,, (1ed, d1, 0)\,, (e1, 1e, 0)\,, (1e, d1, 0)\,,$
$(c1, 1ce, 0)\,, (cd, \Lambda, 1)\,, \qquad (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0),$
$(c1, 1ce, 0), (cd, \Lambda, 1), \quad (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

1

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0),$
$(c1, 1ce, 0), (cd, \Lambda, 1), \quad (\Lambda, c, 0)\rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1)\,, (1ed, d1, 0)\,, (e1, 1e, 0)\,, (1e, d1, 0)\,,$
$(c1, 1ce, 0)\,, (cd, \Lambda, 1)\,, \quad (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0),$
$(c1, 1ce, 0), (cd, \Lambda, 1), \quad (\Lambda, c, 0)\rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (1ce, 0, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (1ce, 0, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce \to \bullet 0$

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (1ce, 0, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce \to \bullet 0$

Надо бы проверить весь алгоритм целиком...

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (1ce, 0, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce \to \bullet 0$

Надо бы проверить весь алгоритм целиком... Сделайте сами.

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0.

Пример 3. *Написать нормальный алгорифм Маркова, оставляющий неизменным слово из единиц четной длины, а случае нечетной длины слова из единиц заменяющего центральную единицу на 0.*

Решение. $\mathcal{A} = \langle (1ced, 0, 1), (1ed, d1, 0), (e1, 1e, 0), (1e, d1, 0), (c1, 1ce, 0), (cd, \Lambda, 1), (1ce, 0, 1), (\Lambda, c, 0) \rangle$

$1 \to c1 \to 1ce \to \bullet 0$

Надо бы проверить весь алгоритм целиком... Сделайте сами.

Идея алгорифма: выставить в начале слова «курсор» c, а в конце — d, заставить их по очереди сходиться к центру:
после появления c или сдвига c вправо высылается «гонец» e в конец слова.
Гонец либо доходит до конца слова, «превращаясь» в d,
либо упирается в d, и перед исчезновением смещает d на шаг влево.

При встрече «курсоры» либо «аннигилируют», либо обращают среднюю единицу в 0. **Вернемся к лекции?**

Задача IV.1. (Ответ приведен на стр.720.) Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Задача IV.2. (Ответ приведен на стр.749.) Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Пример 4. *Проверить, что машина Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$

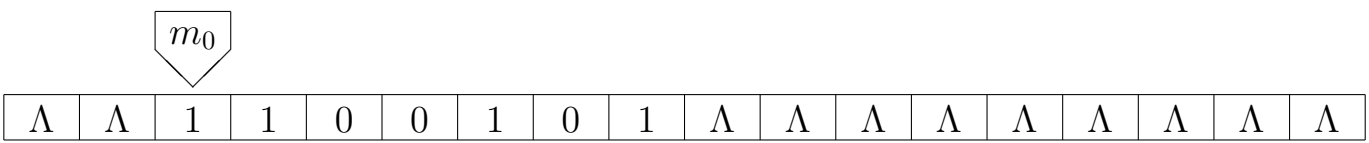
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

с внутренним алфавитом $\{m_0, m_1, a, b\}$ *и внешним алфавитом* $\{0, 1\}$ *делает то же, что и нормальный алгорифм в* ***примере 1****, то есть «гасит» не значащие нули перед двоичным числом. Убедиться в этом, применив эту машину Тьюринга к входным словам* 1100101, 000, 00101 *и пустому слову.*

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

Пример 4. *Применить машину Тьюринга*

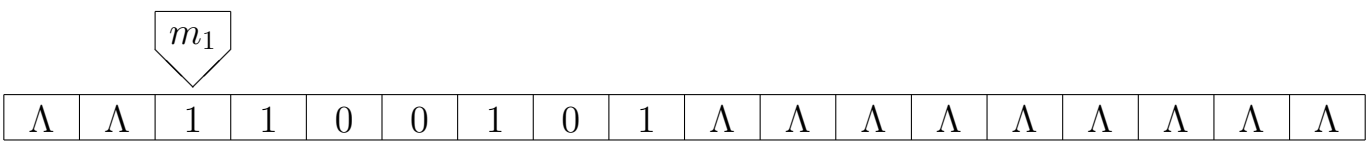
$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

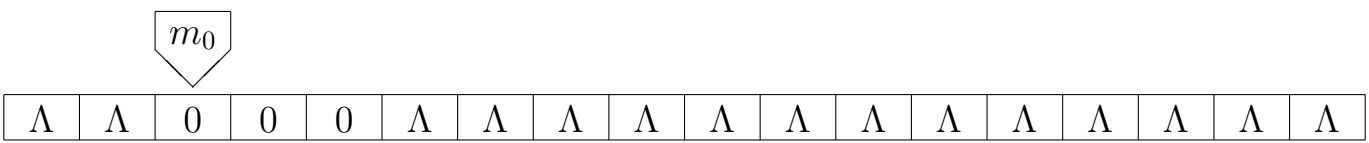


Все правильно, во входном слове не было незначащих нулей.

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

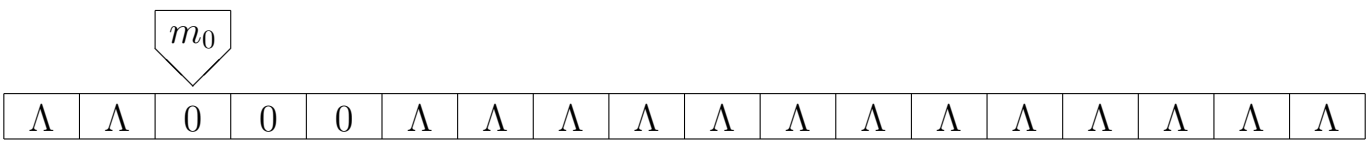
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

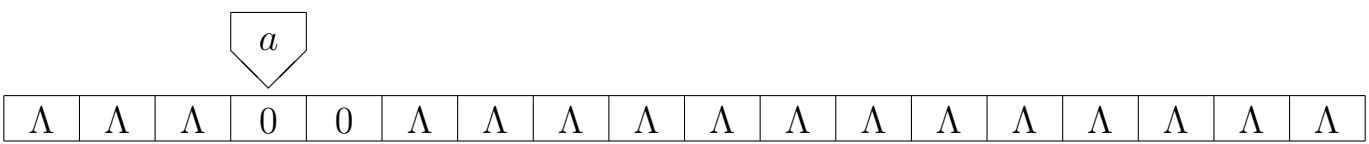
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

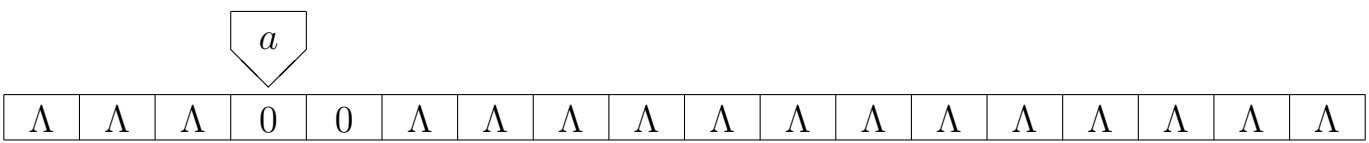
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

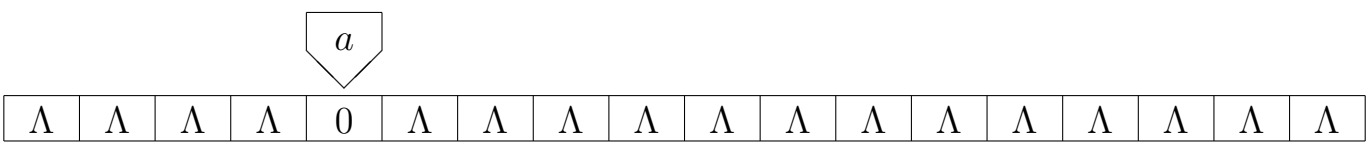
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

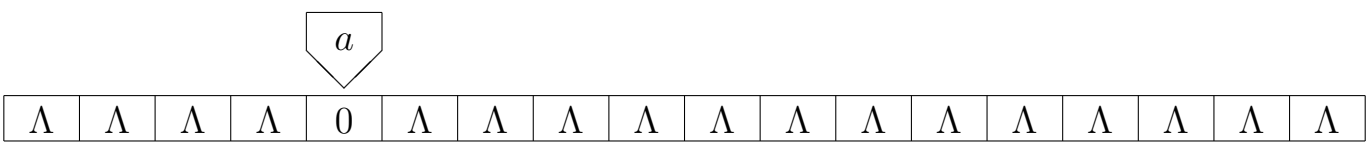
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

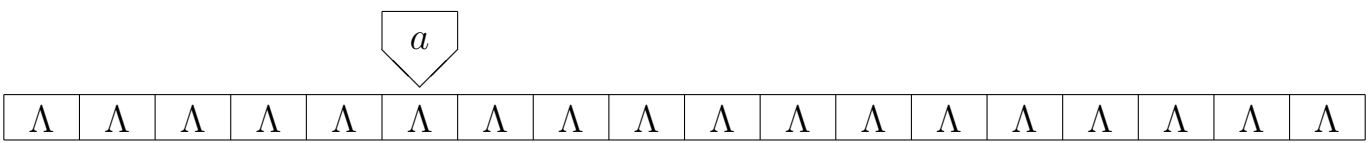
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

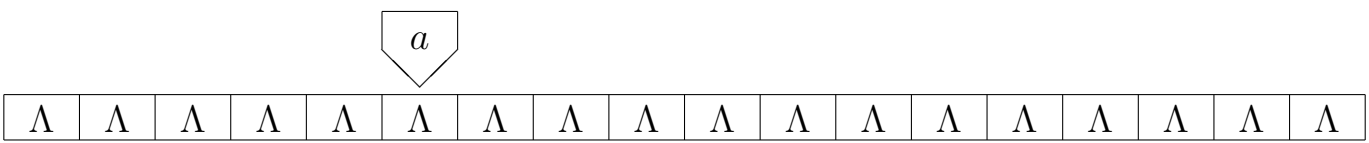
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

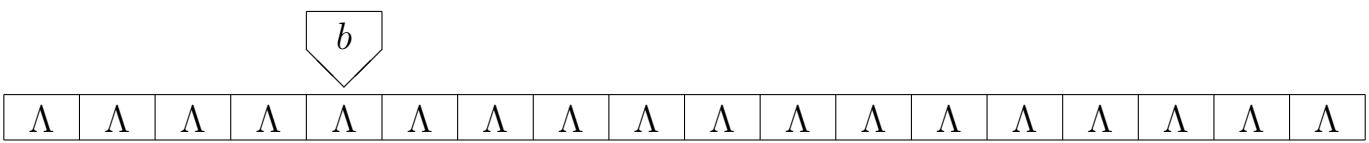
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

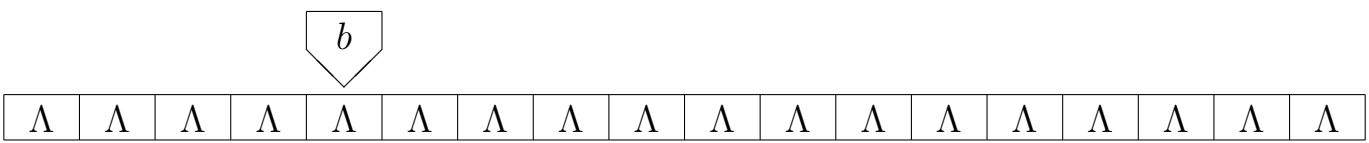
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

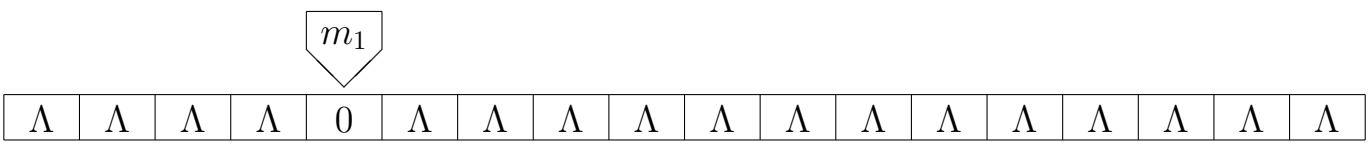
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

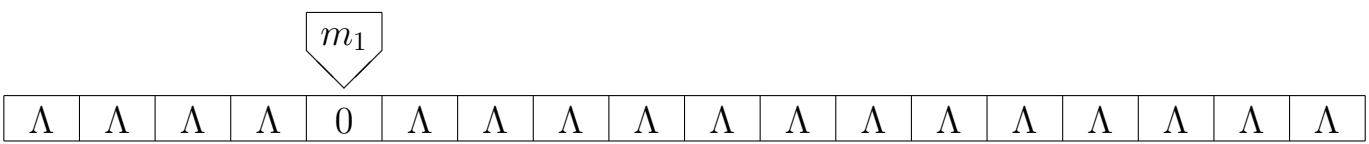
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

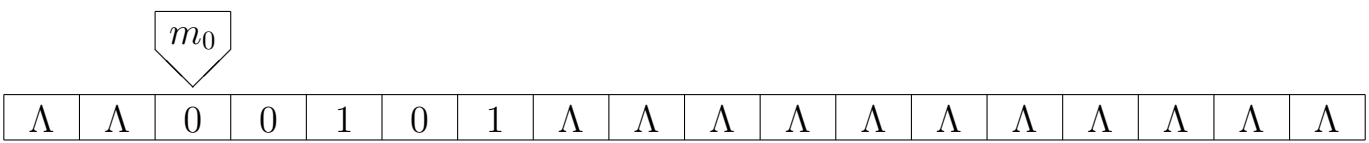


Данный ноль является значимым.

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

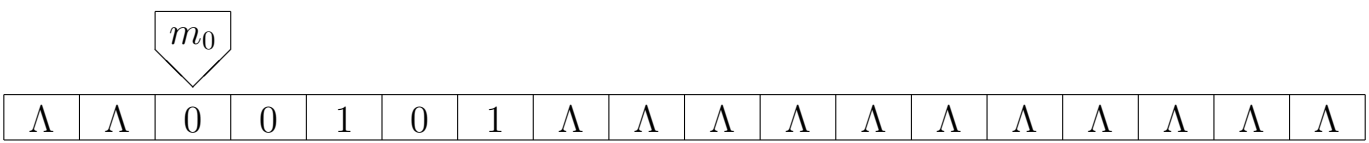
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

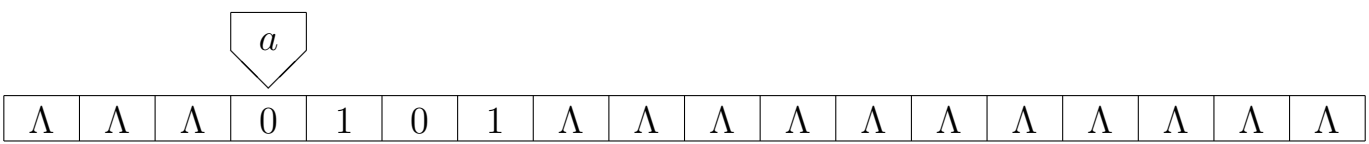
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

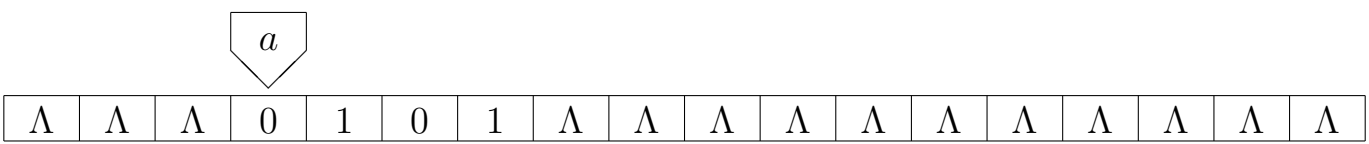
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

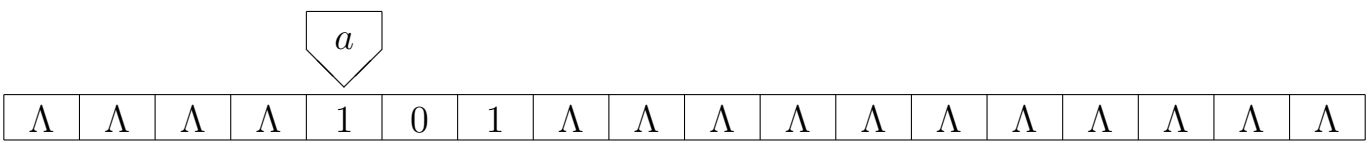
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

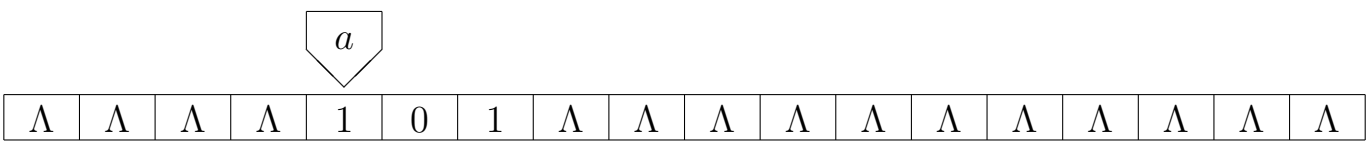
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

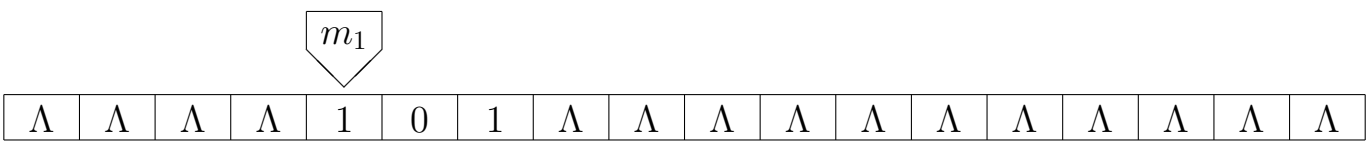
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

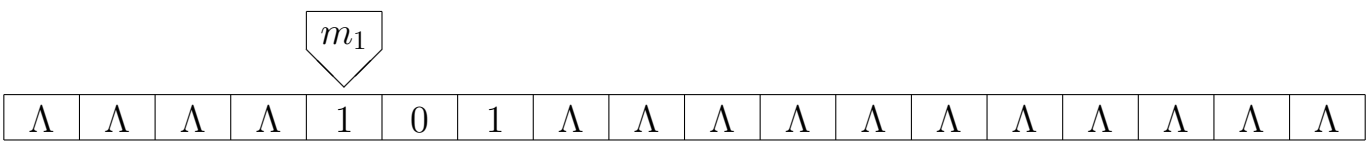
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

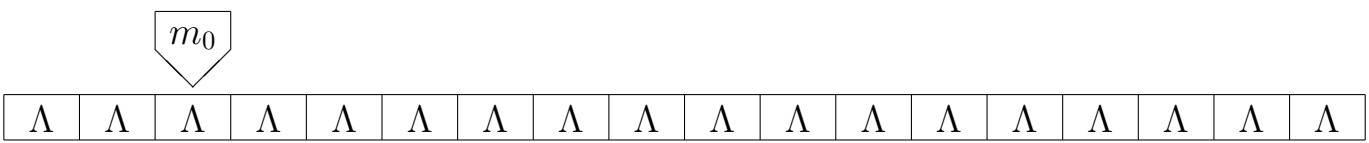


Машина Тьюринга удалила два незначащих нуля.

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

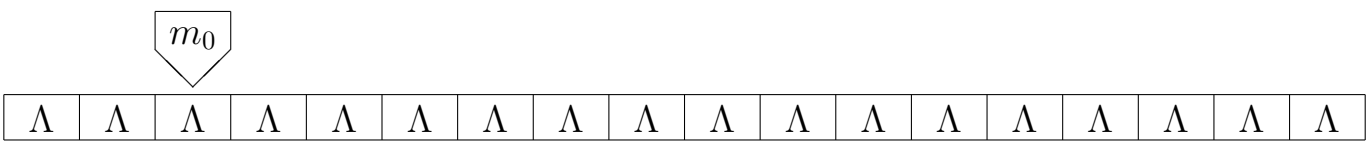
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

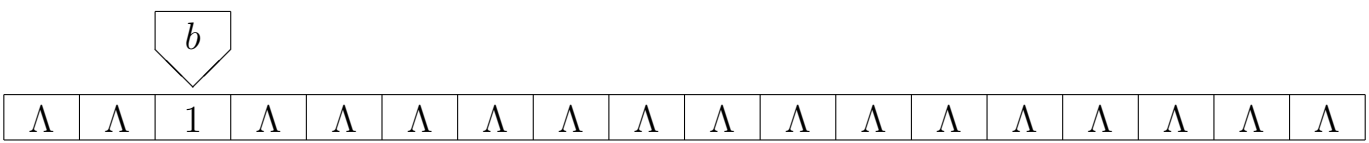
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

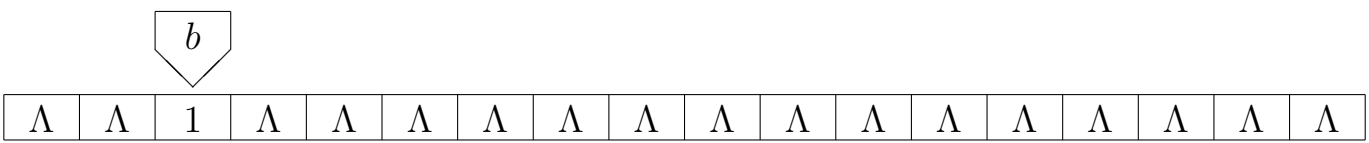
к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.

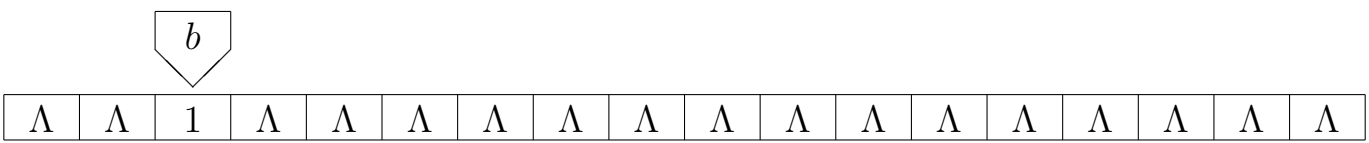


Работа машины Тьюринга завершилась аварийно, поскольку в системе команд нет команды со «входной частью» $(b, 1)$.

Пример 4. *Применить машину Тьюринга*

$$\mathbf{P} = \{(0, m_0)R(\Lambda, a), (\Lambda, m_0)(1, b), (1, m_0)(1, m_1),$$
$$(0, a)R(\Lambda, a), (\Lambda, a)L(\Lambda, b), (\Lambda, b)(0, m_1), (1, a)(1, m_1)\}$$

к входным словам **1100101**, **000**, **00101** *и* ***пустому слову***.



Работа машины Тьюринга завершилась аварийно, поскольку в системе команд нет команды со «входной частью» $(b, 1)$.

Вернемся к лекции или рассмотрим **другой пример?**

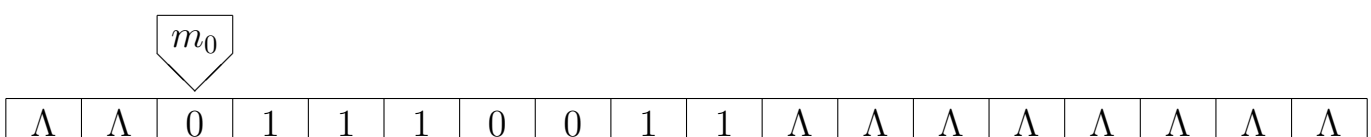
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

Для написания программы применим стратегию предвкушения.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0											
1											



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

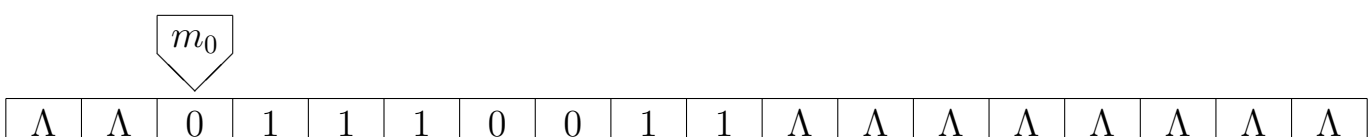
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0											
1											

Λ	Λ	0	1	1	1	0	0	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Если в состоянии m_0 головка видит 0, надо просто сдвинуться вправо, «не меняя выражения лица».

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

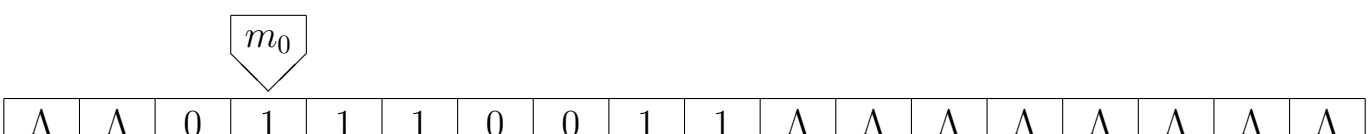
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1											



Если в состоянии m_0 головка видит 0, надо просто сдвинуться вправо, «не меняя выражения лица».

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

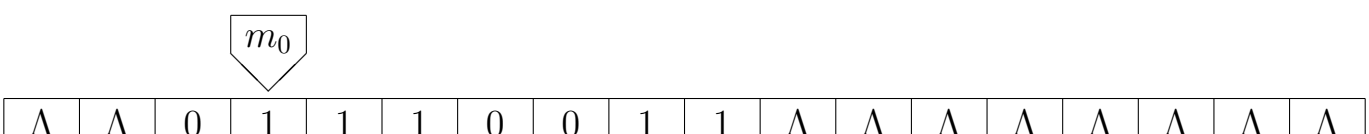
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1											



Если в состоянии m_0 головка видит 0, надо просто сдвинуться вправо, «не меняя выражения лица».

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

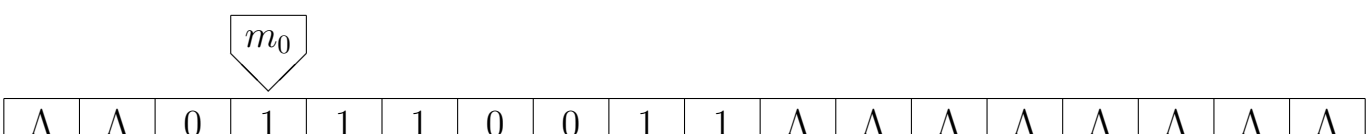
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1											



Если в состоянии m_0 головка видит 1, надо забеспокоиться (а вдруг дальше еще одна 1), т.е. перейти в m_2, и сдвинуться вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

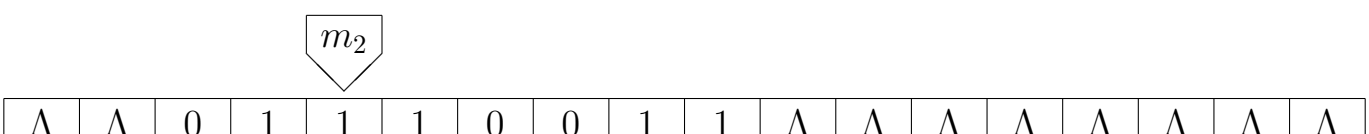
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$										



Если в состоянии m_0 головка видит 1, надо забеспокоиться (а вдруг дальше еще одна 1), т.е. перейти в m_2, и сдвинуться вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

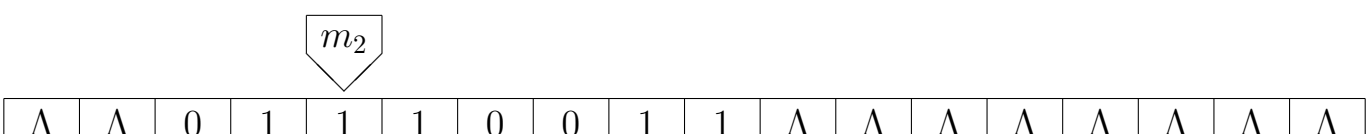
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$										



Если в состоянии m_0 головка видит 1, надо забеспокоиться (а вдруг дальше еще одна 1), т.е. перейти в m_2, и сдвинуться вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

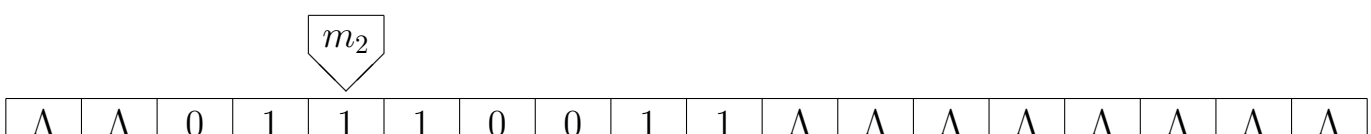
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$										



Обнаружив 1 в состоянии m_2, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

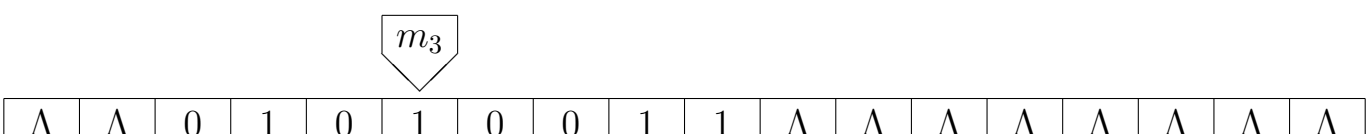
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$								



Обнаружив 1 в состоянии m_2, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

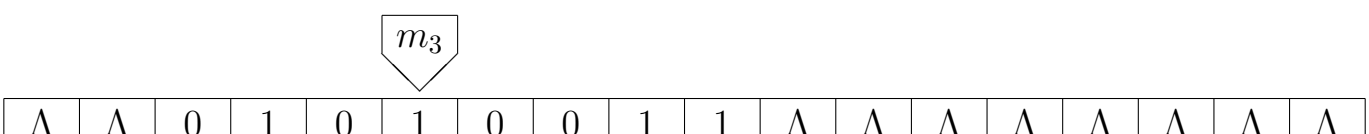
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$								



Обнаружив 1 в состоянии m_2, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

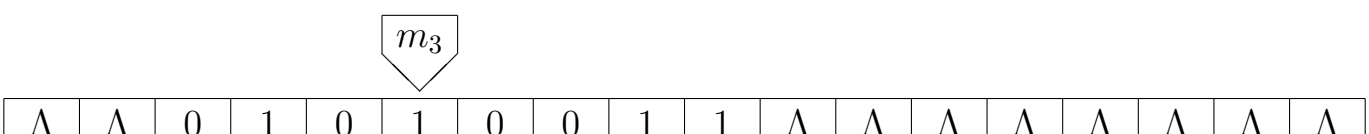
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$								



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

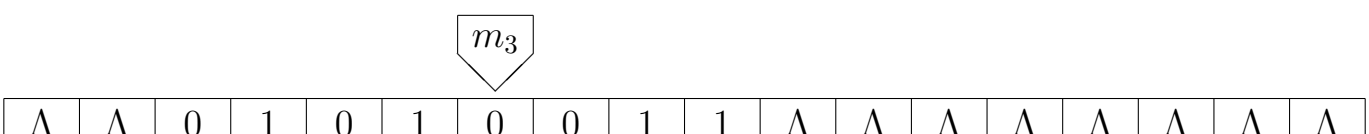
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

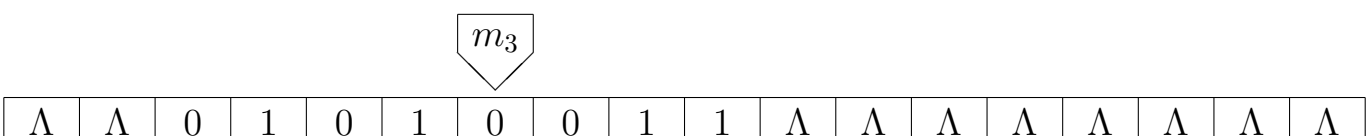
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

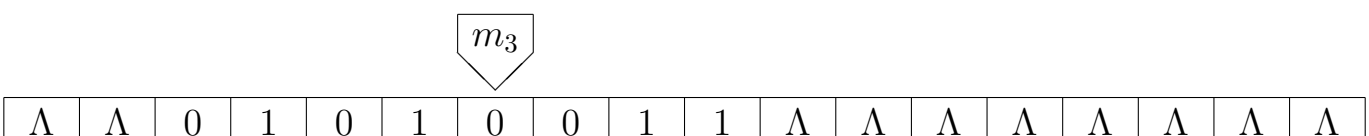
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$										
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_3, запомним этот 0 состоянием m_4, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

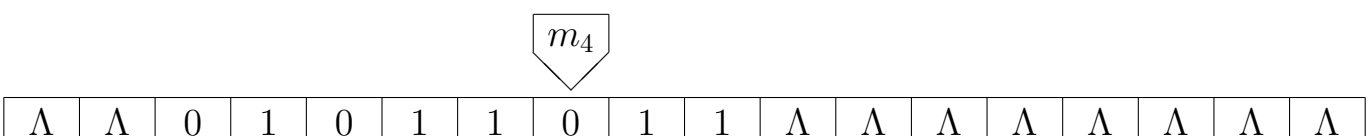
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$							
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_3, запомним этот 0 состоянием m_4, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

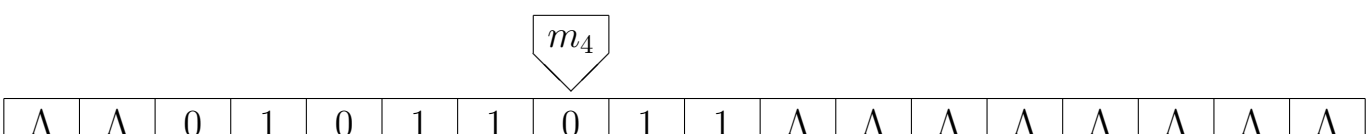
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$							
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_3, запомним этот 0 состоянием m_4, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

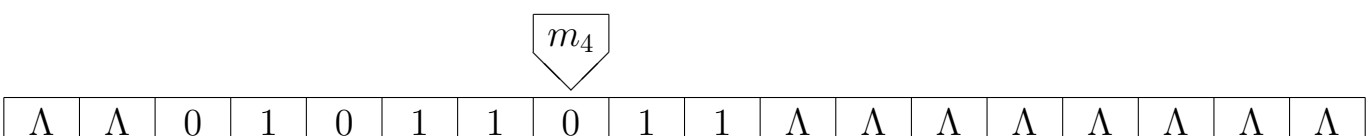
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$							
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_4, запомним этот 0 состоянием m_4, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

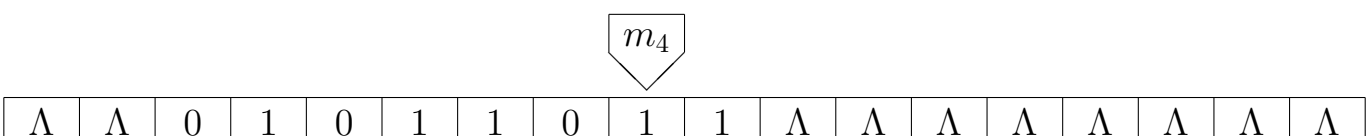
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_4, запомним этот 0 состоянием m_4, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							



Обнаружив 0 в состоянии m_4, запомним этот 0 состоянием m_4, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$							

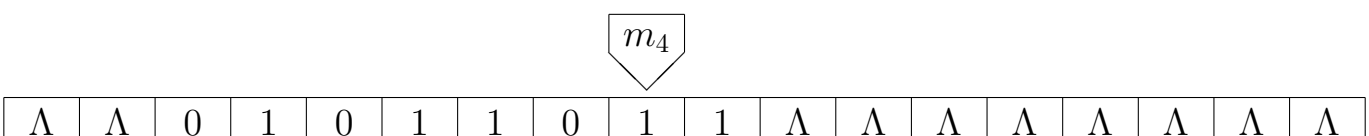
m_4

Λ	Λ	0	1	0	1	1	0	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Обнаружив 1 в состоянии m_4, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

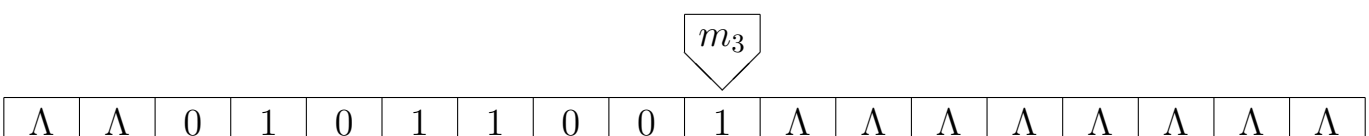
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_4, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

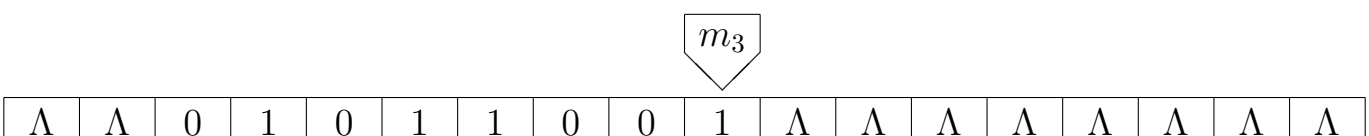
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_4, запомним эту 1 состоянием m_3, «вывалим» в ячейку 0 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

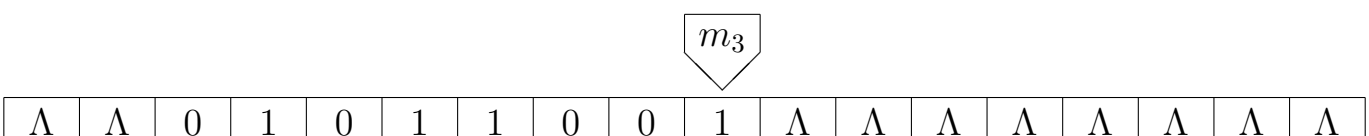
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

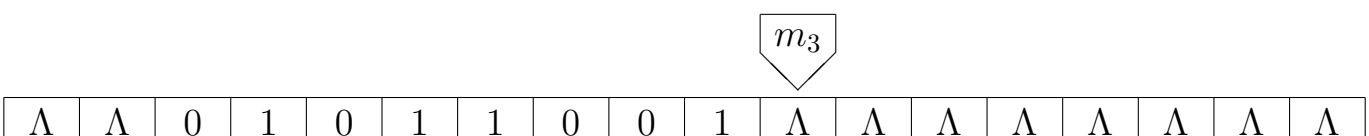
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

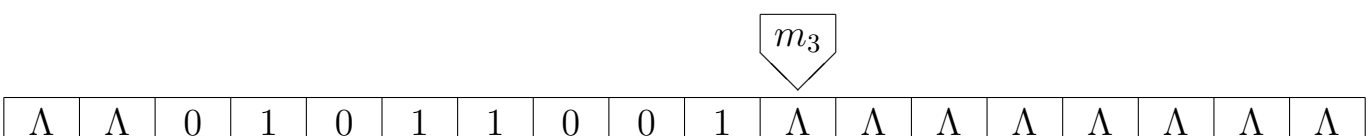
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_3, запомним эту 1 состоянием m_3, «вывалим» в ячейку 1 и сдвинемся вправо.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

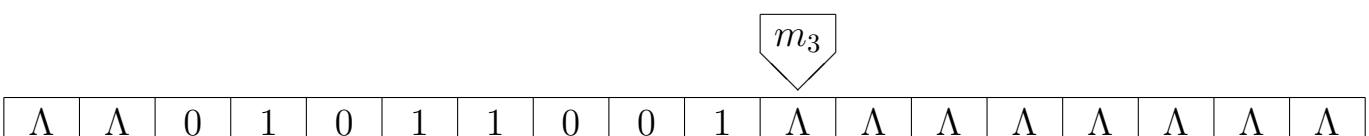
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ											
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив Λ в состоянии m_3, перейдем в состояние m_5 (поиск начала слова), «вывалим» в ячейку 1 и сдвинемся влево.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

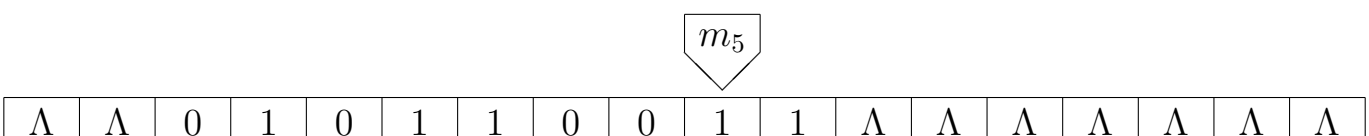
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив Λ в состоянии m_3, перейдем в состояние m_5 (поиск начала слова), «вывалим» в ячейку 1 и сдвинемся влево.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

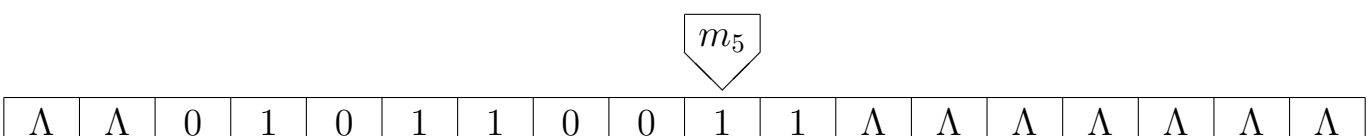
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив Λ в состоянии m_3, перейдем в состояние m_5 (поиск начала слова), «вывалим» в ячейку 1 и сдвинемся влево.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

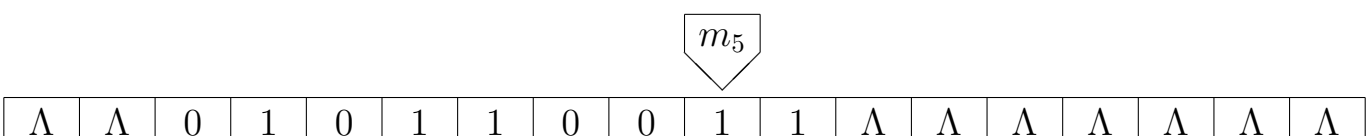
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$						



Обнаружив 1 в состоянии m_5 сдвинемся влево, оставив в ячейке 1.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

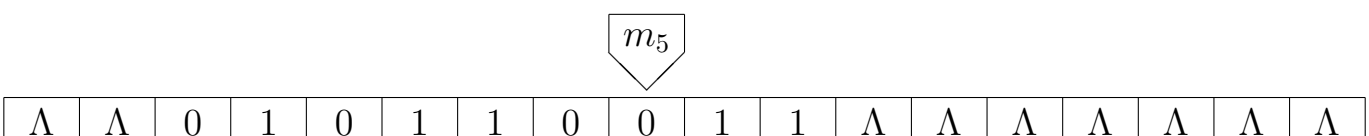
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 1 в состоянии m_5 сдвинемся влево, оставив в ячейке 1.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

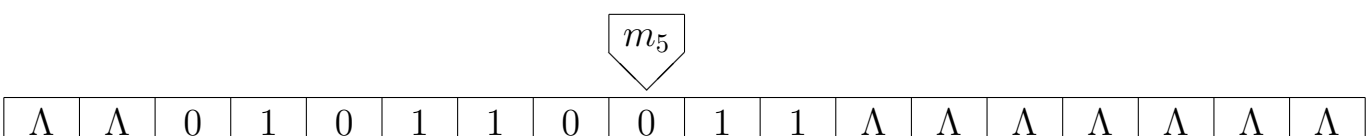
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 1 в состоянии m_5 сдвинемся влево, оставив в ячейке 1.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

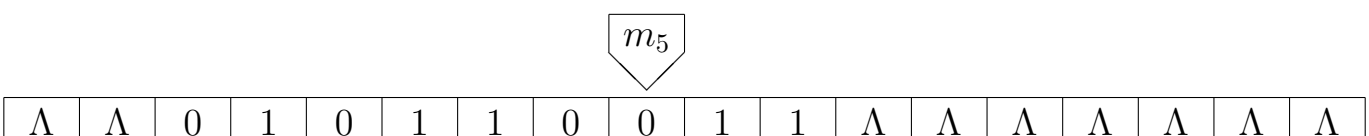
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$						
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_5 сдвинемся влево, оставив в ячейке 0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

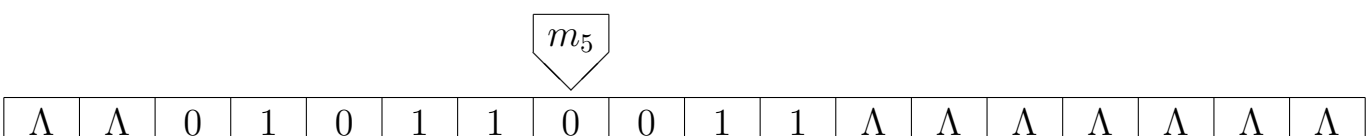
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_5 сдвинемся влево, оставив в ячейке 0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

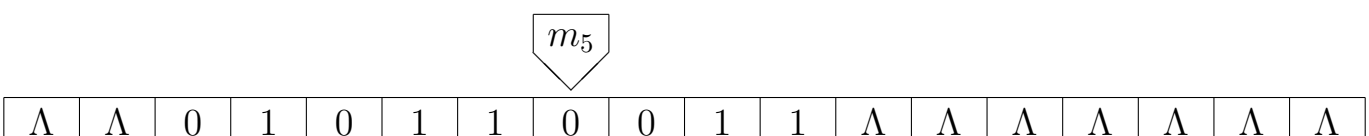
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_5 сдвинемся влево, оставив в ячейке 0.

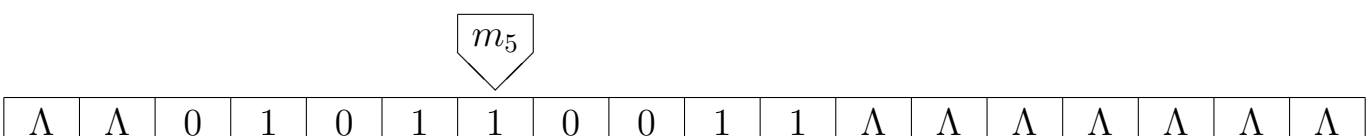
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



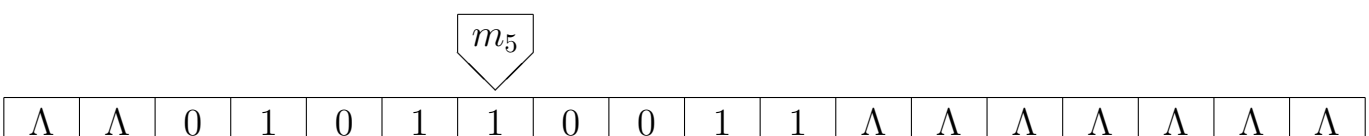
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



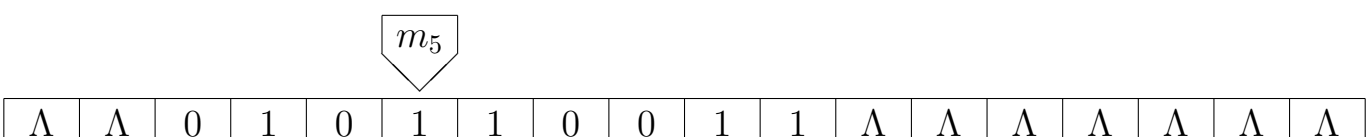
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



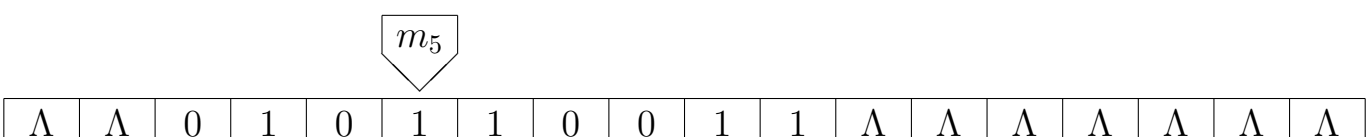
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



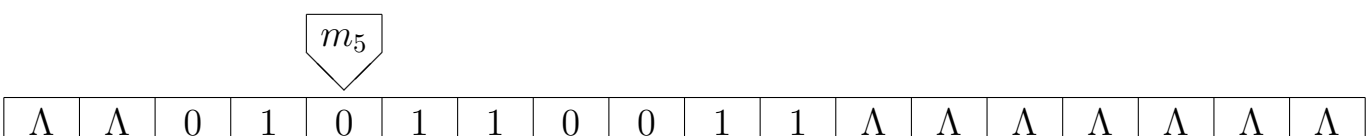
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



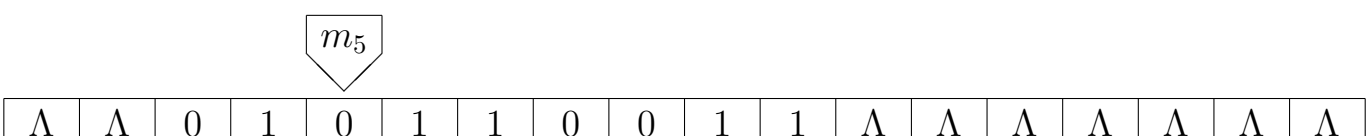
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



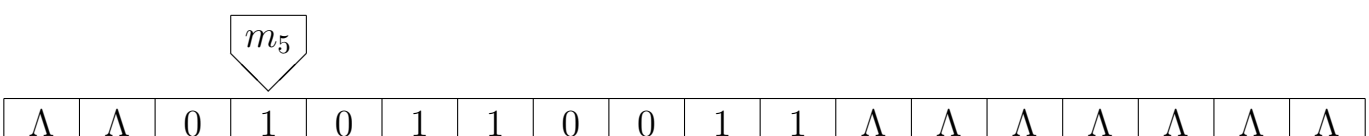
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



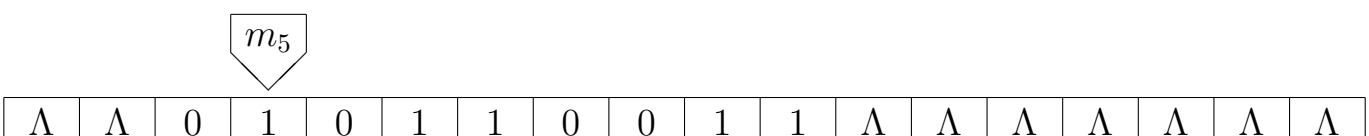
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



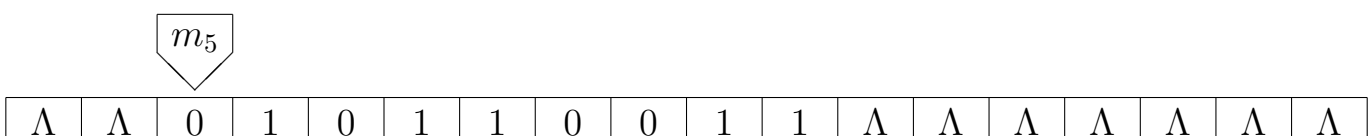
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

Λ	Λ	0	1	0	1	1	0	0	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ

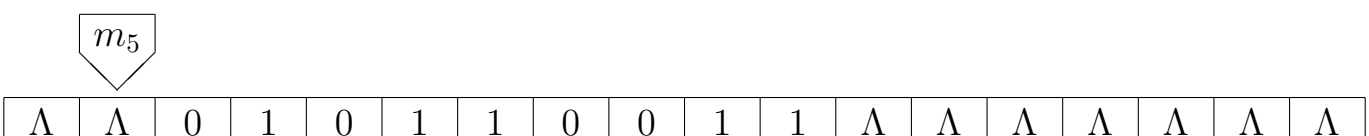
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



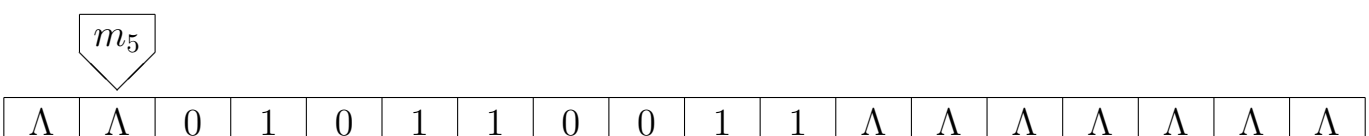
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

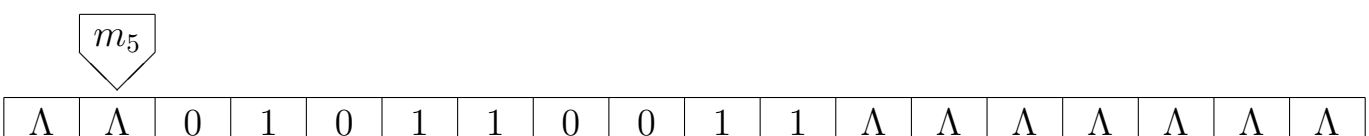
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$							
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив Λ в состоянии m_5 сдвинемся вправо, оставив в ячейке Λ, перейдя в состояние m_0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

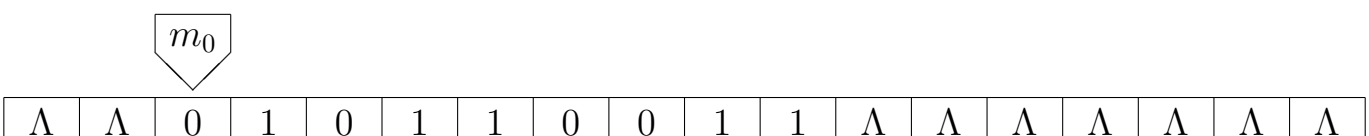
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив Λ в состоянии m_5 сдвинемся вправо, оставив в ячейке Λ, перейдя в состояние m_0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

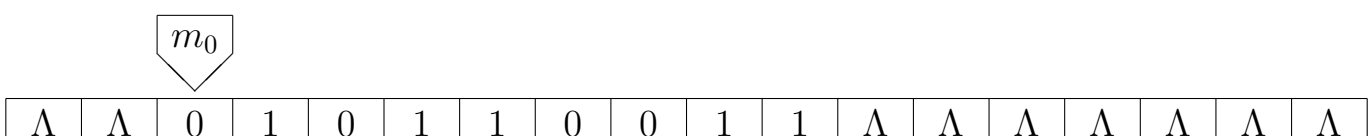
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив Λ в состоянии m_5 сдвинемся вправо, оставив в ячейке Λ, перейдя в состояние m_0.

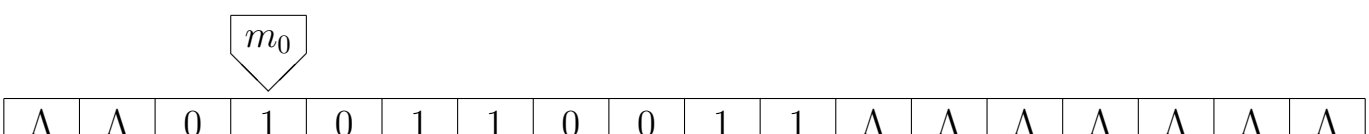
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



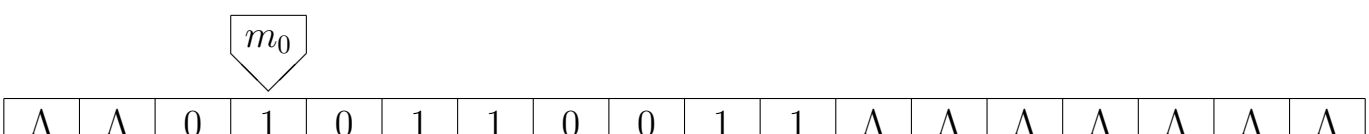
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



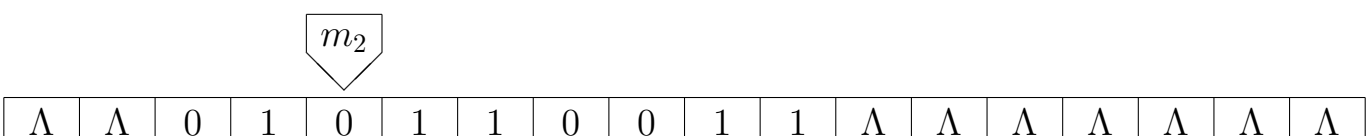
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



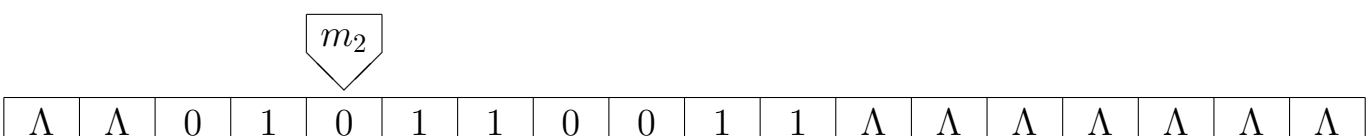
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

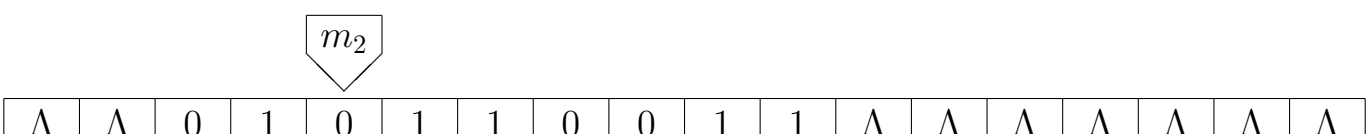
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$			$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_2 сдвинемся вправо, оставив в ячейке 0, вернувшись в состояние m_0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

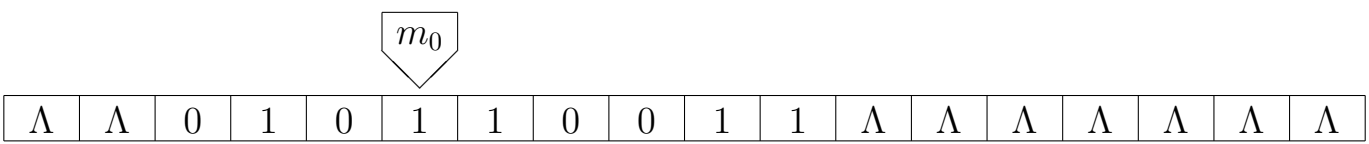
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_2 сдвинемся вправо, оставив в ячейке 0, вернувшись в состояние m_0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

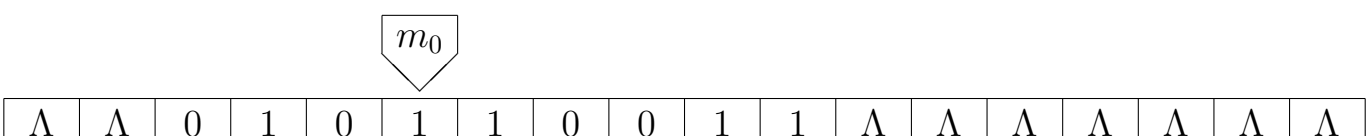
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Обнаружив 0 в состоянии m_2 сдвинемся вправо, оставив в ячейке 0, вернувшись в состояние m_0.

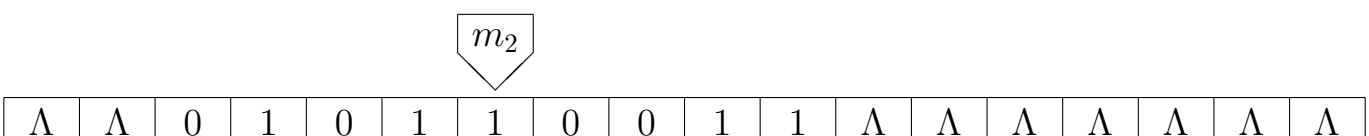
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



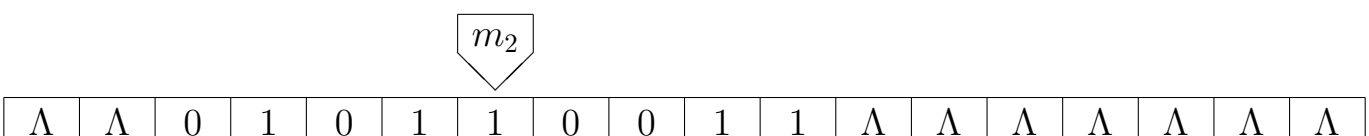
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



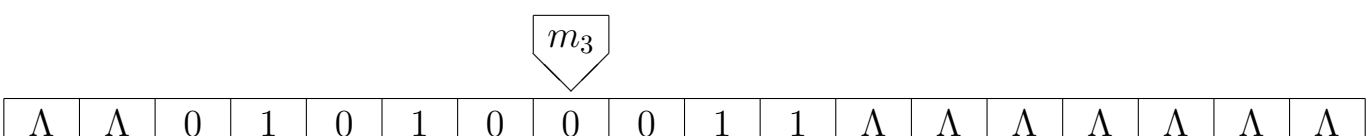
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



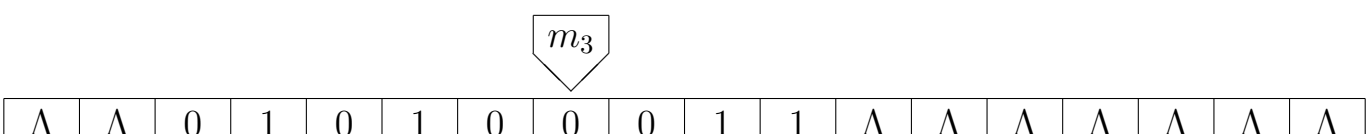
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



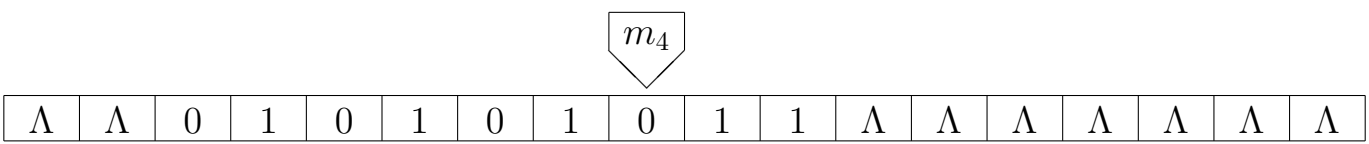
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



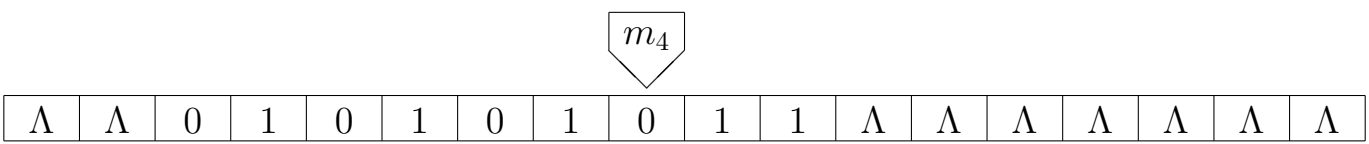
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



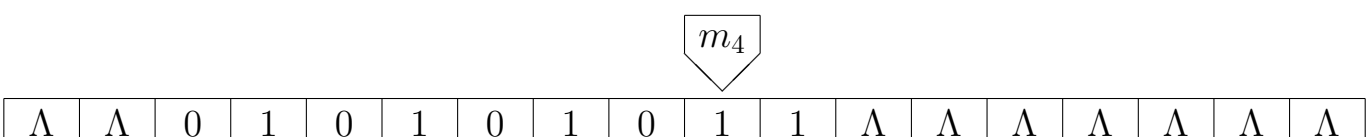
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



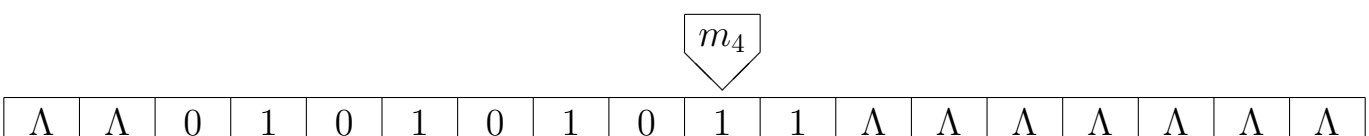
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



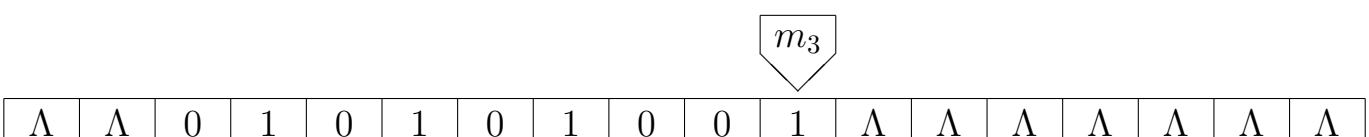
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



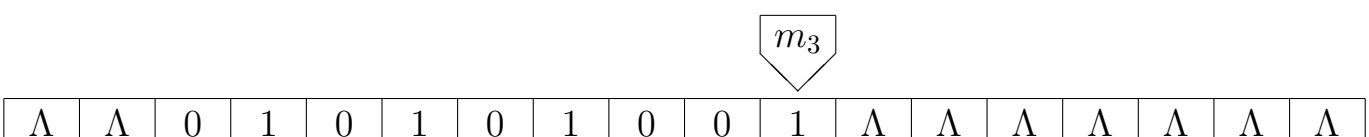
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



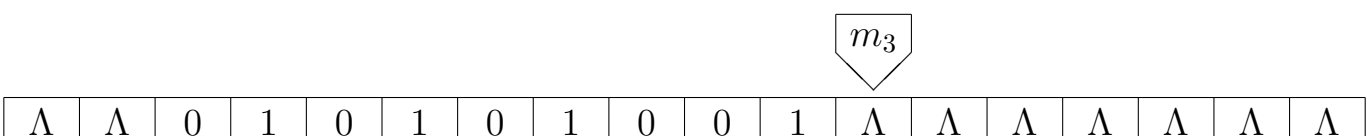
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



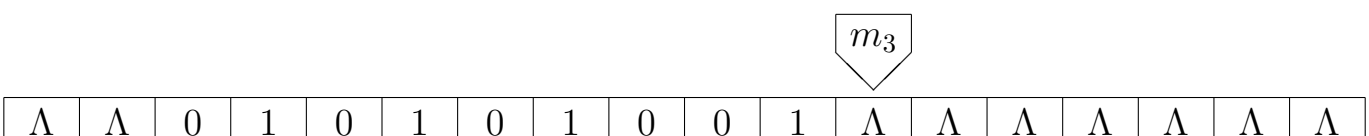
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



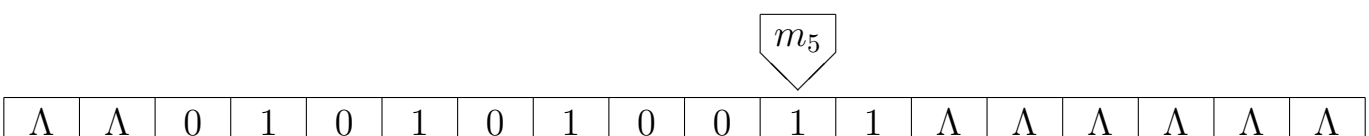
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



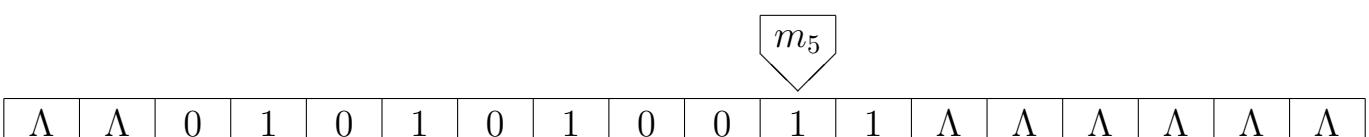
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



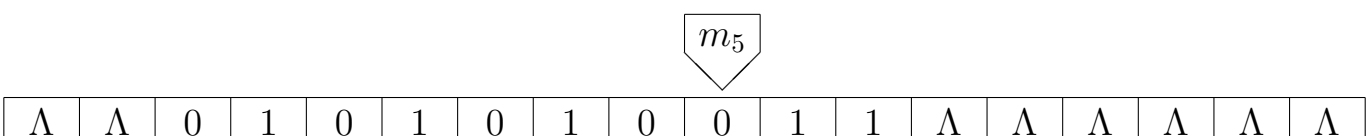
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



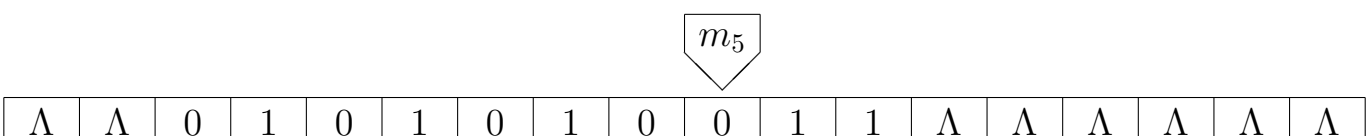
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



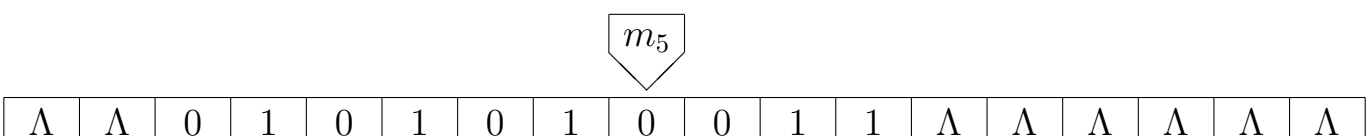
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



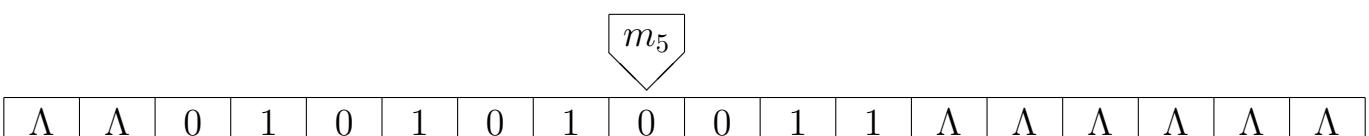
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



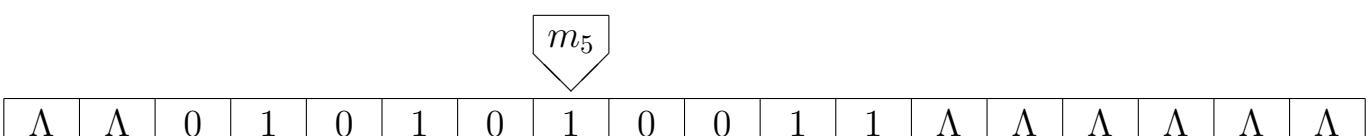
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



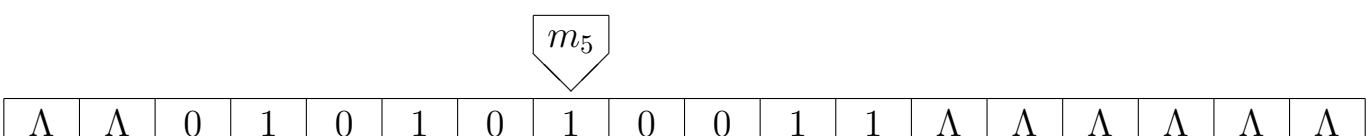
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



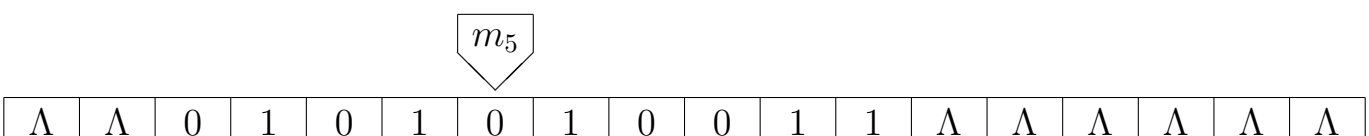
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



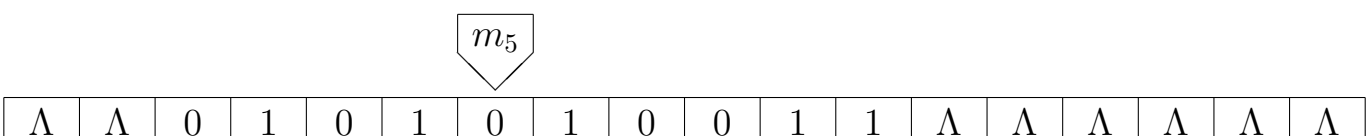
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



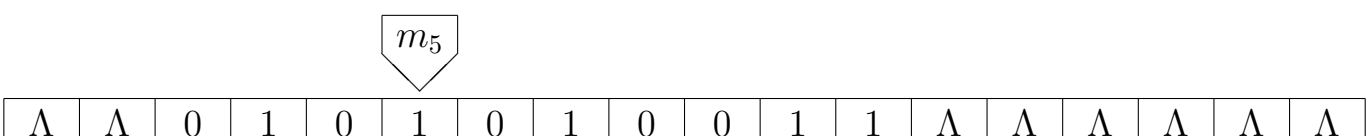
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



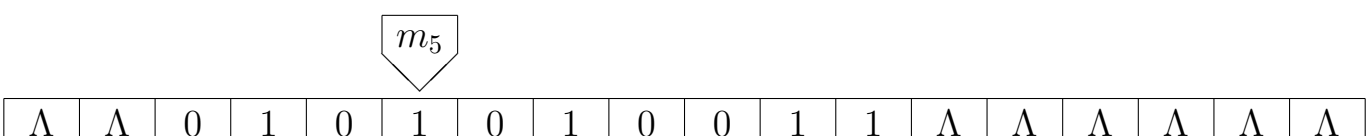
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



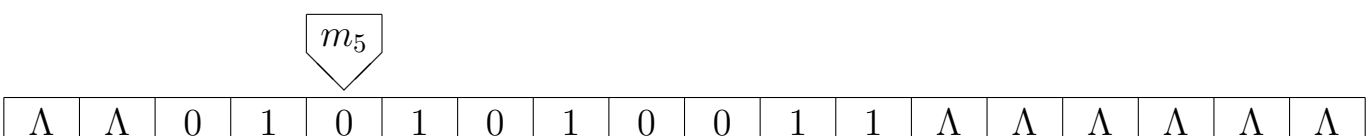
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



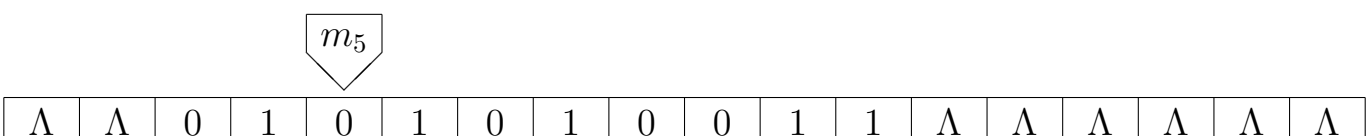
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



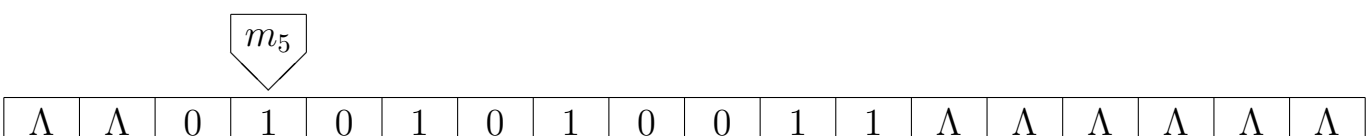
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



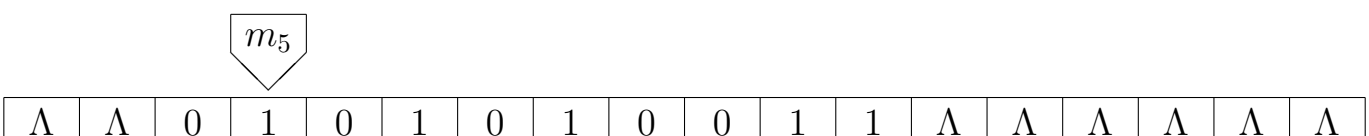
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



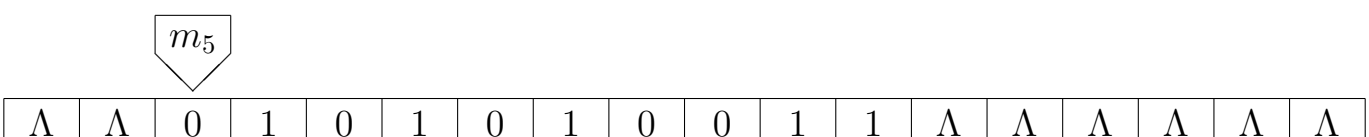
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



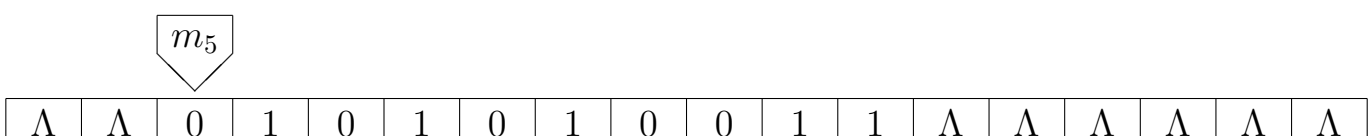
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



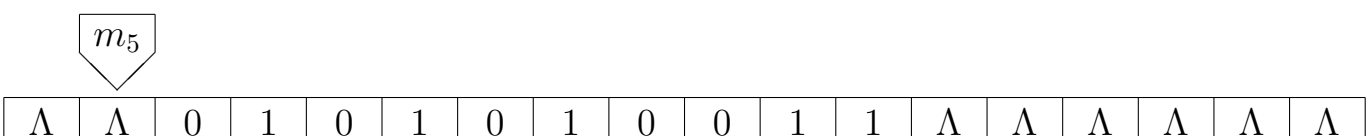
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



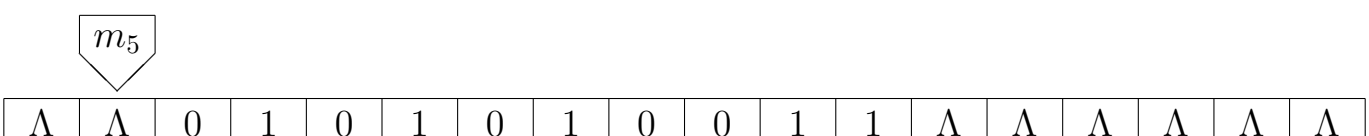
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



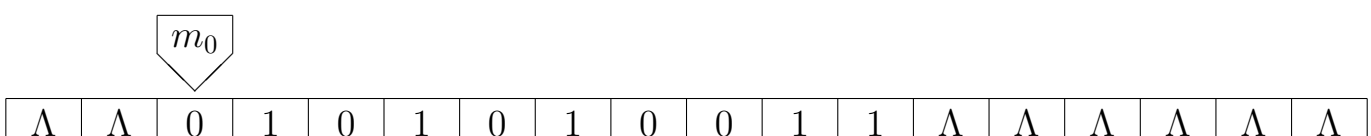
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



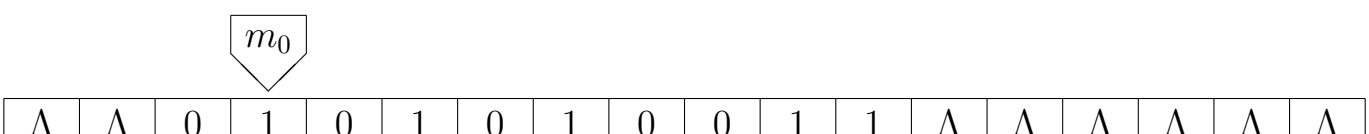
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

Λ	Λ	0	1	0	1	0	1	0	0	1	1	Λ	Λ	Λ	Λ	Λ	Λ

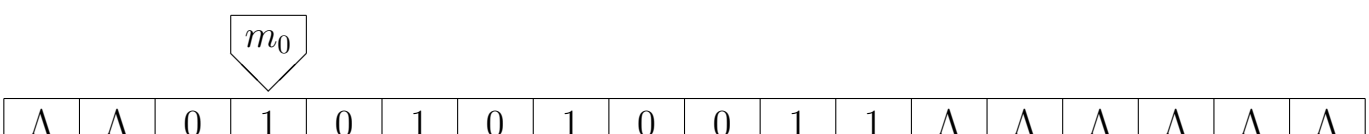
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



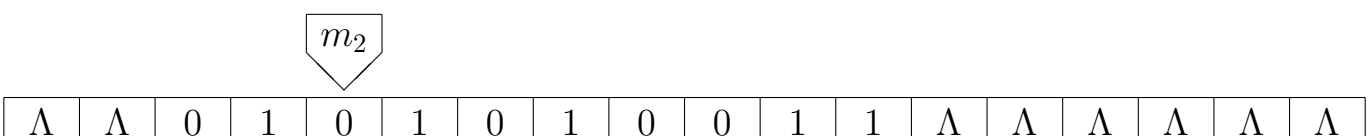
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



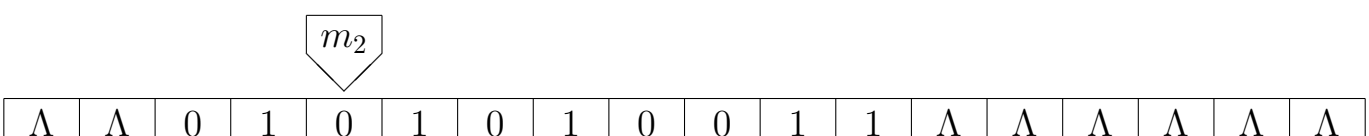
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



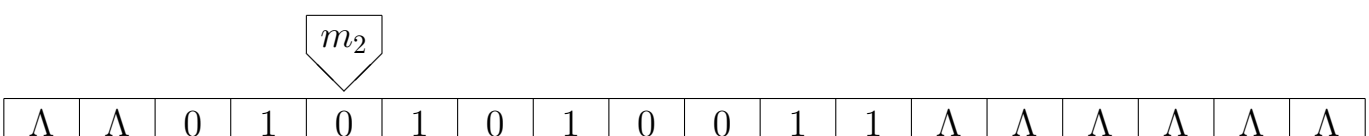
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



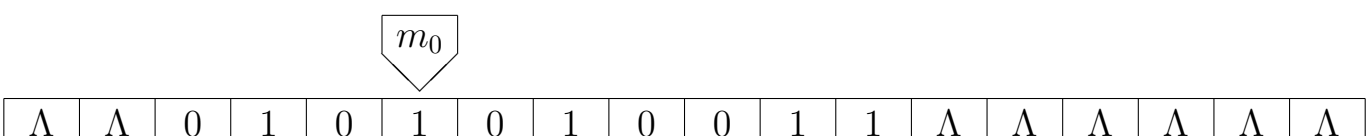
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



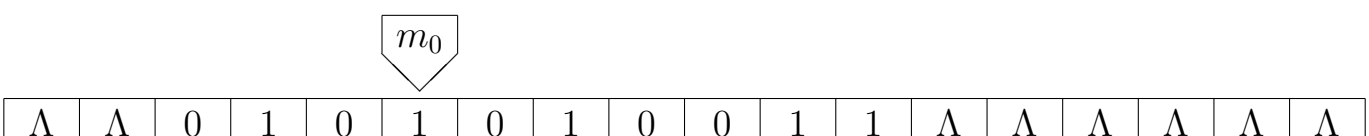
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



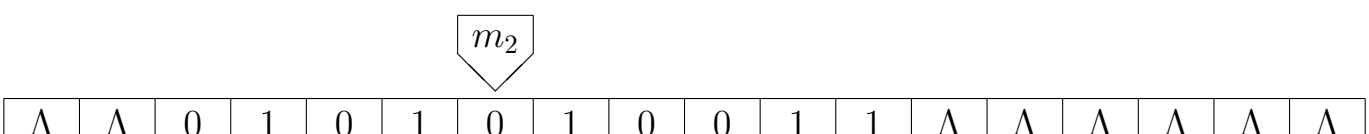
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



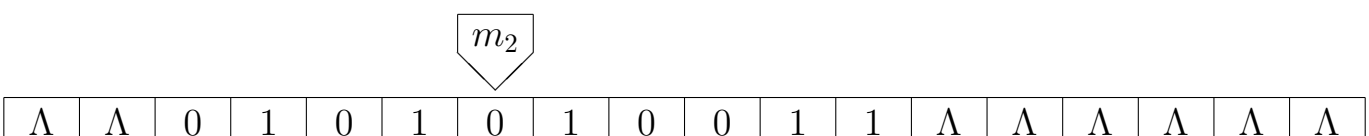
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



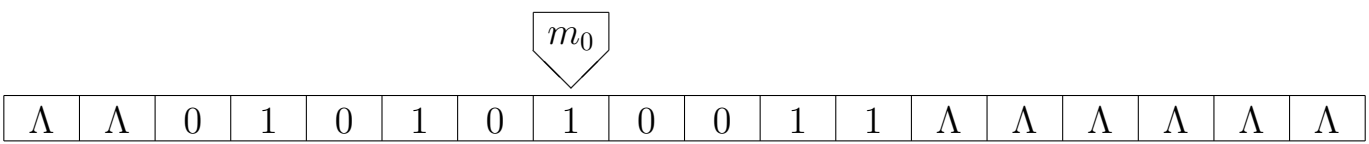
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



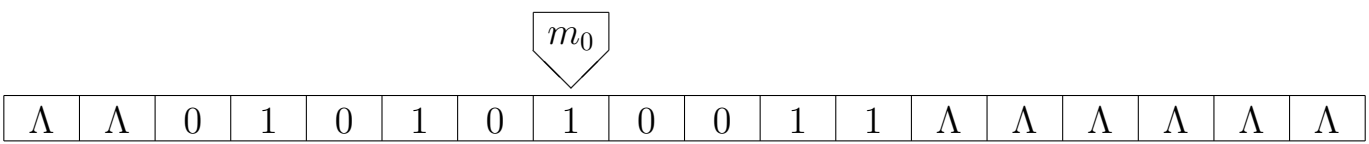
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



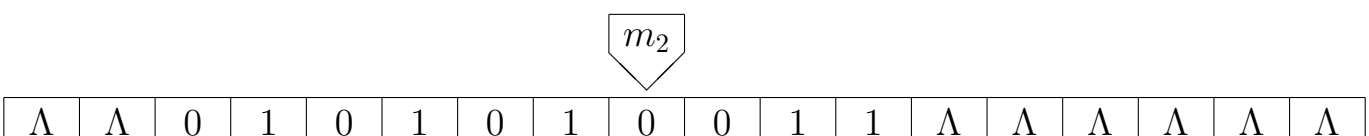
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



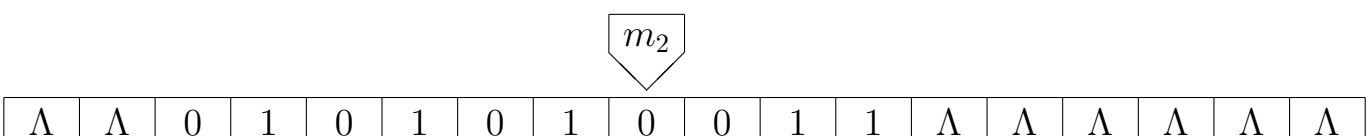
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



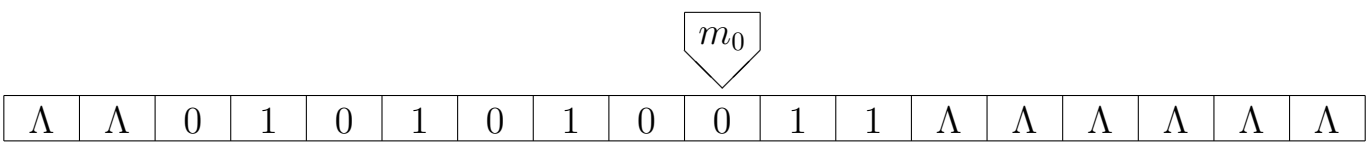
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



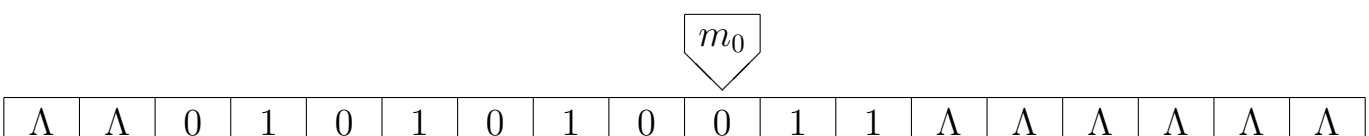
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



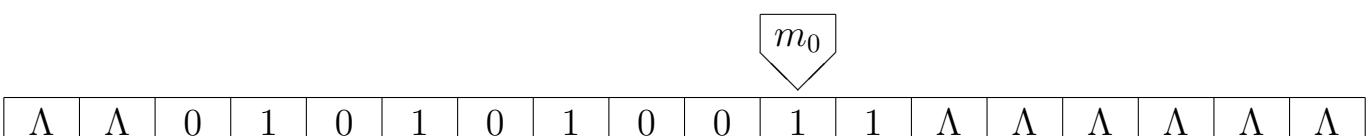
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



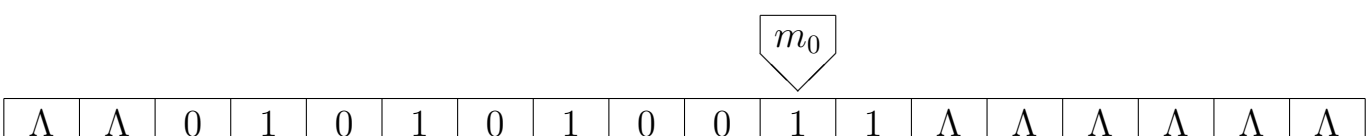
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



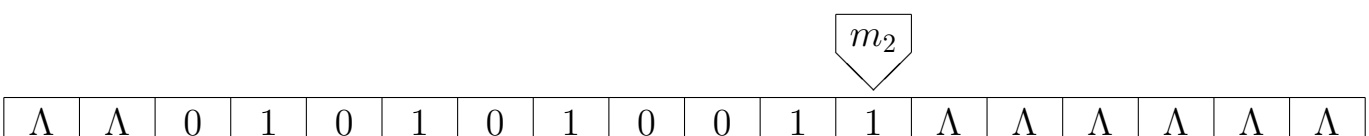
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



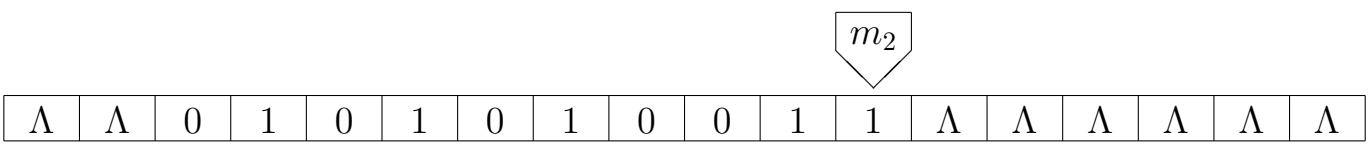
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



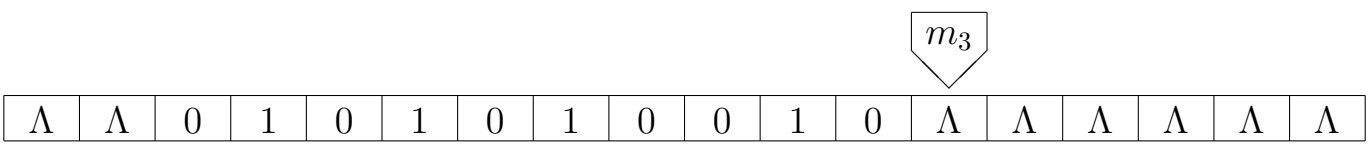
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



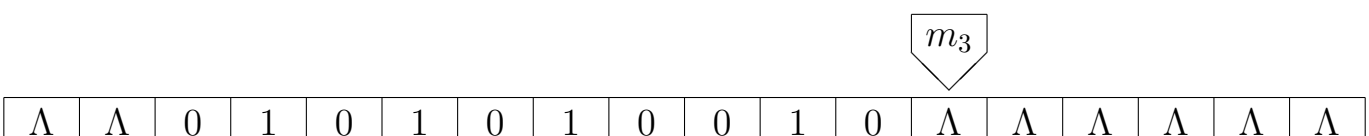
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



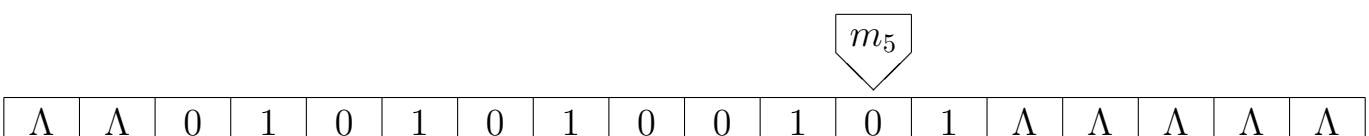
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



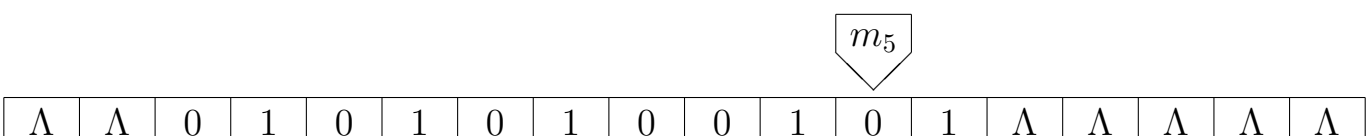
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



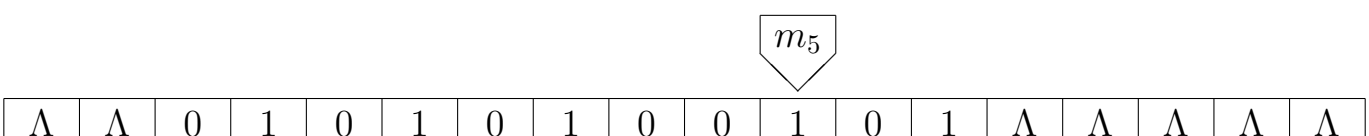
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



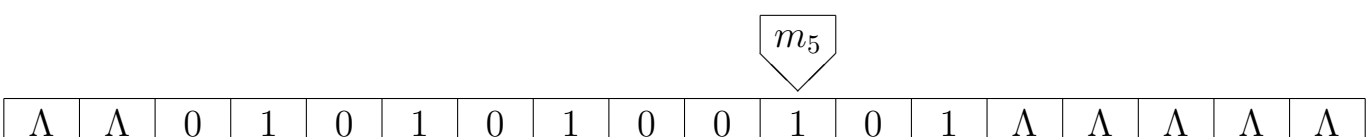
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



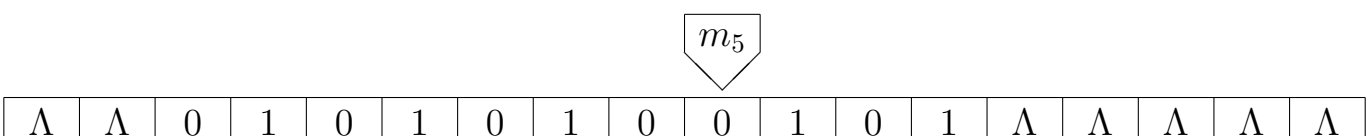
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



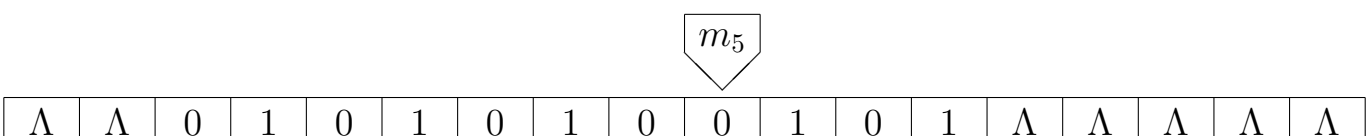
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



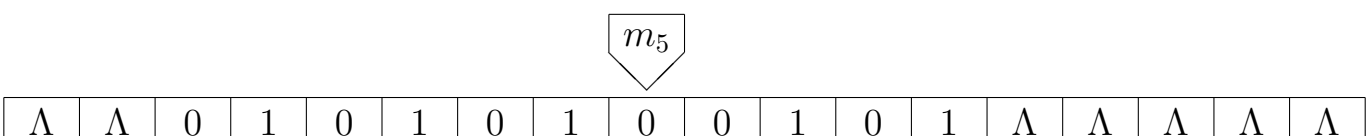
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



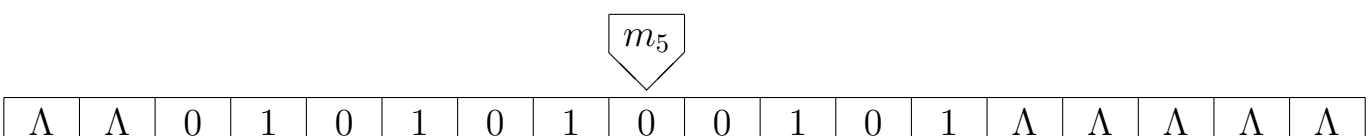
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



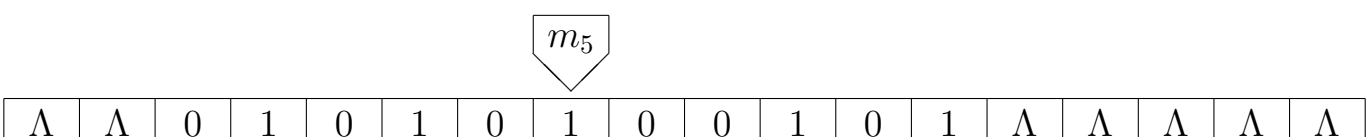
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



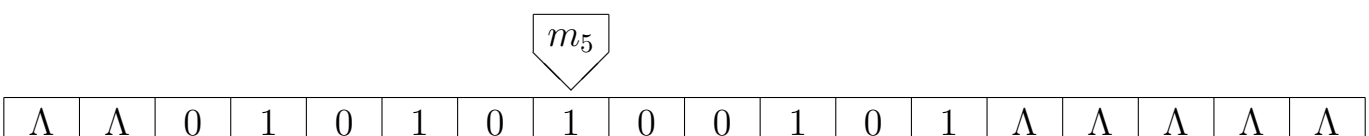
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



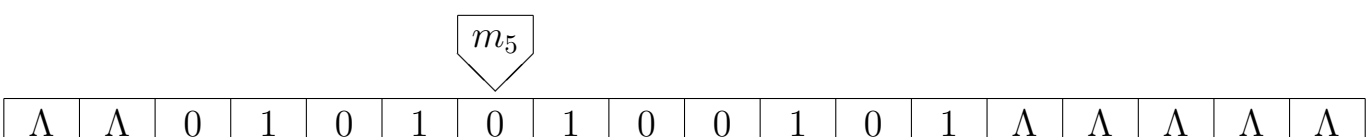
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



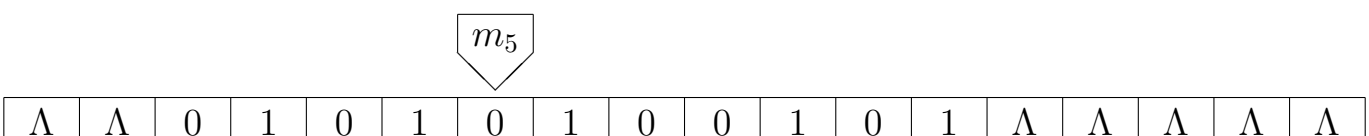
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



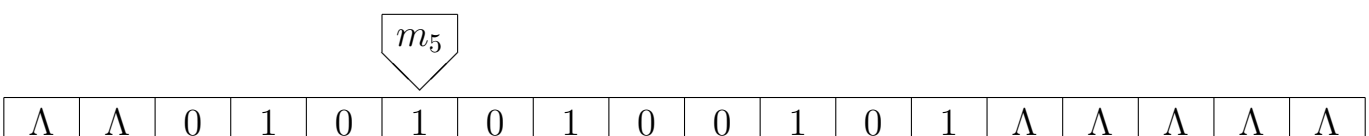
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



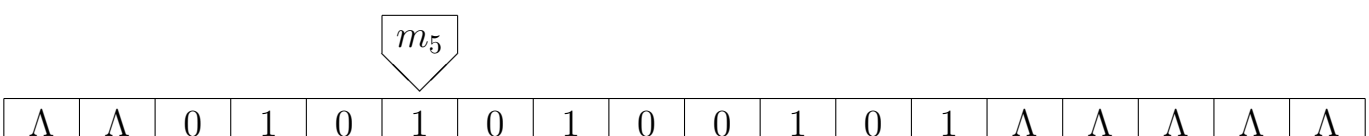
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



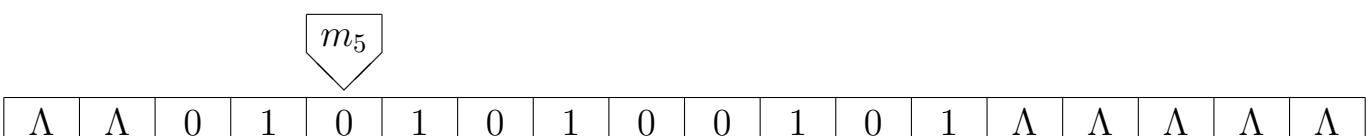
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



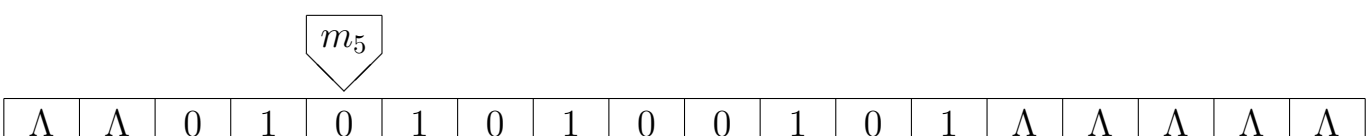
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



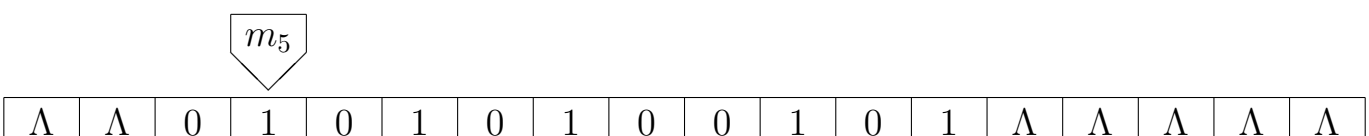
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



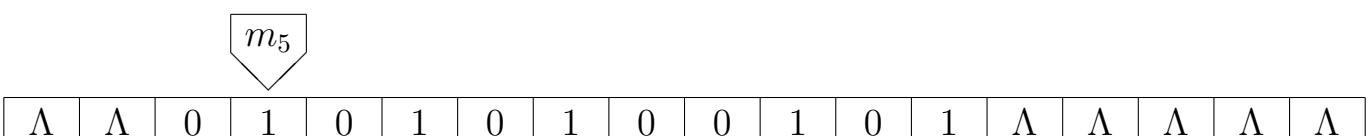
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



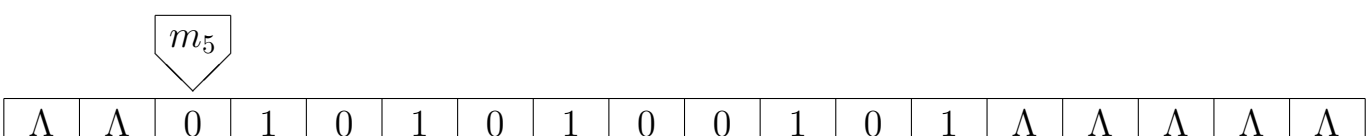
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



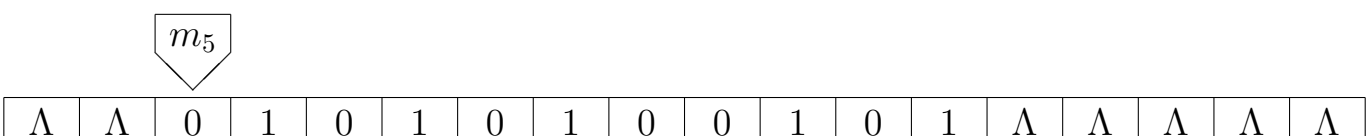
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



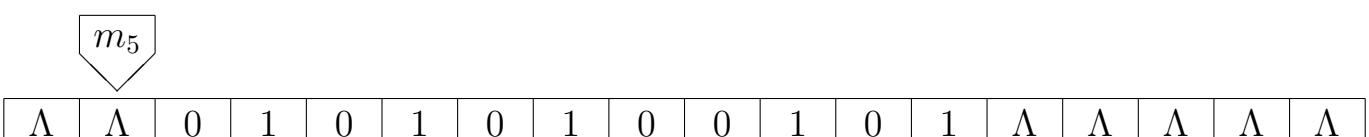
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



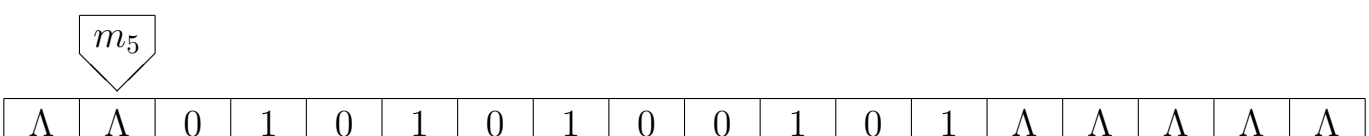
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



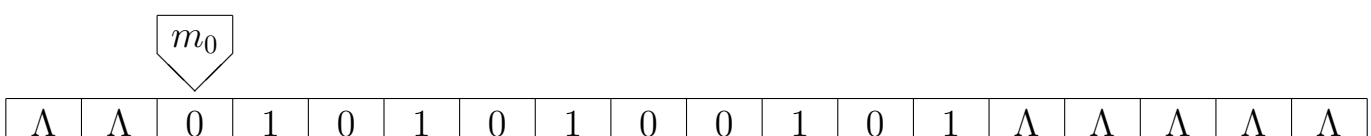
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

Λ	Λ	0	1	0	1	0	1	0	0	1	0	1	Λ	Λ	Λ	Λ	Λ

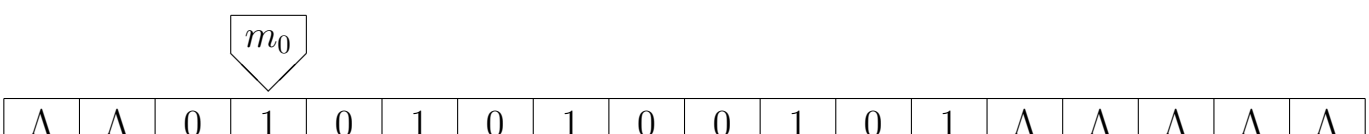
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

Λ	Λ	0	1	0	1	0	1	0	0	1	0	1	Λ	Λ	Λ	Λ	Λ

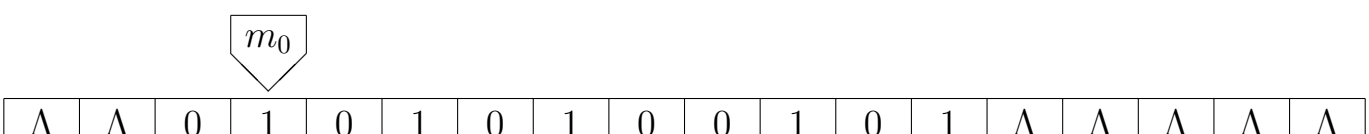
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



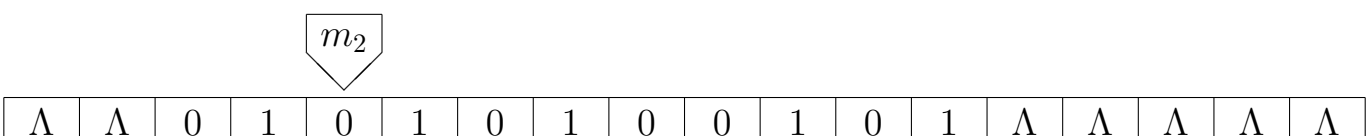
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



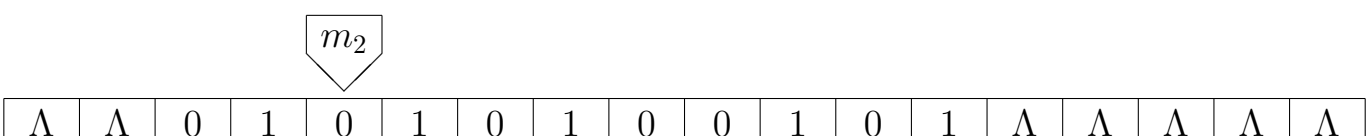
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



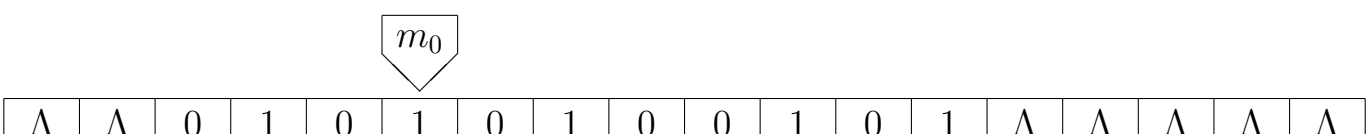
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



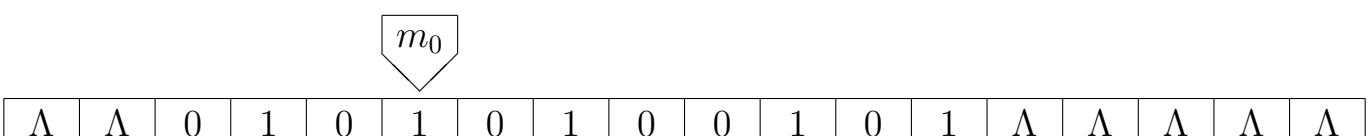
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



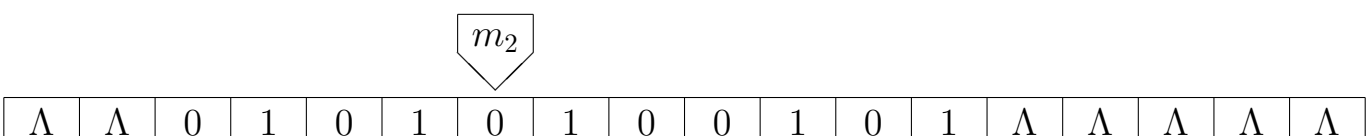
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



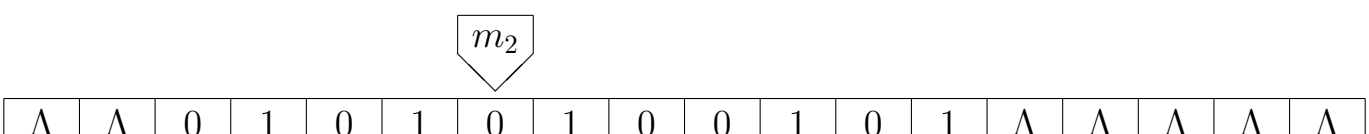
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



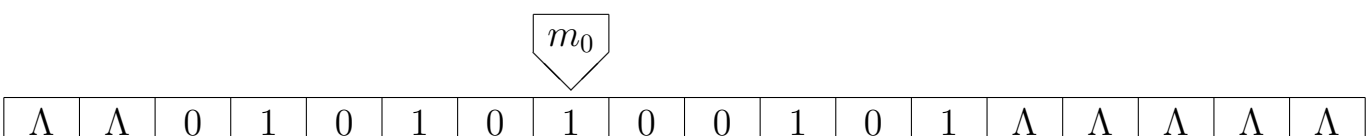
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



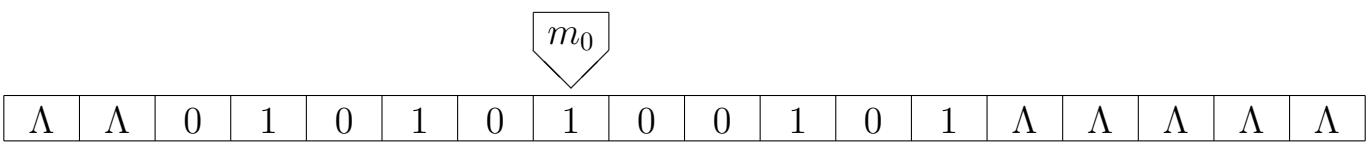
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



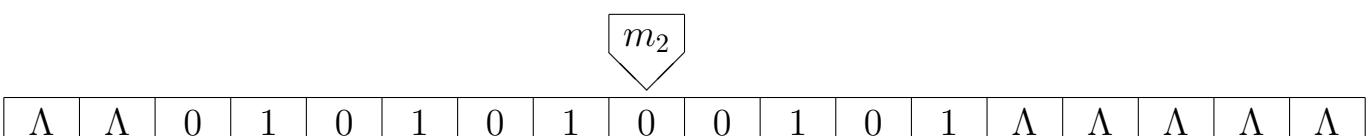
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



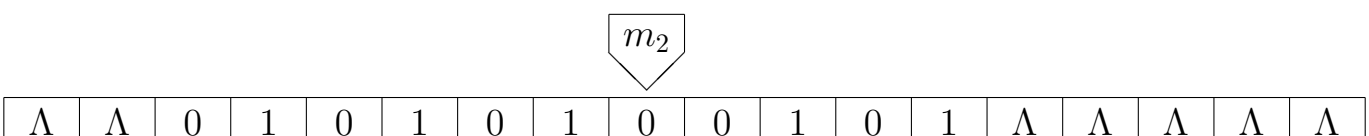
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



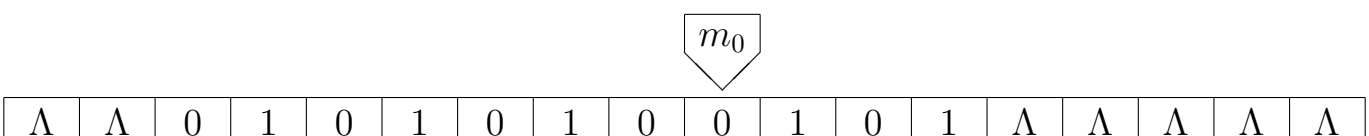
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



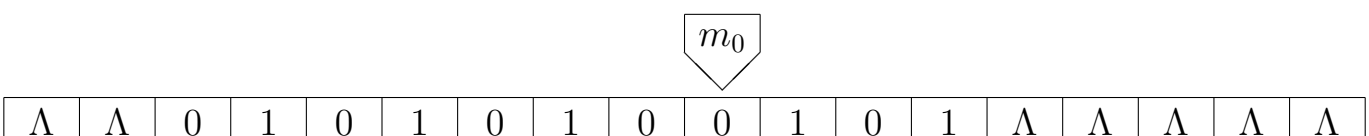
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



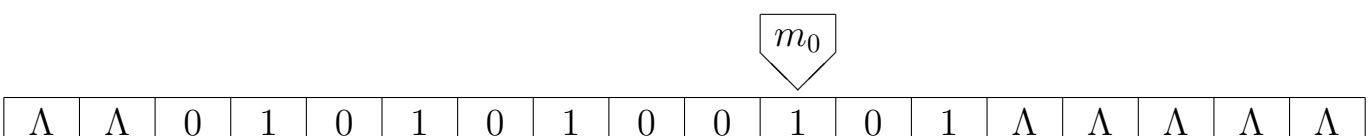
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



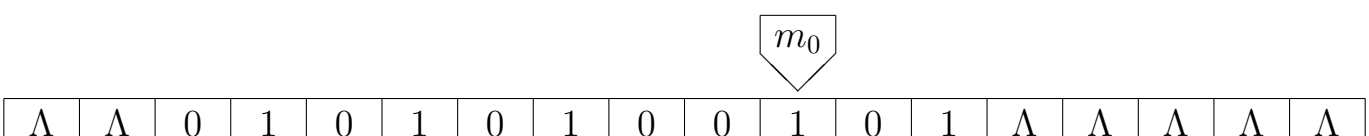
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



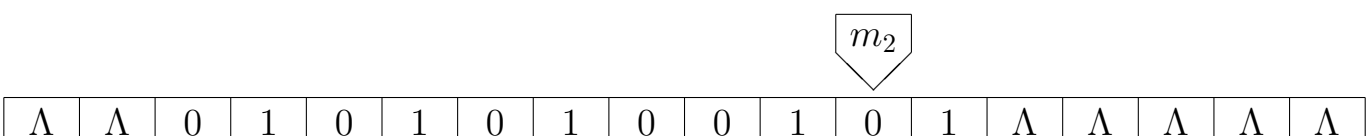
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



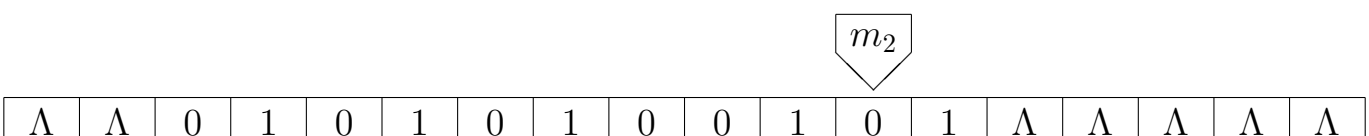
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



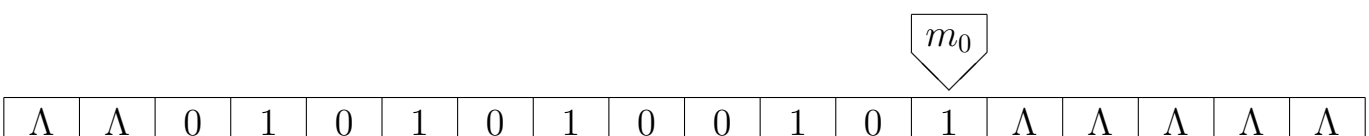
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



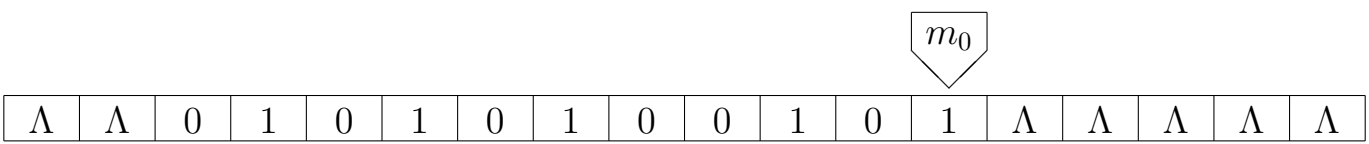
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



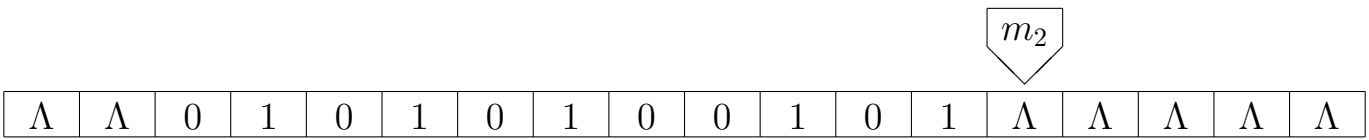
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



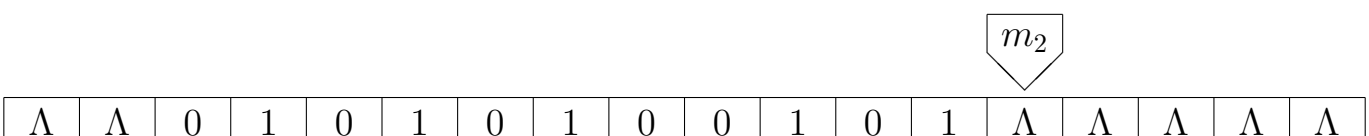
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



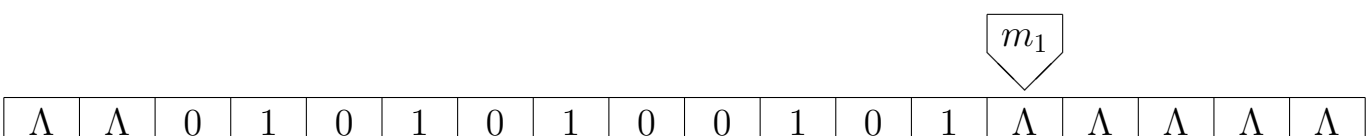
Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			$(\Lambda; m_1)$	$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

													m_2				
Λ	Λ	0	1	0	1	0	1	0	0	1	0	1	Λ	Λ	Λ	Λ	Λ

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			$(\Lambda; m_1)$	$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			$(\Lambda; m_1)$	$L(1, m_5)$		$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

													m_1					
Λ	Λ	0	1	0	1	0	1	0	0	1	0	1	Λ	Λ	Λ	Λ	Λ

Надо предусмотреть ситуацию, когда последней цифрой входного слова является 0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

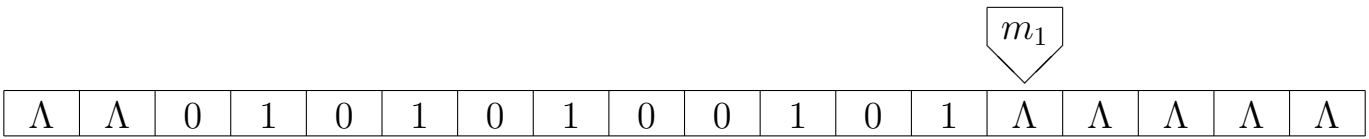
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			$(\Lambda; m_1)$	$L(1, m_5)$	$L(0; m_5)$	$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					

													m_1					
Λ	Λ	0	1	0	1	0	1	0	0	1	0	1	Λ	Λ	Λ	Λ	Λ

Надо предусмотреть ситуацию, когда последней цифрой входного слова является 0.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

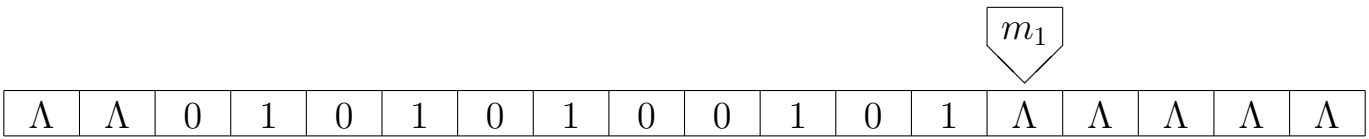
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			$(\Lambda; m_1)$	$L(1, m_5)$	$L(0; m_5)$	$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Надо предусмотреть ситуацию, когда последней цифрой входного слова является 0. Вставим обработку пустого слова.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

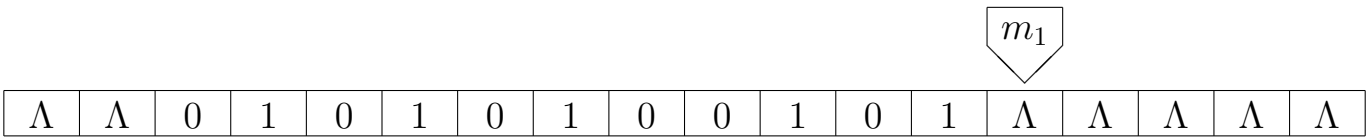
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$(\Lambda; m_1)$		$(\Lambda; m_1)$	$L(1, m_5)$	$L(0; m_5)$	$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Надо предусмотреть ситуацию, когда последней цифрой входного слова является 0. Вставим обработку пустого слова.

Пример 5. *Написать машину Тьюринга, вставляющую 0 между «сдвоенными единицами»:* $11 \to 101$ *во входном слове в алфавите* $\{0;\ 1\}$, *см.* ***пример 2***.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$(\Lambda; m_1)$		$(\Lambda; m_1)$	$L(1, m_5)$	$L(0; m_5)$	$R(\Lambda; m_0)$					
0	$R(0, m_0)$		$R(0, m_0)$	$R(1, m_4)$	$R(0, m_4)$	$L(0, m_5)$					
1	$R(1, m_2)$		$R(0, m_3)$	$R(1, m_3)$	$R(0, m_3)$	$L(1, m_5)$					



Надо предусмотреть ситуацию, когда последней цифрой входного слова является 0. Вставим обработку пустого слова.

Вернемся к лекции или рассмотрим **пример 6?**

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										

Применим ту же идею, что в **примере 3**:

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										

Λ	1	1	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										

m_0

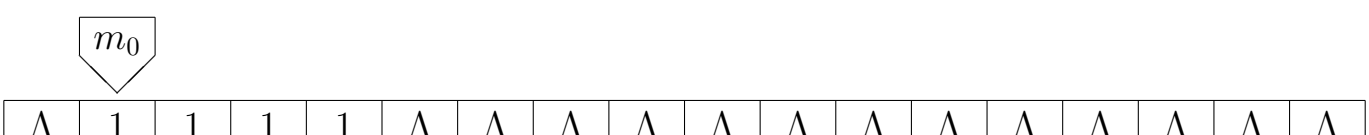
Λ	1	1	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$									
c										

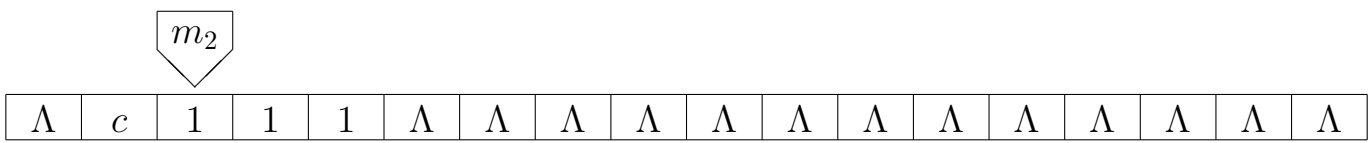


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$									
c										

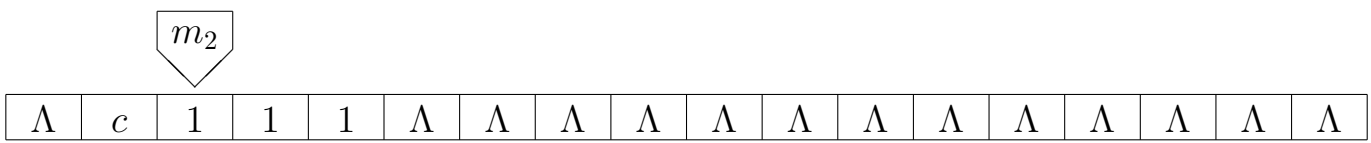


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

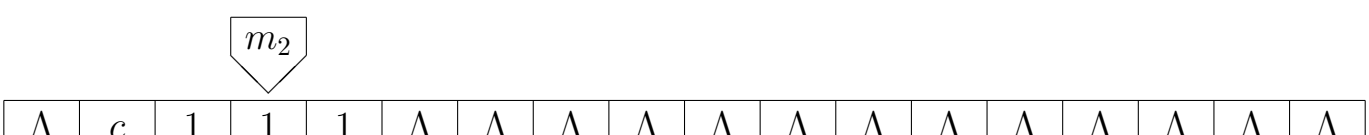


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

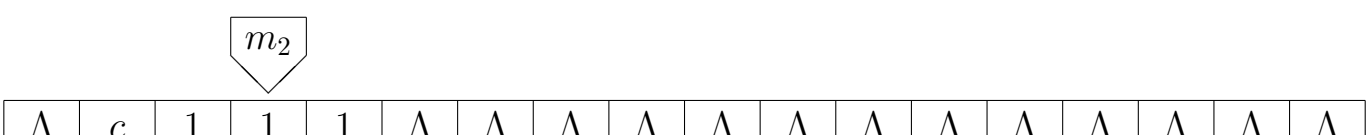


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

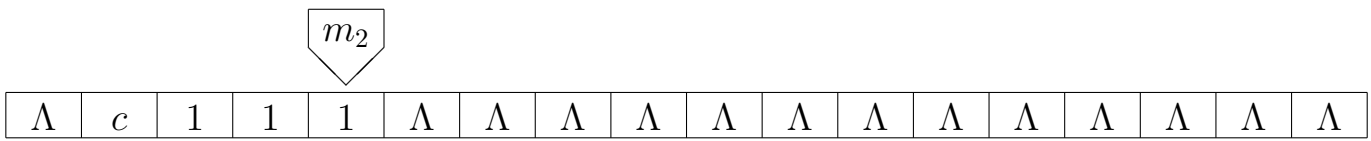


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

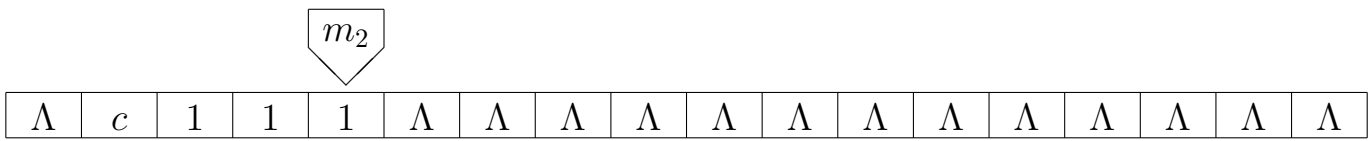


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

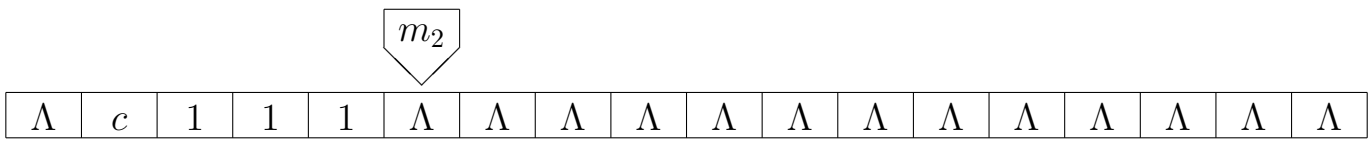


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1	$R(c, m_2)$	$R(1, m_2)$								
c										

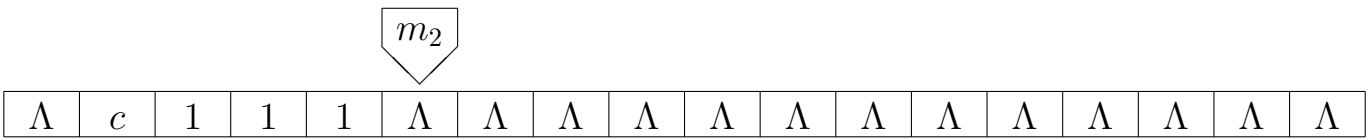


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$								
c										

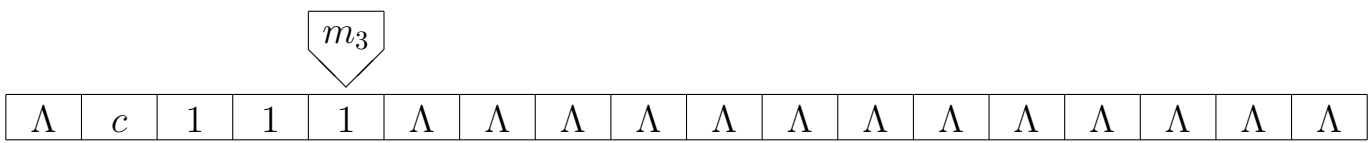


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$								
c										

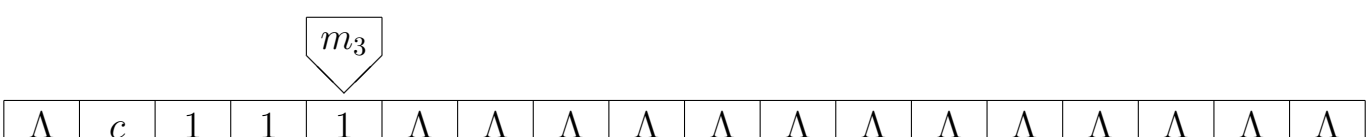


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$							
c										

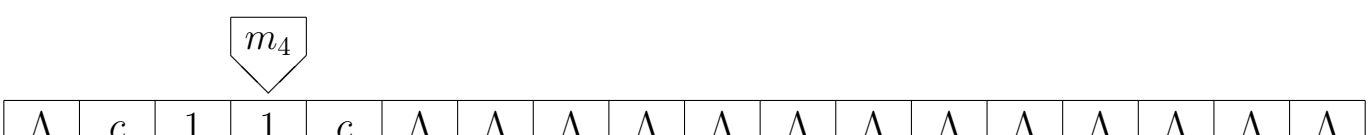


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$							
c										

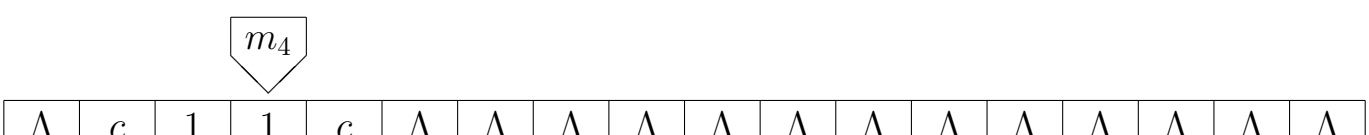


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c										



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c										

Λ	c	1	1	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c										

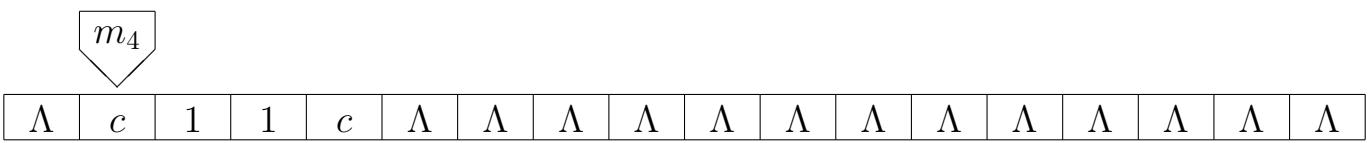
Λ	c	1	1	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c										



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c				$R\,(c, m_0)$						

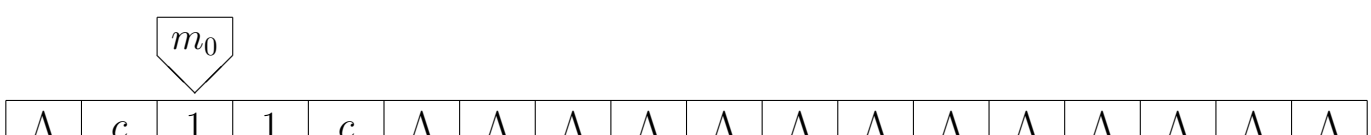
Λ	c	1	1	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c				$R\,(c, m_0)$						



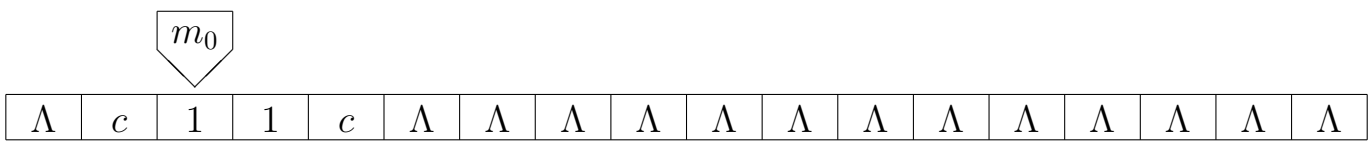
Применим ту же идею, что в **примере 3**:

«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L(\Lambda, m_3)$								
0										
1	$R(c, m_2)$	$R(1, m_2)$	$L(c, m_4)$	$L(1, m_4)$						
c				$R(c, m_0)$						

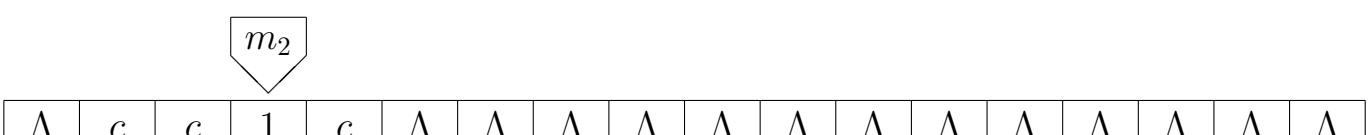


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L(\Lambda, m_3)$								
0										
1	$R(c, m_2)$	$R(1, m_2)$	$L(c, m_4)$	$L(1, m_4)$						
c				$R(c, m_0)$						

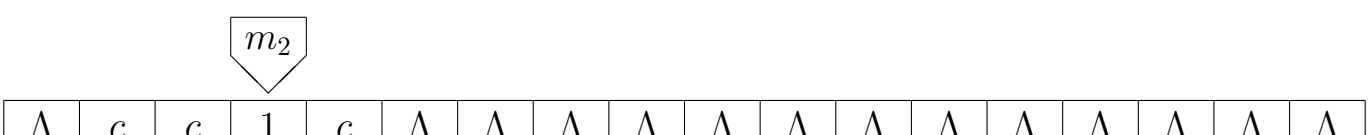


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c				$R\,(c, m_0)$						

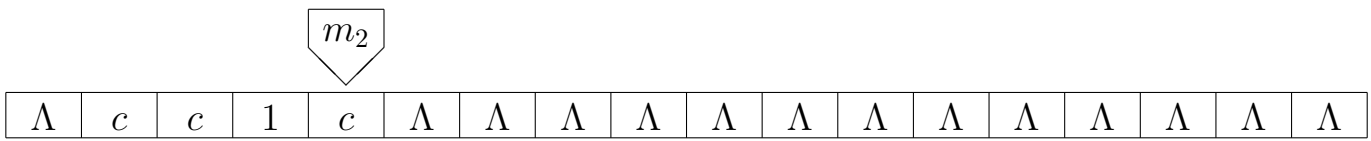


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c				$R\,(c, m_0)$						

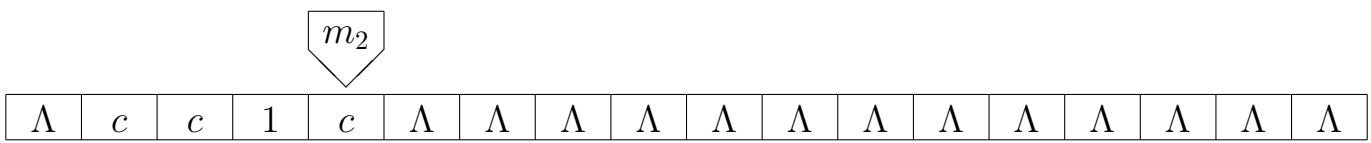


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						



Применим ту же идею, что в **примере 3**:

«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						

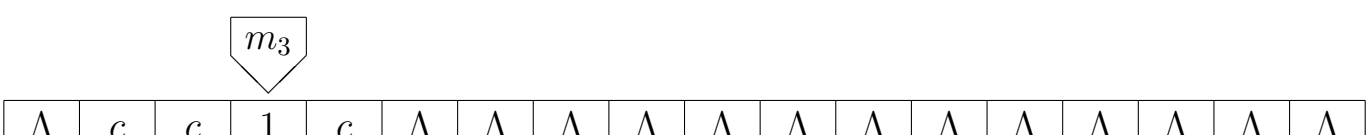
Λ	c	c	1	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						

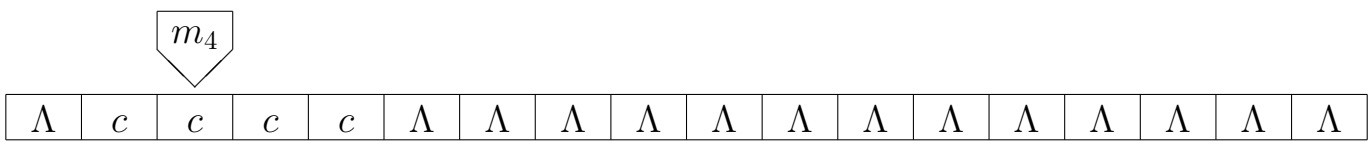
Λ	c	c	c	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c		$L\,(c, m_3)$		$R\,(c, m_0)$						

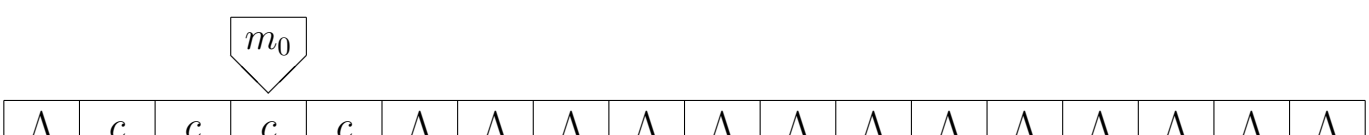
Λ	c	c	c	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$						



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$						

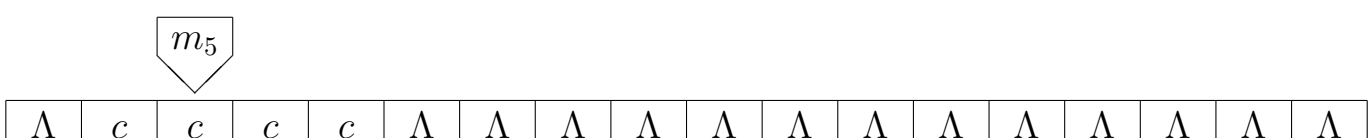
Λ	c	c	c	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					

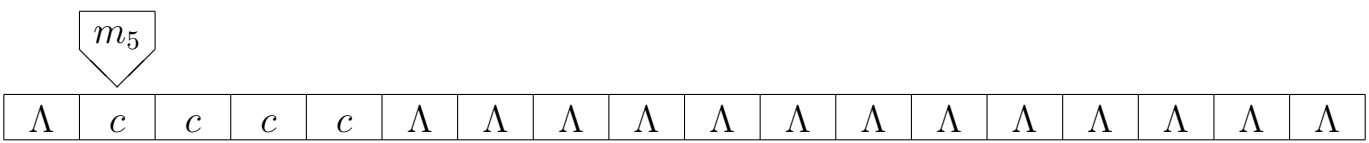


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					

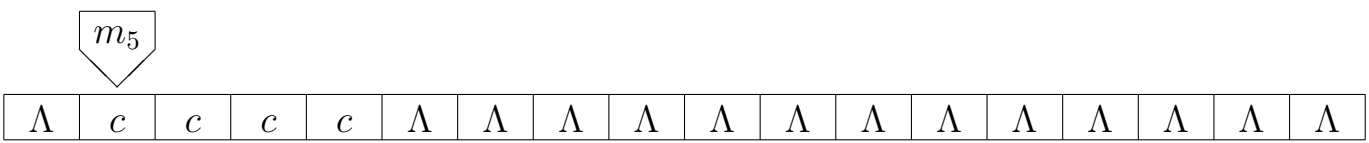


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$								
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					

m_5

Λ	c	c	c	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					

m_5

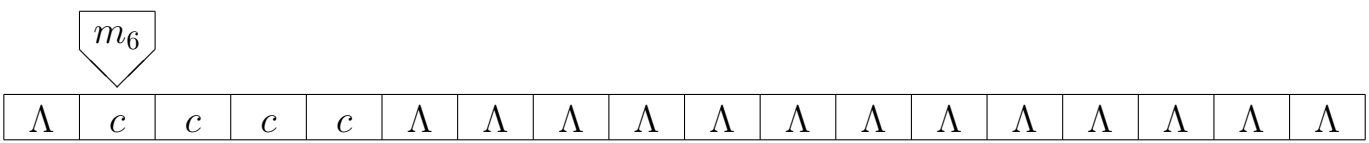
Λ	c	c	c	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$					

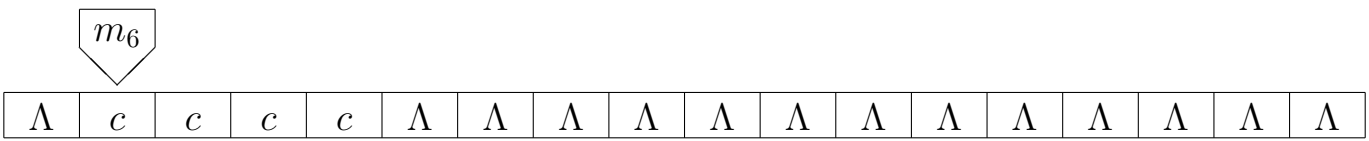


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

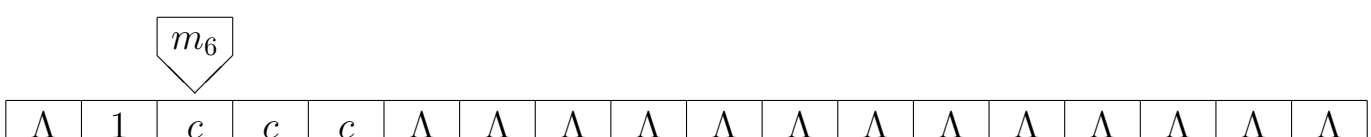


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

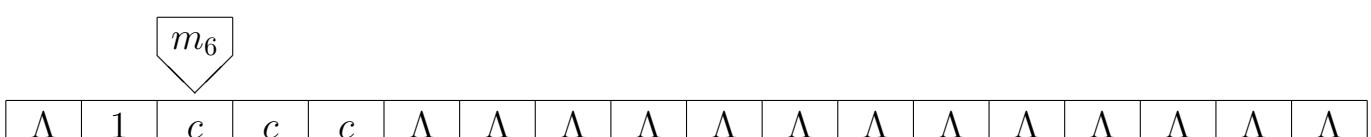


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

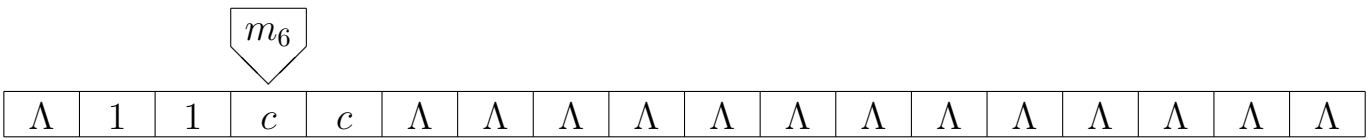


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

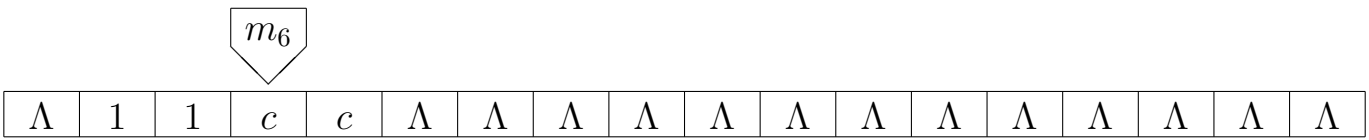


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

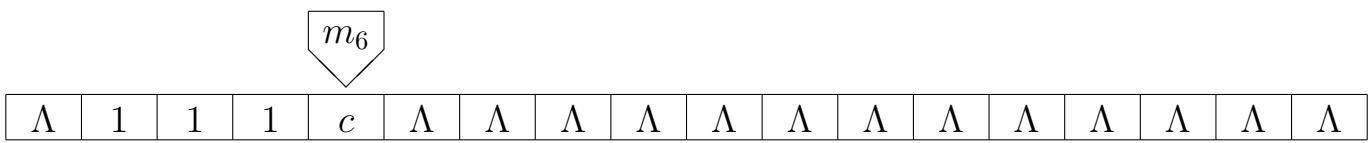


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L(\Lambda, m_3)$			$R(\Lambda, m_6)$					
0										
1	$R(c, m_2)$	$R(1, m_2)$	$L(c, m_4)$	$L(1, m_4)$						
c	$L(c, m_5)$	$L(c, m_3)$		$R(c, m_0)$	$L(c, m_5)$	$R(1, m_6)$				

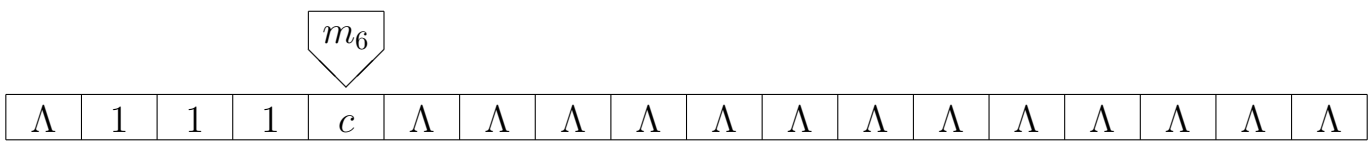


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L(\Lambda, m_3)$			$R(\Lambda, m_6)$					
0										
1	$R(c, m_2)$	$R(1, m_2)$	$L(c, m_4)$	$L(1, m_4)$						
c	$L(c, m_5)$	$L(c, m_3)$		$R(c, m_0)$	$L(c, m_5)$	$R(1, m_6)$				

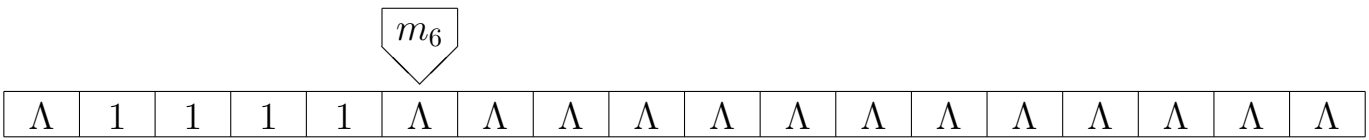


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$					
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

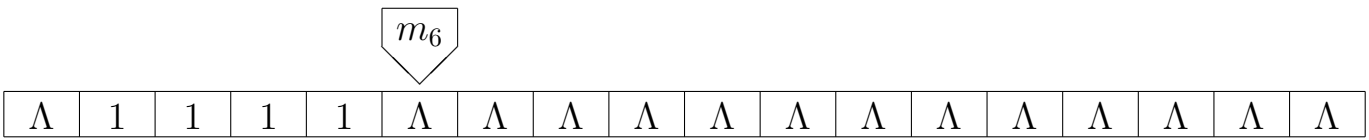


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

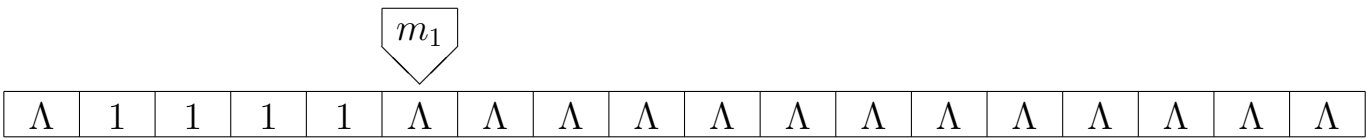


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

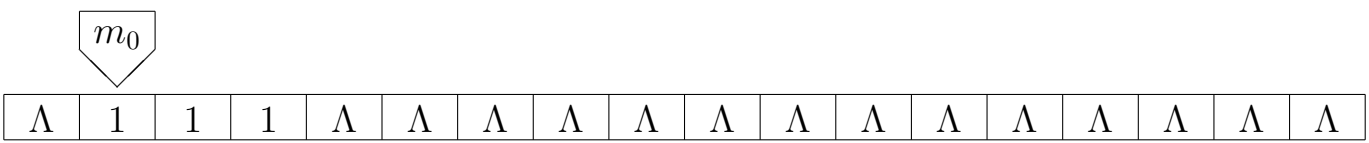
Λ	1	1	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

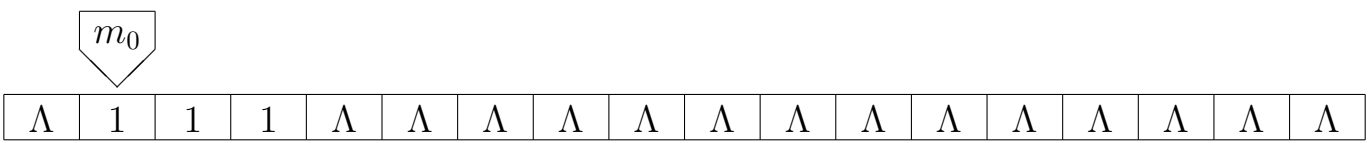


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

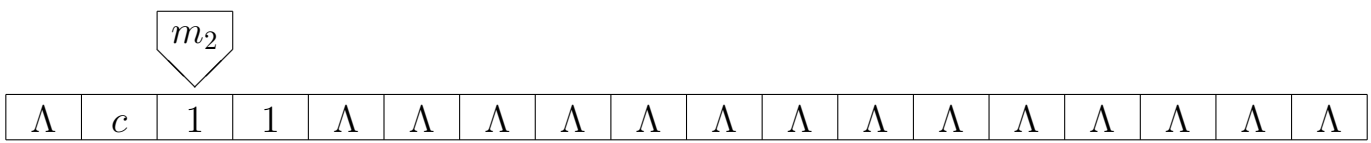


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

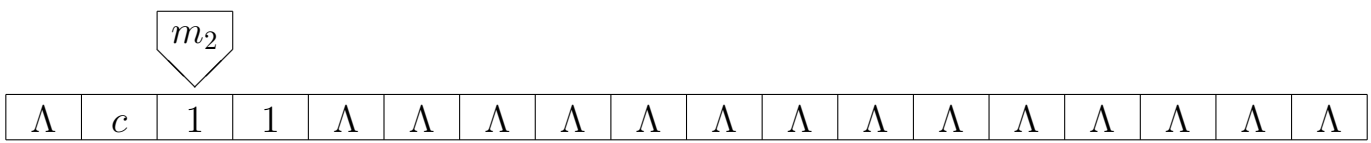


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

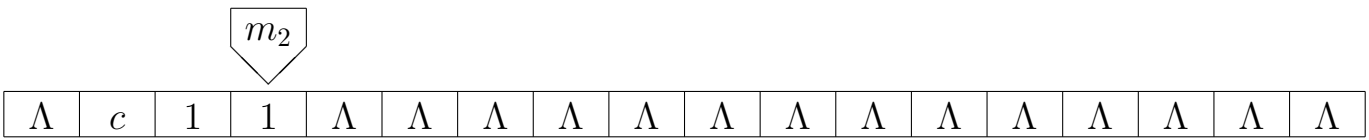


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

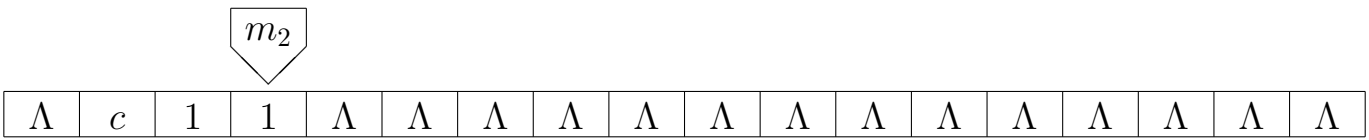


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

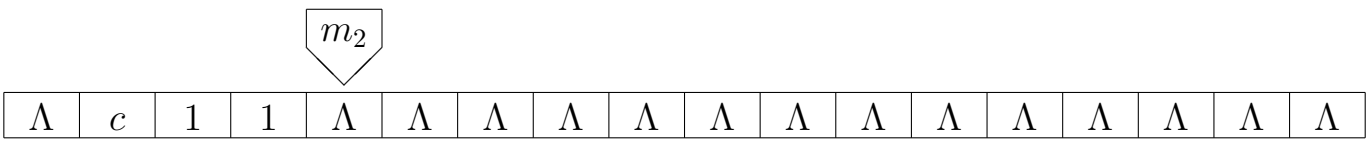


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

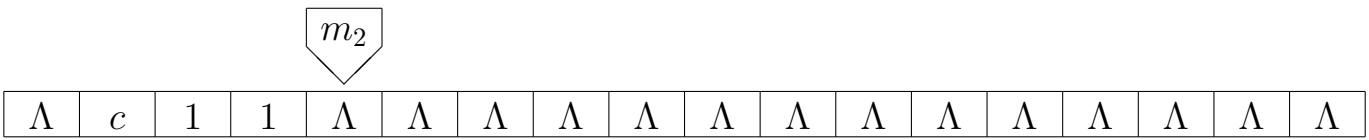


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

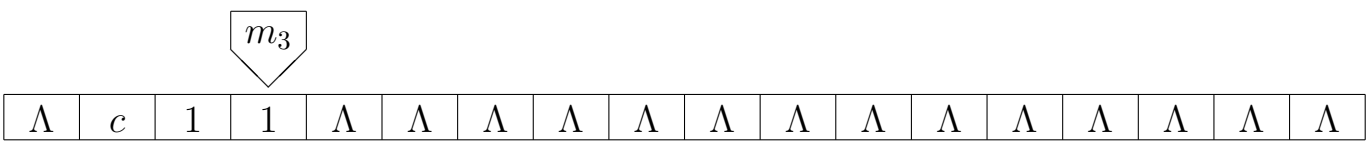


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

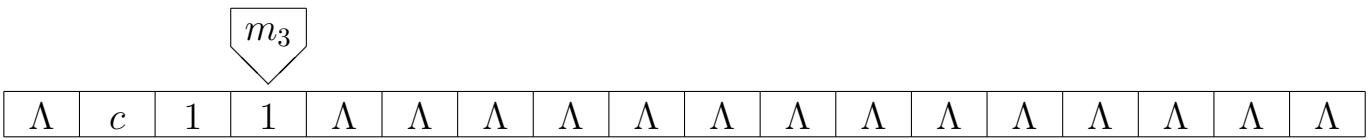


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

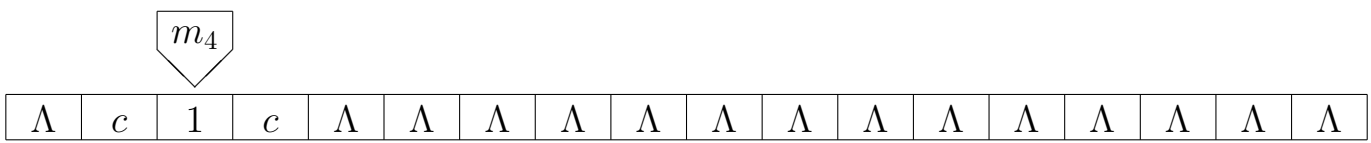


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

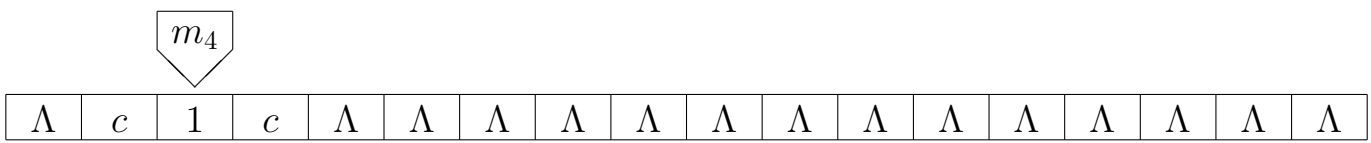


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

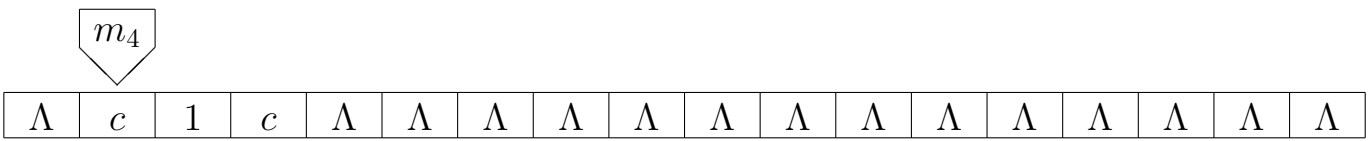


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

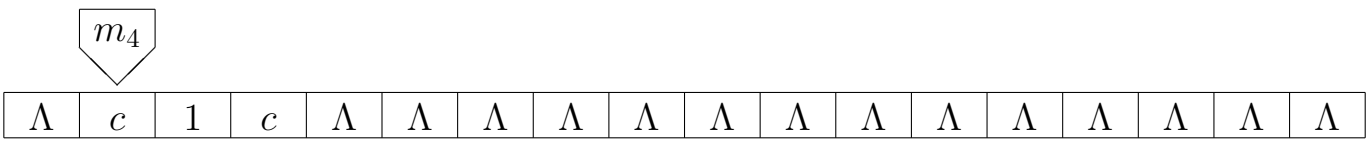


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

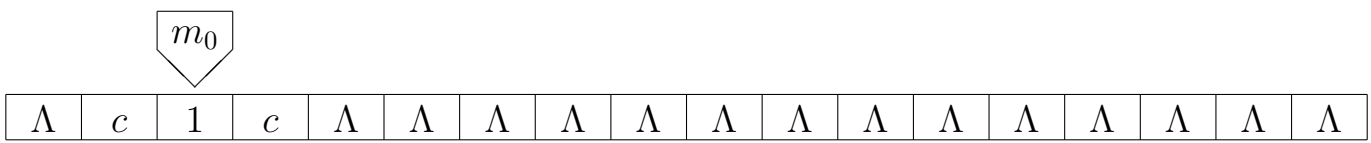


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

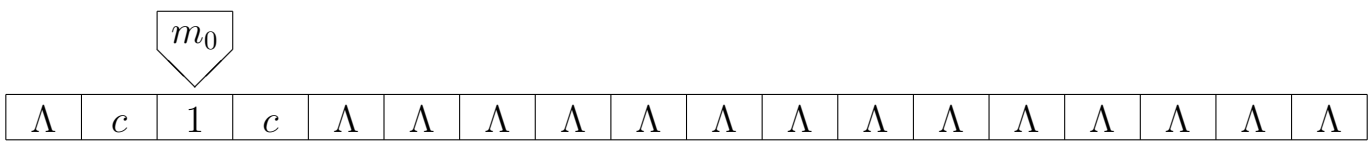


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

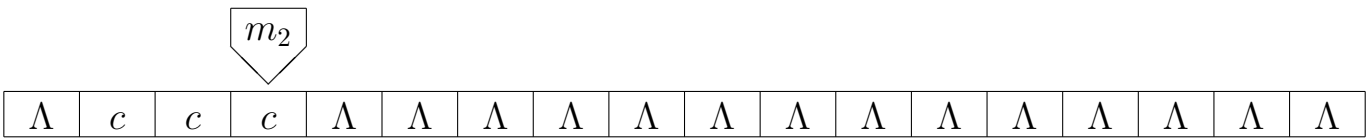


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

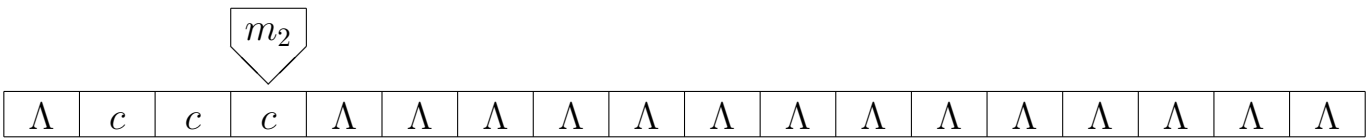


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

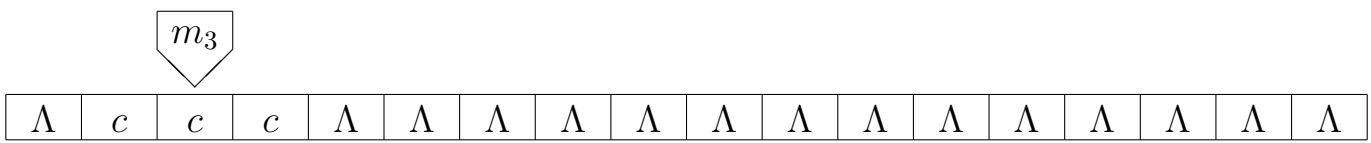


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$		$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

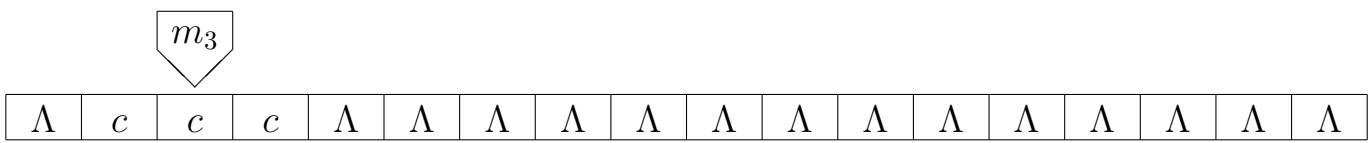


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

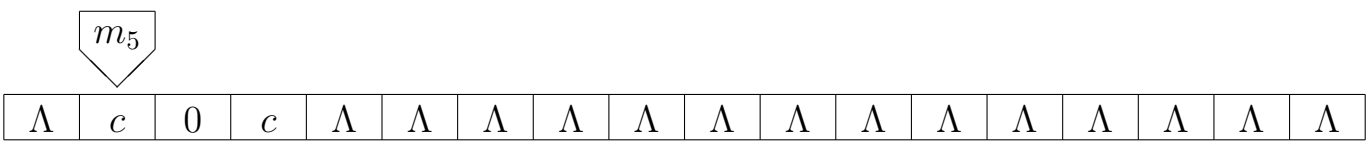


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

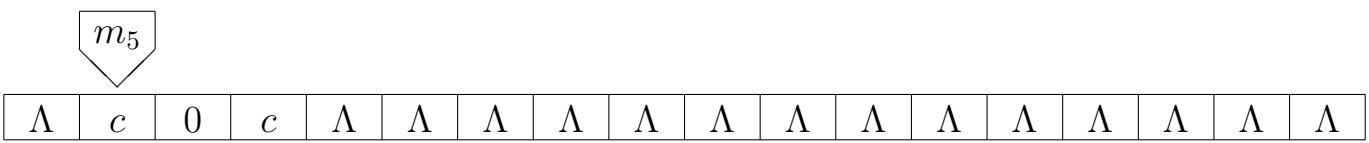


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

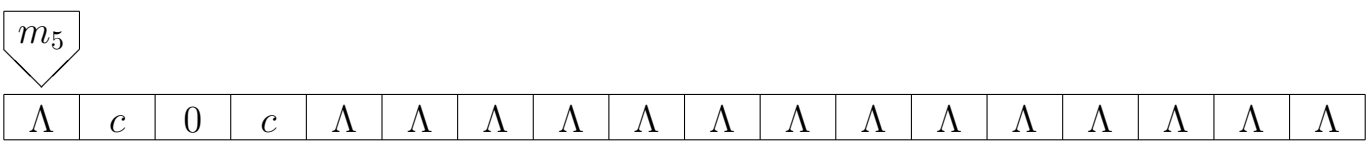
Λ	c	0	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

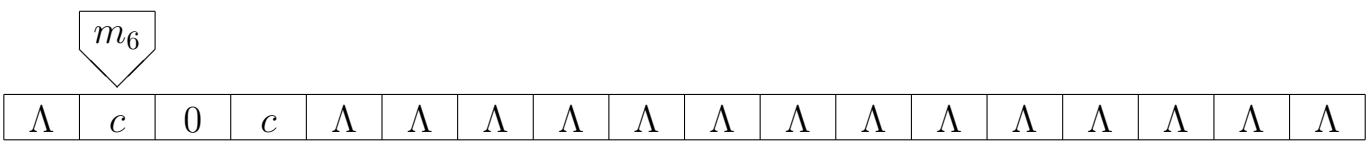


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

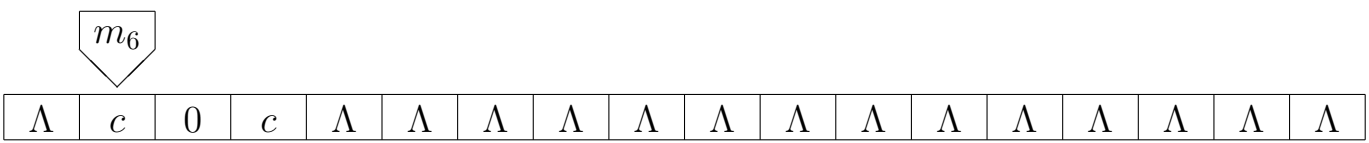


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

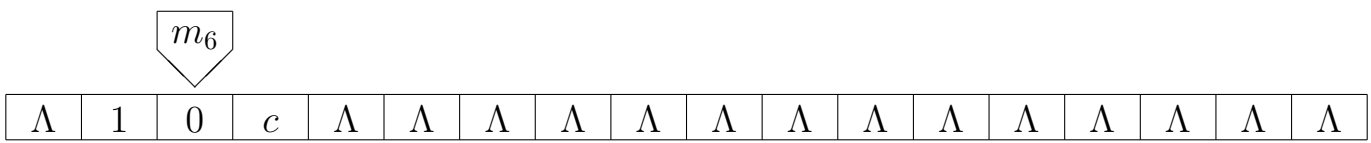


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0										
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

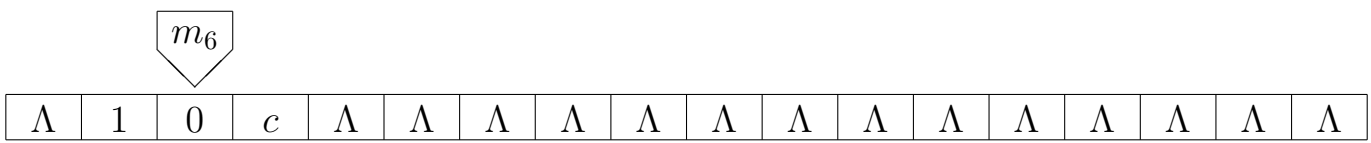


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

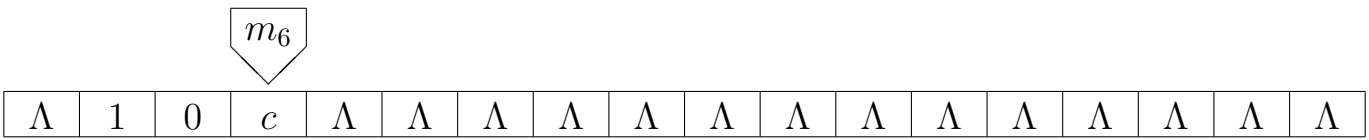


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

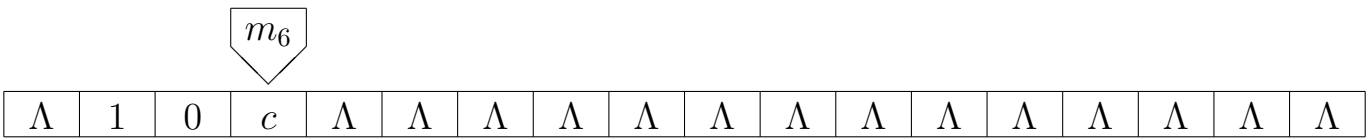


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

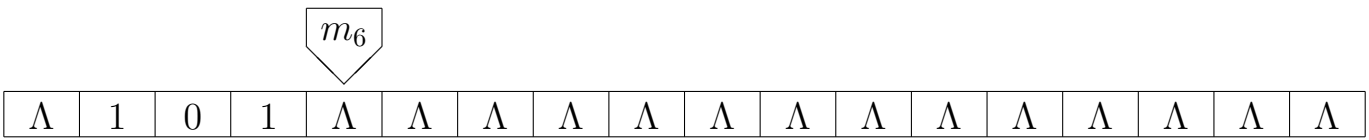


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

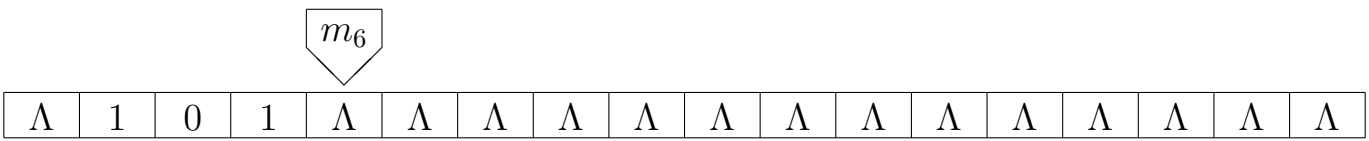


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

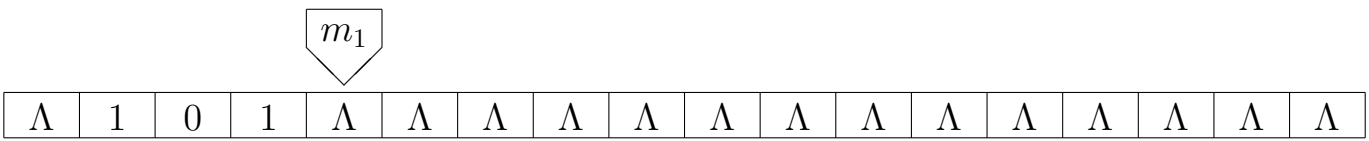


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

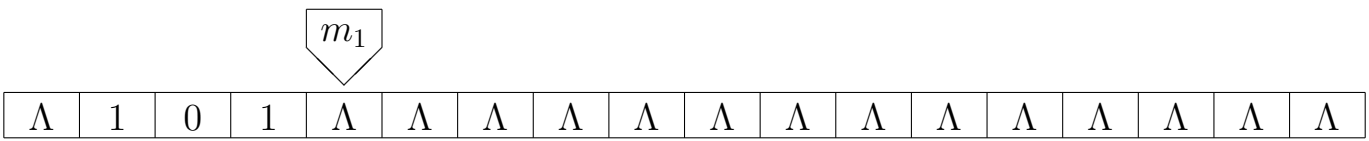


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

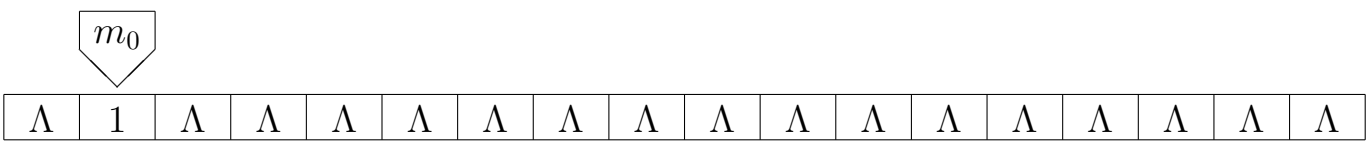
Λ	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

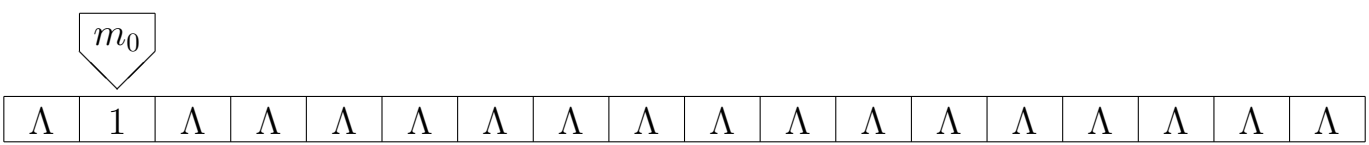


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

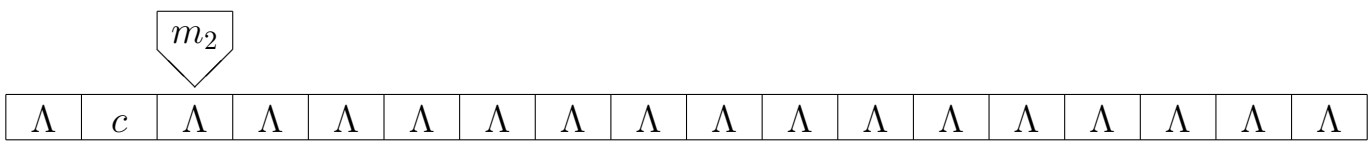


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

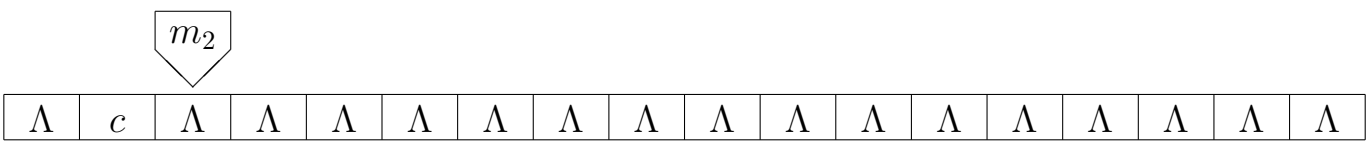


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

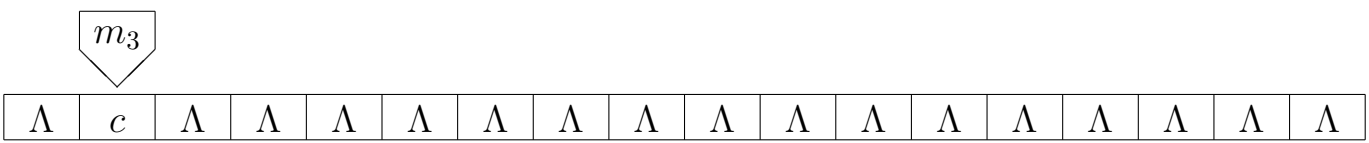


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

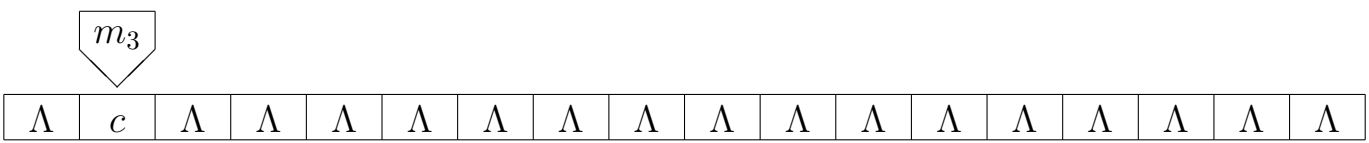


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

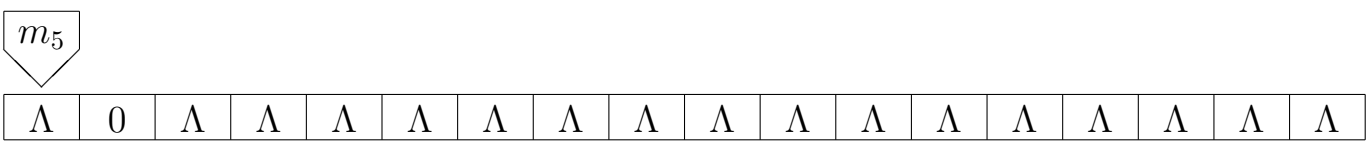


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

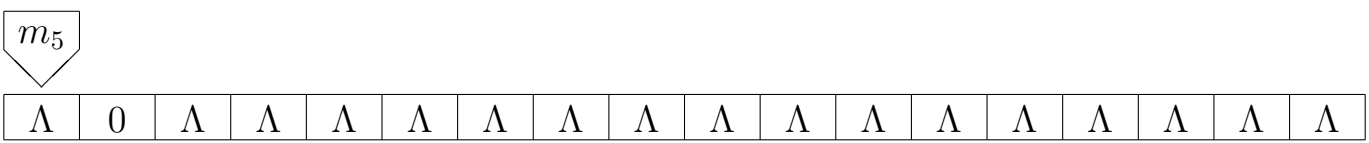


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

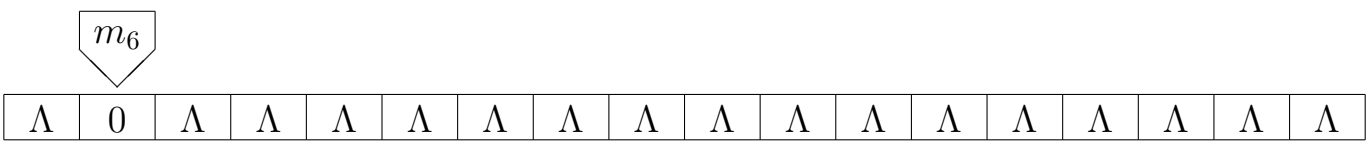


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

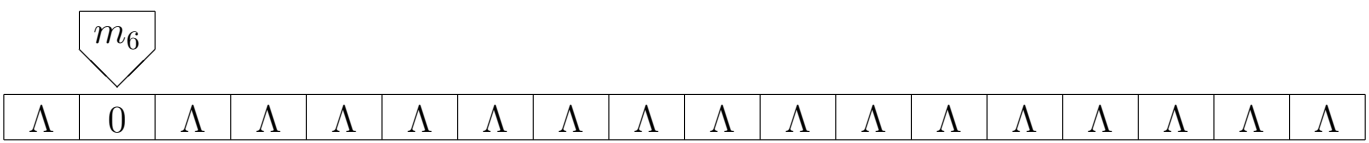


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

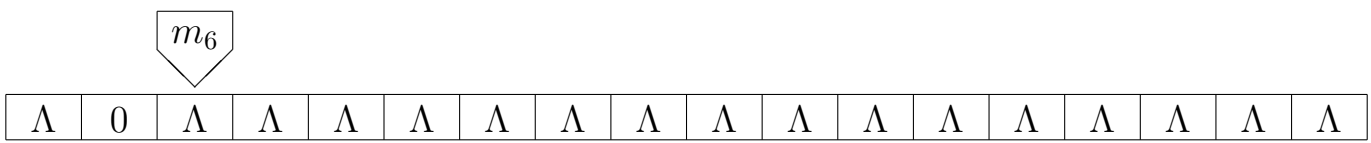


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

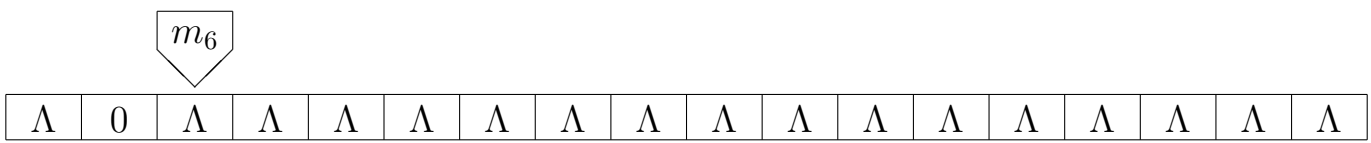


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

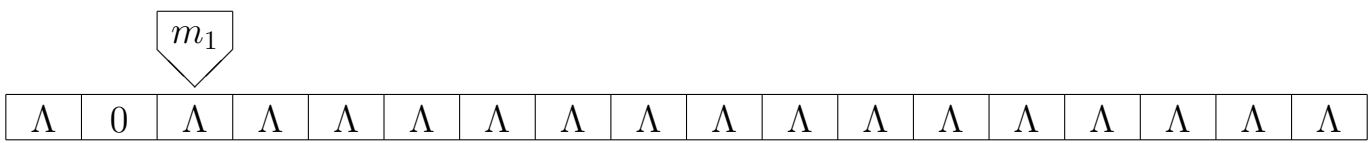


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

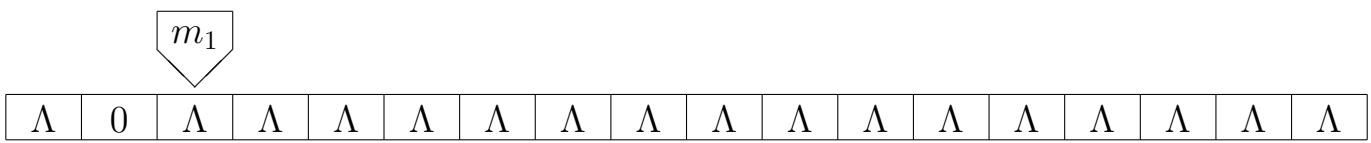


Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				



Применим ту же идею, что в **примере 3**:
«Перекодовываем» единицы в c на левом и правом концах слова, и в зависимости от состояния машины к моменту встречи «левых» c и «правых» c будем либо «расколдовывать» слово либо перед «расколдовыванием» заменим центральную 1 на 0.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

В данном случае нет необходимости проверять действие полного алгоритма на тех входных словах, на которых мы строили его, поскольку

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

В данном случае нет необходимости проверять действие полного алгоритма на тех входных словах, на которых мы строили его, поскольку команды неупорядочены и их выполнение обусловлено только левой частью команды $(a, m_i)Q(b, mj)$, где $Q \in \{L, R\}$.

Пример 6. *Напишите машину Тьюринга, не меняющее слово четной длины в алфавите* $\{1\}$*, а в слове нечетной длины в алфавите* $\{1\}$ *заменяющего центральную единицу на 0 (см.* ***прим.3****).*

Решение.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$L\,(\Lambda, m_3)$			$R\,(\Lambda, m_6)$	(Λ, m_1)				
0						$R\,(0, m_6)$				
1	$R\,(c, m_2)$	$R\,(1, m_2)$	$L\,(c, m_4)$	$L\,(1, m_4)$						
c	$L\,(c, m_5)$	$L\,(c, m_3)$	$L\,(0, m_5)$	$R\,(c, m_0)$	$L\,(c, m_5)$	$R\,(1, m_6)$				

Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Вернемся к лекции?

Задача IV.3. (Ответ приведен на стр.822.) Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Задача IV.4. (Ответ приведен на стр.849.) Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Задача IV.5. (Ответ приведен на стр.915.) Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\underbrace{0\,0}_{0}1\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответы и решения

Решение задачи 1.

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \quad \rangle$.

Здесь Λ — пустой символ.

1111

11111

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \qquad \rangle$.

Здесь Λ — пустой символ.

$1111 \to\ c1111$

11111

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \qquad \rangle$.
Здесь Λ — пустой символ.

$1111 \to\ c1111$

$11111 \to\ c11111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.

Здесь Λ — пустой символ.

$1111 \to\ c1111$

$11111 \to\ c11111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.

Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111$

$11111 \to\ c11111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$.

Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111$

$11111 \to\ c11111 \to\ 1d1111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; \qquad (\Lambda; c; 0) \rangle\,.$

Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111$

$11111 \to\ c11111 \to\ 1d1111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; \qquad (\Lambda; c; 0) \rangle\,.$

Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11$

$11111 \to\ c11111 \to\ 1d1111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; \qquad (\Lambda; c; 0) \rangle\,.$

Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; (d1; 1c; 0)\,; \qquad\qquad (\Lambda; c; 0) \rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; (d1; 1c; 0)\,; \qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; (d1; 1c; 0)\,; \qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; (d1; 1c; 0)\,; \qquad\qquad (\Lambda; c; 0) \rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,; (d1; 1c; 0)\,; \qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\qquad\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\qquad\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\qquad\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\qquad\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\qquad\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\ (1t; t; 0)\,;\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\ (1t; t; 0)\,;\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\ (1t; t; 0)\,;\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t \to\ 11t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\ (1t; t; 0)\,;\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t \to\ 11t \to\ 1t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\ (d1; 1c; 0)\,;\ (c; \Lambda; 1)\,;\ (d; t; 0)\,;\ (1t; t; 0)\,;\qquad (\Lambda; c; 0)\rangle\,.$
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t \to\ 11t \to\ 1t \to\ t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\, (d1; 1c; 0)\,;\, (c; \Lambda; 1)\,;\, (d; t; 0)\,;\, (1t; t; 0)\,;\, (t; \Lambda; 1)\,;\, (\Lambda; c; 0) \rangle$.
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t \to\ 11t \to\ 1t \to\ t$

Задача 1. Дано входное слово, состоящее из единиц. Написать нормальный алгорифм Маркова, стирающий входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ. $\mathcal{A} = \langle (c1; 1d; 0)\,;\, (d1; 1c; 0)\,;\, (c; \Lambda; 1)\,;\, (d; t; 0)\,;\, (1t; t; 0)\,;\, (t; \Lambda; 1)\,;\, (\Lambda; c; 0) \rangle$.
Здесь Λ — пустой символ.

$1111 \to\ c1111 \to\ 1d111 \to\ 11c11 \to\ 111d1 \to\ 1111c \to\bullet\ 1111.$

$11111 \to\ c11111 \to\ 1d1111 \to\ 11c111 \to\ 111d11 \to\ 1111c1 \to\ 11111d \to\ 11111t \to\ 1111t\ \to$
$\to\ 111t \to\ 11t \to\ 1t \to\ t \to\bullet\ \Lambda$

Решение задачи 2.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0; 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

$110\to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$\rangle$.

$110\to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle$

$(\Lambda; c; 0)\rangle$.

$110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110}\mapsto\underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A}=\langle$

$(\Lambda;c;0)\rangle$.

$110\to c110\to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle$

$(\Lambda; c; 0)\rangle$.

$110 \to c110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$

$(\Lambda; c; 0)\rangle\,.$

$110 \to c110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$

$(\Lambda; c; 0)\rangle\,.$

$110 \to c110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$

$(\Lambda; c; 0)\rangle\,.$

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$

$(\Lambda; c; 0)\rangle\,.$

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;\ (2e; e2; 0)\,;\ (\Lambda; c; 0) \rangle$.

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;\ (2e; e2; 0)\,;\ (\Lambda; c; 0) \rangle$.

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$ $(2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;$ $(\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;$ $(2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;$ $(\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\ ;\ (c1; e3; 0)\ ;\ (c2; 2c; 0)\ ;\ (c3; 3c; 0)\ ;\ (2d; d2; 0)\ ;\ (3d; d3; 0)\ ;\ (3e; e3; 0)\ ;$
$(2e; e2; 0)\ ;\ (d; 2c; 0)\ ;\ (e; 3c; 0)\ ;\ (\Lambda; c; 0) \rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\ ;\ (c1; e3; 0)\ ;\ (c2; 2c; 0)\ ;\ (c3; 3c; 0)\ ;\ (2d; d2; 0)\ ;\ (3d; d3; 0)\ ;\ (3e; e3; 0)\ ;$
$(2e; e2; 0)\ ;\ (d; 2c; 0)\ ;\ (e; 3c; 0)\ ;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to t011110 \to \bullet$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (t; \Lambda; 1)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to t011110 \to \bullet$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ \ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (t; \Lambda; 1)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to t011110 \to \bullet$

Описание алгоритма: в начале слова «включаем» «курсор» c.
«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).
«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.
Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.
«Курсор» c движется вправо, перепрыгивая 2 и 3.
Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.
Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ \ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (t; \Lambda; 1)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to t011110 \to \bullet$
$\to \bullet 011110$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Задача 2. Написать нормальный алгорифм Маркова, приписывающий слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\underbrace{110}$.

Ответ. $\mathcal{A} = \langle (c0; d2; 0)\,;\ (c1; e3; 0)\,;\ (c2; 2c; 0)\,;\ (c3; 3c; 0)\,;\ (2d; d2; 0)\,;\ (3d; d3; 0)\,;\ (3e; e3; 0)\,;$
$(2e; e2; 0)\,;\ (d; 2c; 0)\,;\ (e; 3c; 0)\,;\ (c; t; 0)\,;\ (2t; t0; 0)\,;\ (3t; t1; 0)\,;\ (t; \Lambda; 1)\,;\ (\Lambda; c; 0)\rangle$.

$110 \to c110 \to e310 \to 3c310 \to 33c10 \to 33e30 \to 3e330 \to e3330 \to 3c3330 \to 33c330 \to 333c30 \to 3333c0 \to$
$\to 3333d2 \to 333d32 \to 33d332 \to 3d3332 \to d33332 \to 2c33332 \to 23c3332 \to 233c332 \to 2333c32 \to$
$\to 23333c2 \to 233332c \to 233332t \to 23333t0 \to 2333t10 \to 233t110 \to 23t1110 \to 2t11110 \to t011110 \to \bullet$
$\to \bullet 011110$

Описание алгоритма: в начале слова «включаем» «курсор» c.

«Курсор» c, двигаясь слева-направо и встретив 0, «превращает» ее в 2, а встретив 1, «превращает» ее 3, в сам «превращается» либо в d (d «несет влево» 0), либо в e (e «несет влево» 1).

«Курсор» d или e движется влево, «перепрыгивая» через 2 и 3.

Когда d или e дойдет до начала изменяемого слова, после чего «вываливает», соответственно, 2 или 3, «превращаясь» в «курсор» c.

«Курсор» c движется вправо, перепрыгивая 2 и 3.

Если «курсор» c, двигаясь вправо, наталкивается на конец слова, значит, процесс перекодирования и дублирования завершен, но мы получили «заколдованное слово». Осталось его «расколдовать», поэтому «курсор» c превращается в t, который движется влево, причем, «перепрыгивая» 2 или 3 «превращает» из, соответственно, в 0 или 1.

Перепрыгнув самый левый символ слова, курсор t с чувством глубокого удовлетворения исчезает, чем завершает процесс.

Решение задачи 3.

Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

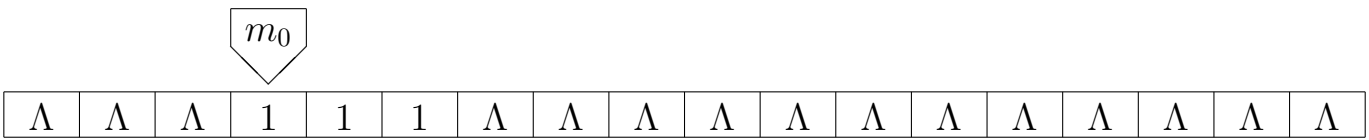
Ответ.

	m_0	m_1	m_2	m_3
Λ				
1				

Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

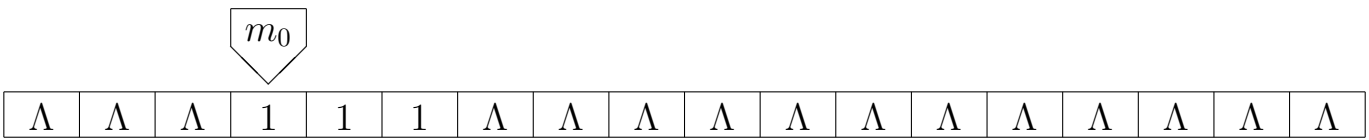
	m_0	m_1	m_2	m_3
Λ				
1				



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

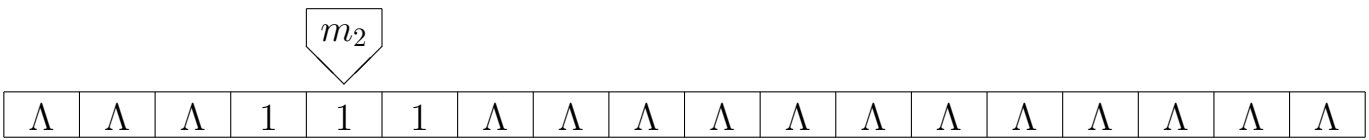
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$			



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

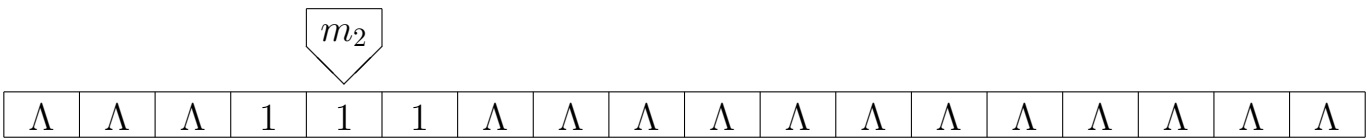
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$			



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

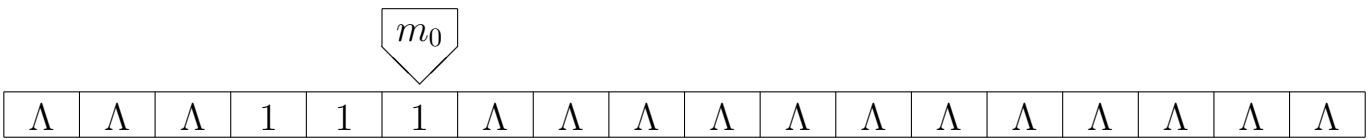
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$		$R(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

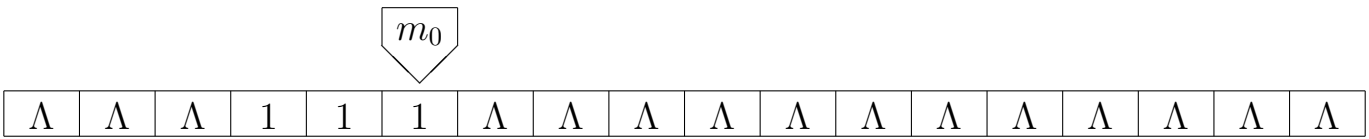
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$		$R(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

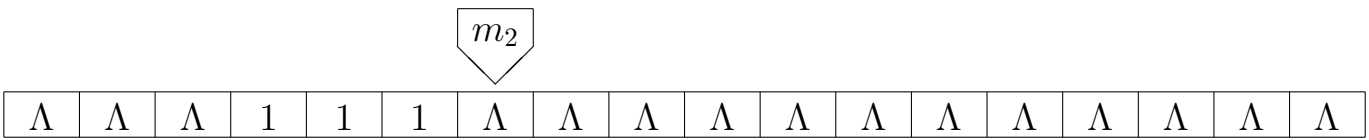
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$		$R(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

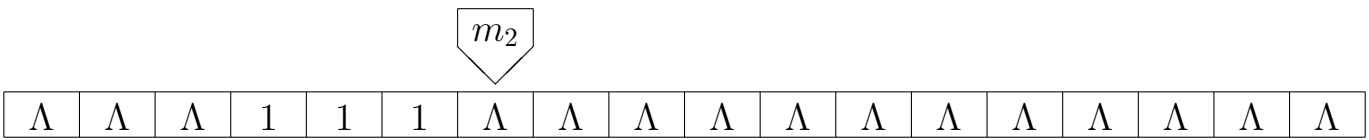
	m_0	m_1	m_2	m_3
Λ				
1	$R(1, m_2)$		$R(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

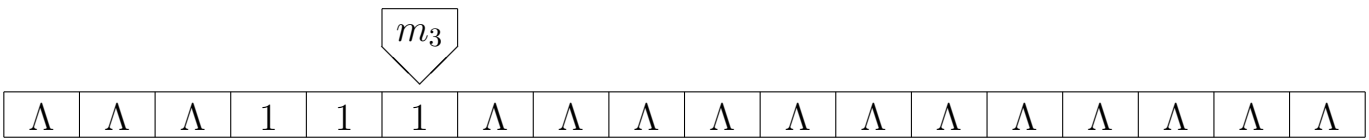
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

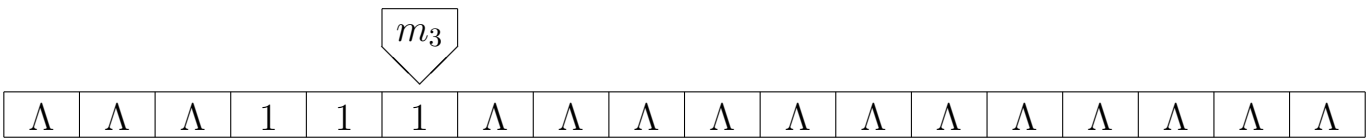
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

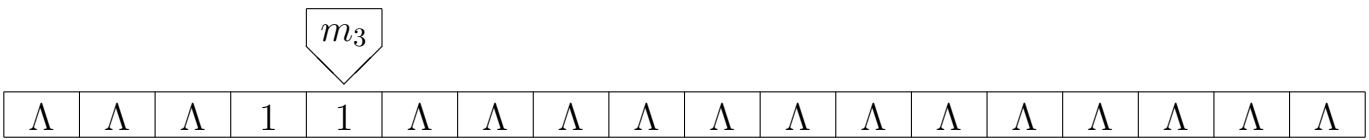
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

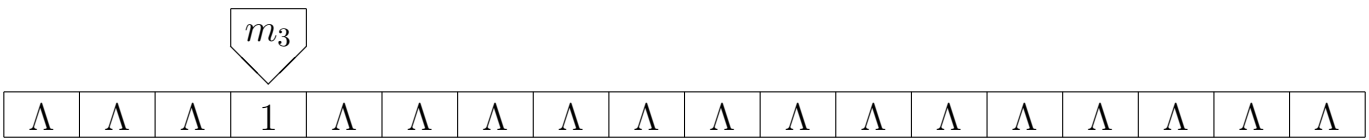
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

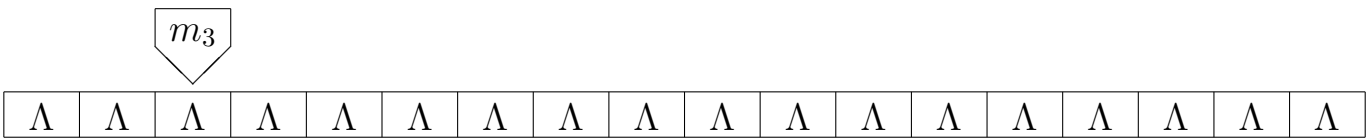
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

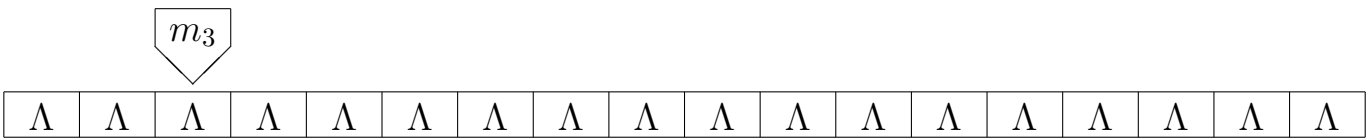
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

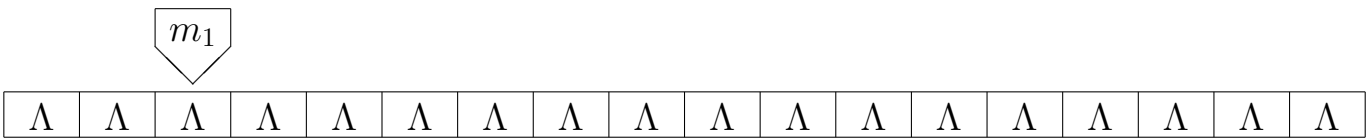
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

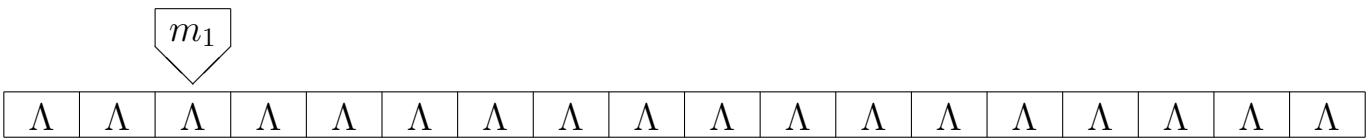
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

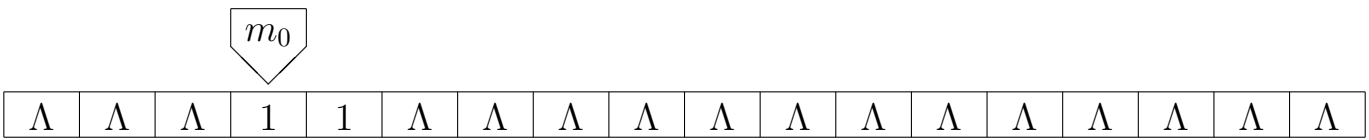
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

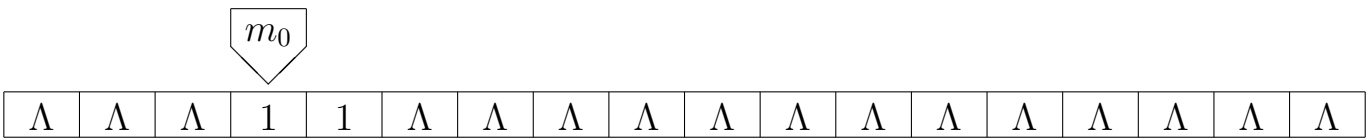
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

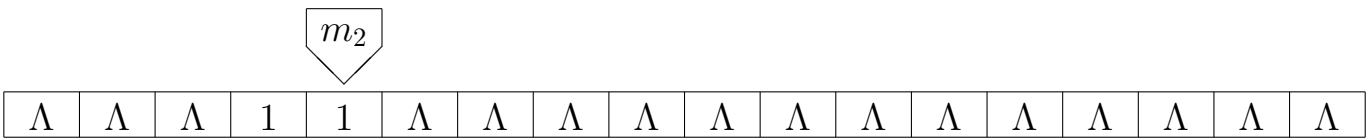
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

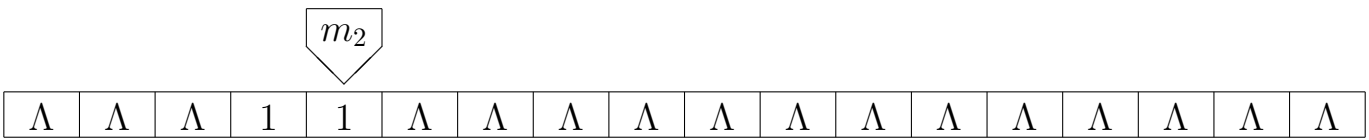
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

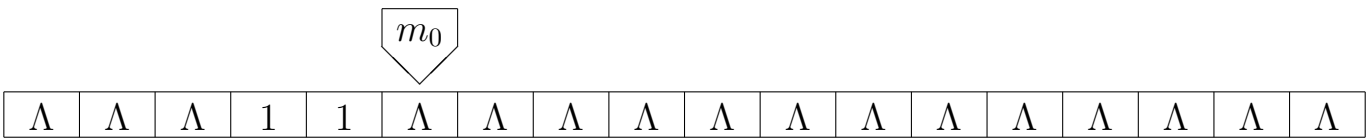
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

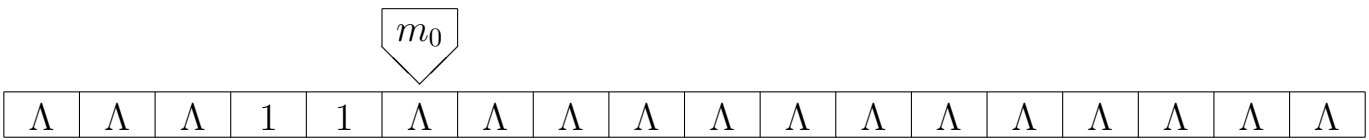
	m_0	m_1	m_2	m_3
Λ			$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

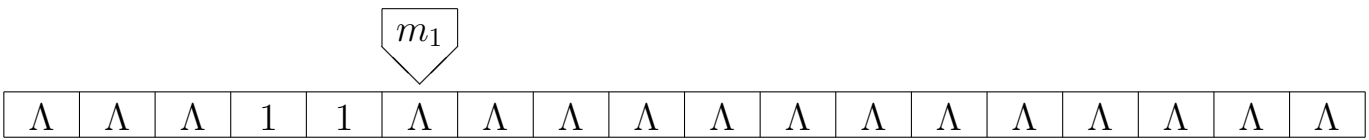
	m_0	m_1	m_2	m_3
Λ	(Λ, m_1)		$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

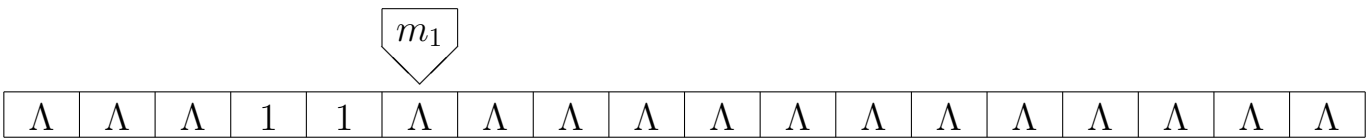
	m_0	m_1	m_2	m_3
Λ	(Λ, m_1)		$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Задача 3. Дано входное слово, состоящее из единиц. Написать машину Тьюринга, стирающую входное слово нечетной длины, и оставляющее неизменным входное слово четной длины.

Ответ.

	m_0	m_1	m_2	m_3
Λ	(Λ, m_1)		$L\,(\Lambda, m_3)$	(Λ, m_1)
1	$R\,(1, m_2)$		$R\,(1, m_0)$	$L\,(\Lambda, m_3)$



Решение задачи 4.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

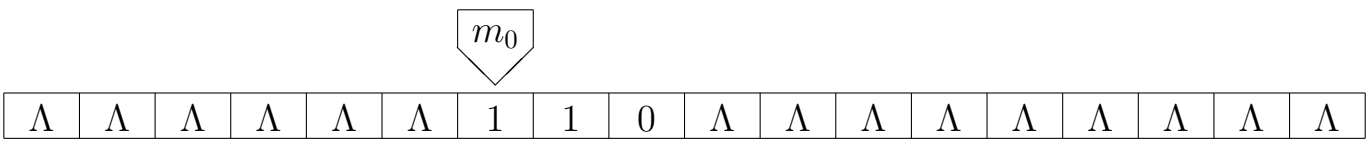
Ответ.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ							
0							
1							
2							
3							

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

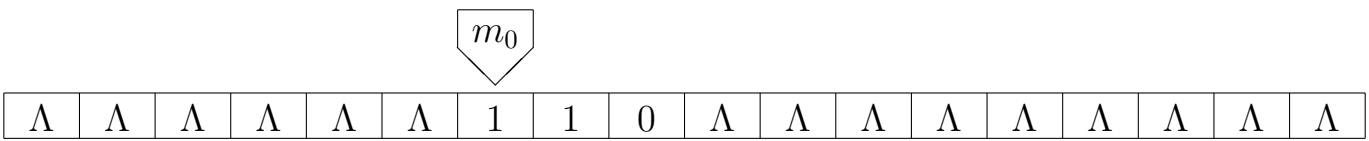
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ							
0							
1							
2							
3							



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

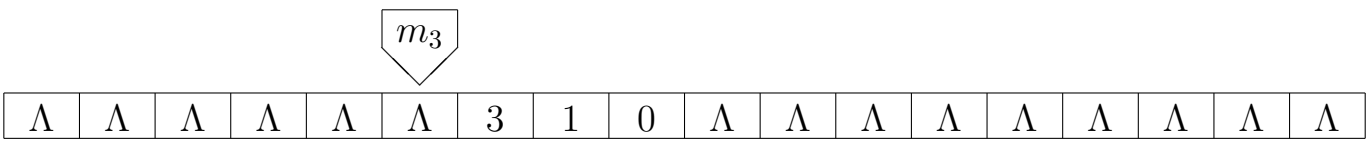
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ							
0							
1	$L\,(3, m_3)$						
2							
3							



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

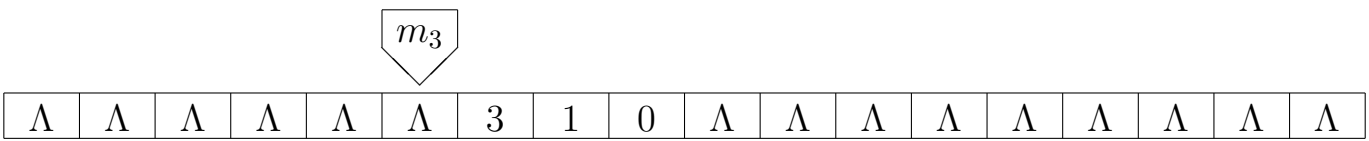
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ							
0							
1	$L\,(3, m_3)$						
2							
3							



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

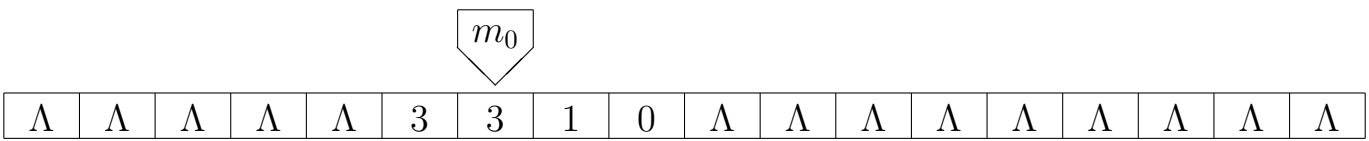
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3							



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

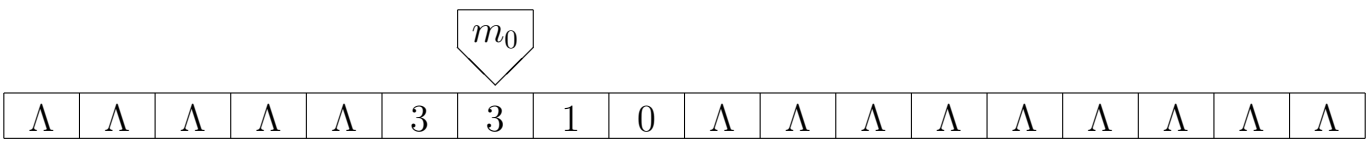
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3							



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

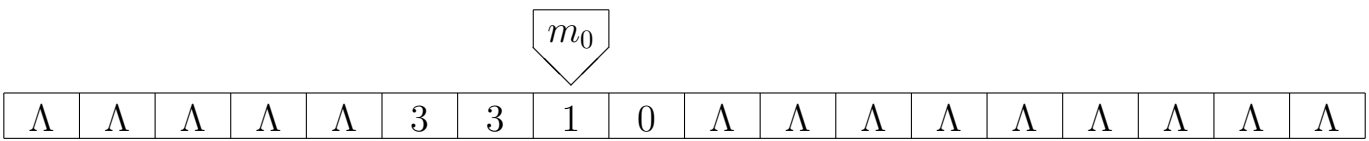
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$						



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

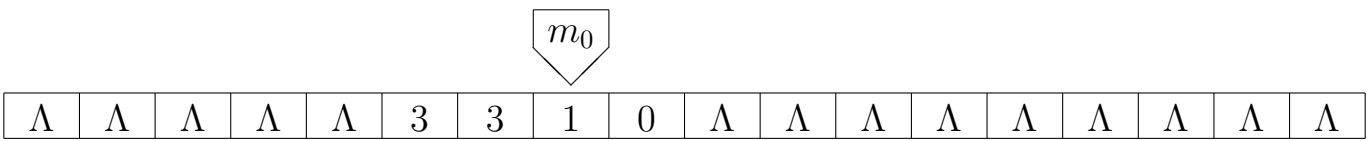
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$						



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

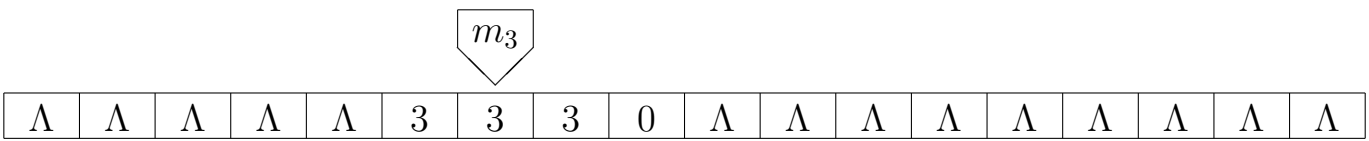
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$R\,(3, m_0)$						
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$						



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

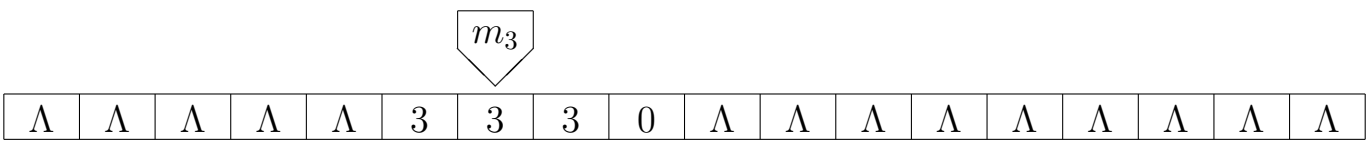
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$						



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

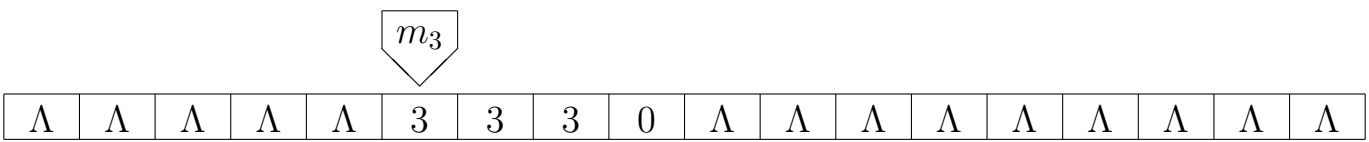
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

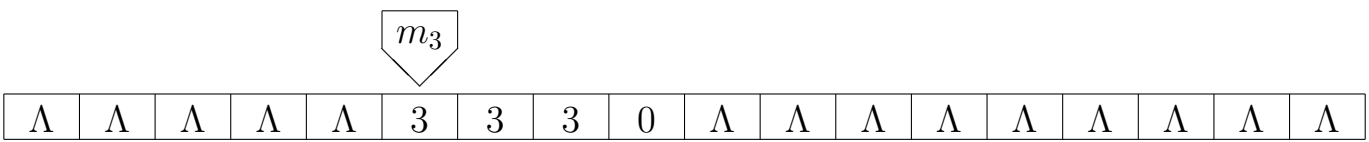
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

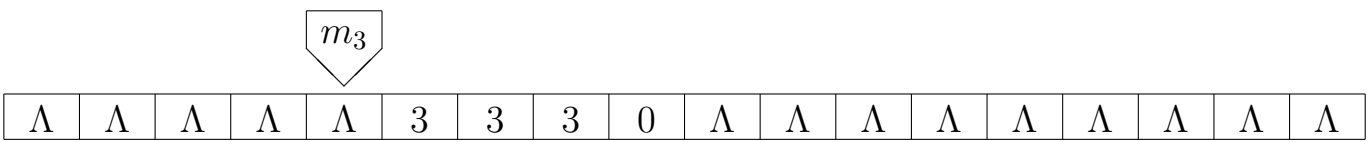
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

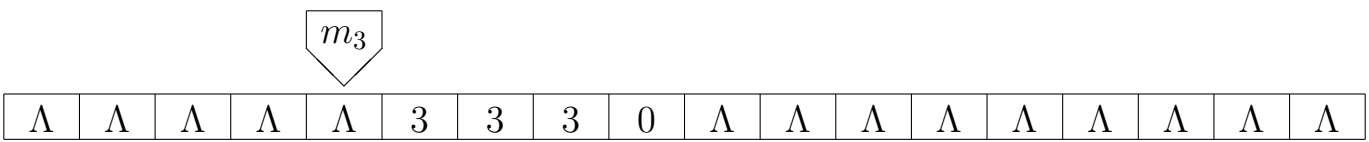
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

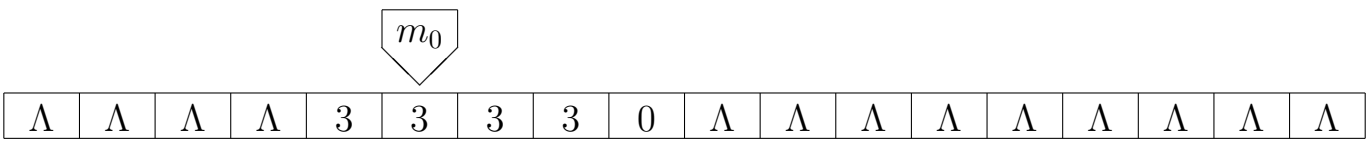
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

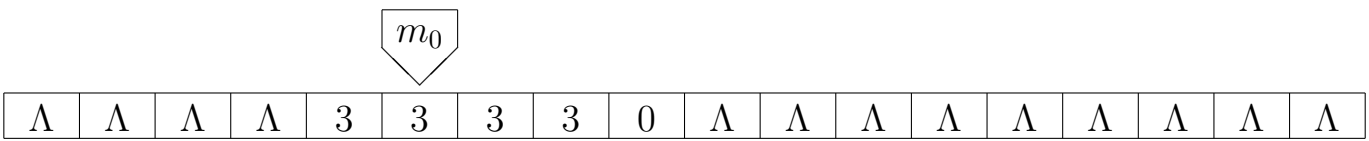
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

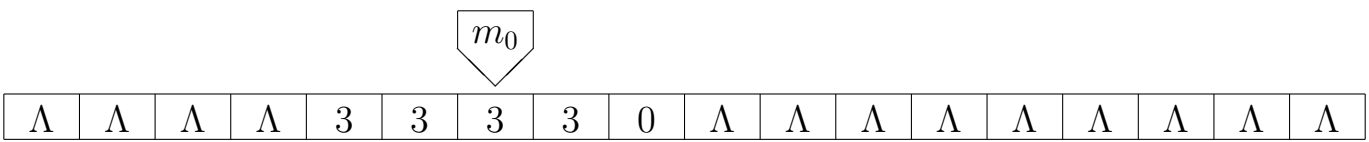
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

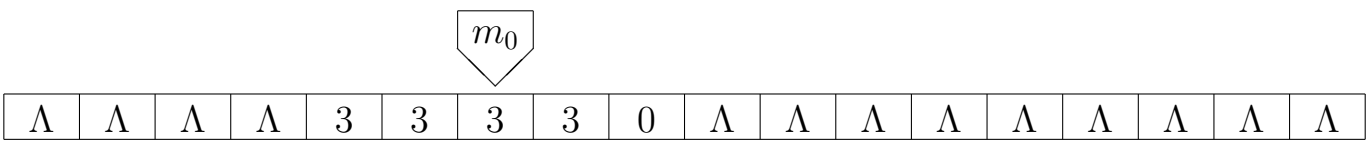
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

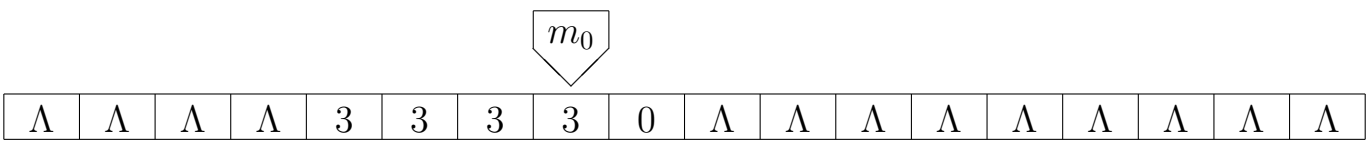
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

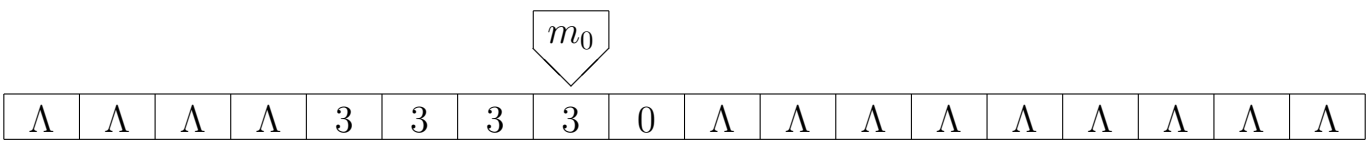
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

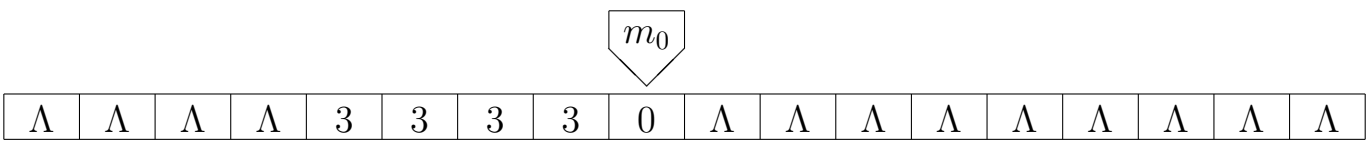
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

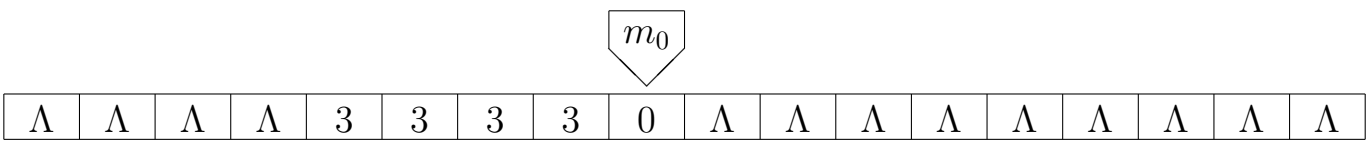
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0							
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

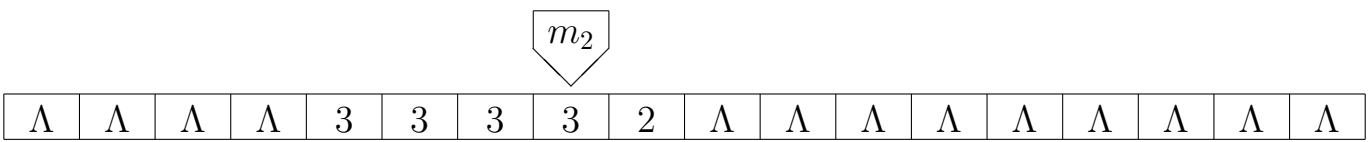
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

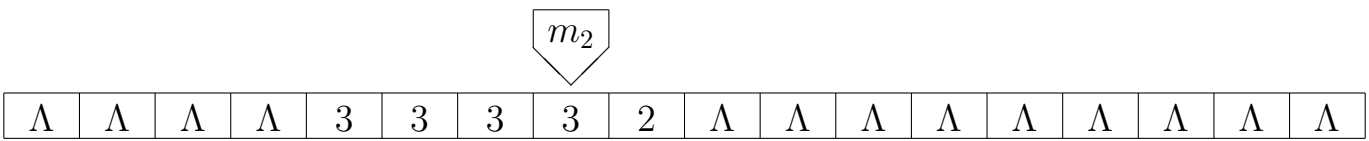
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$			$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

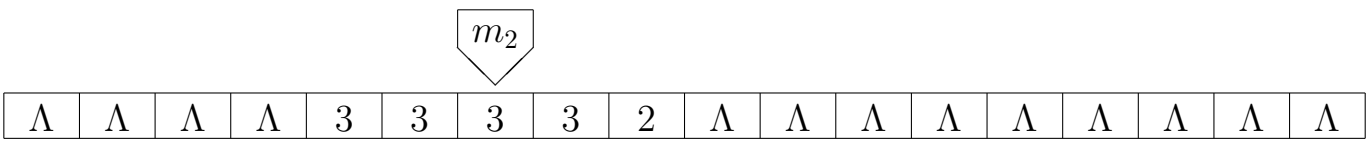
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

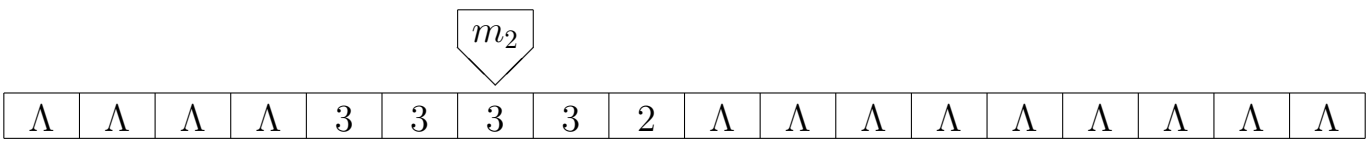
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

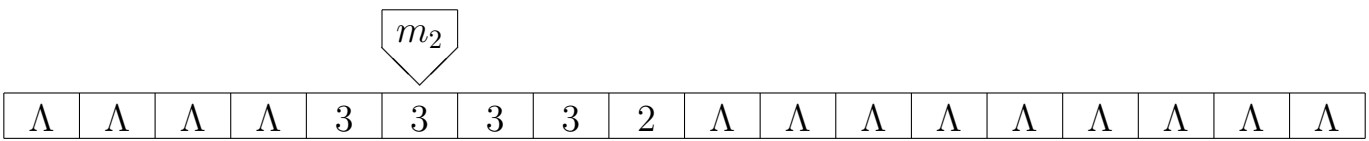
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

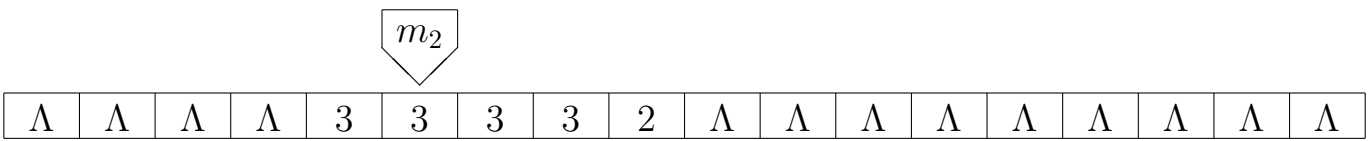
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

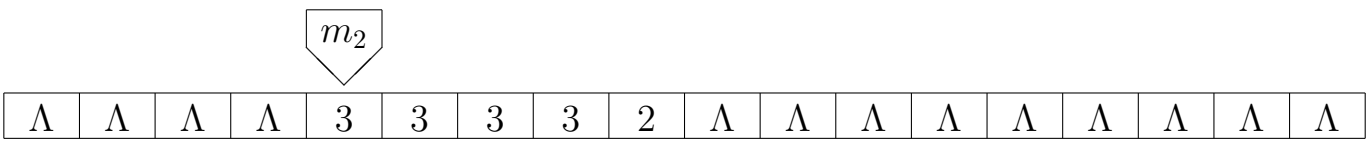
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

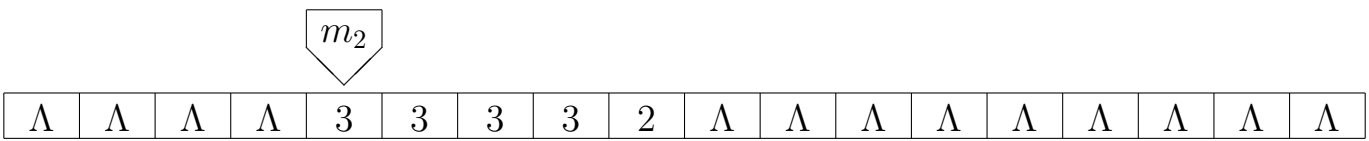
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

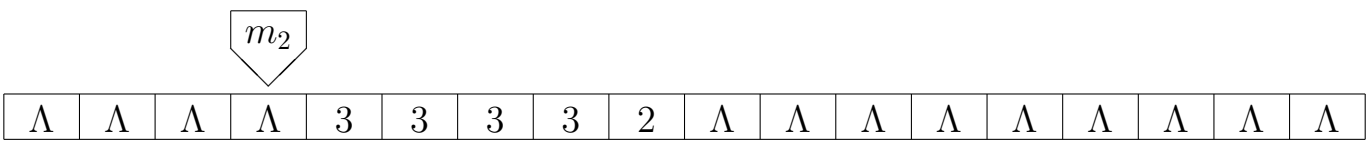
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

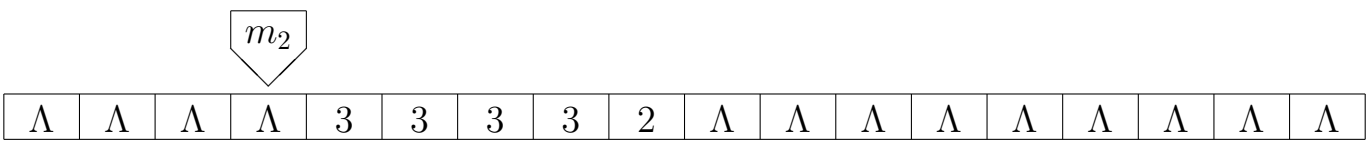
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ				$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

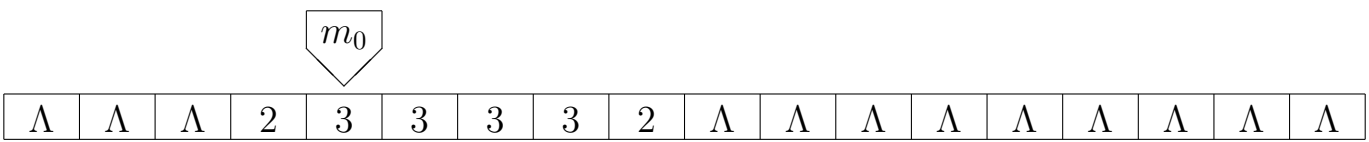
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

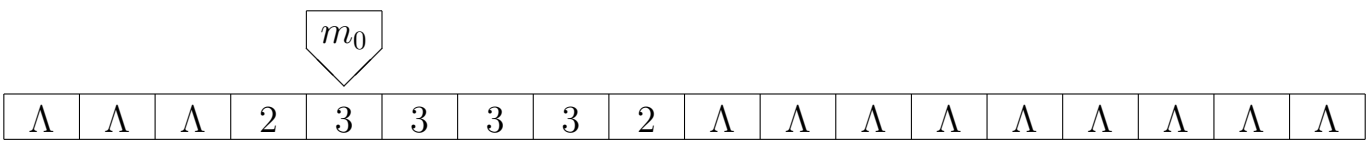
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

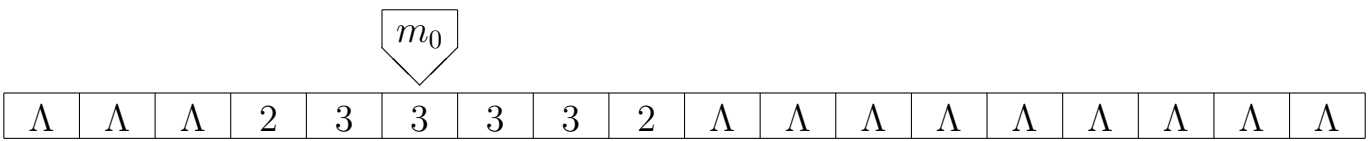
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

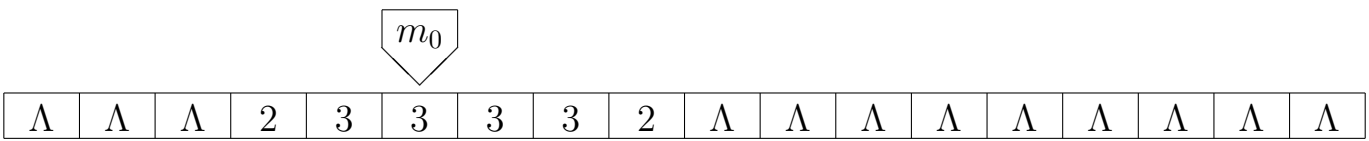
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

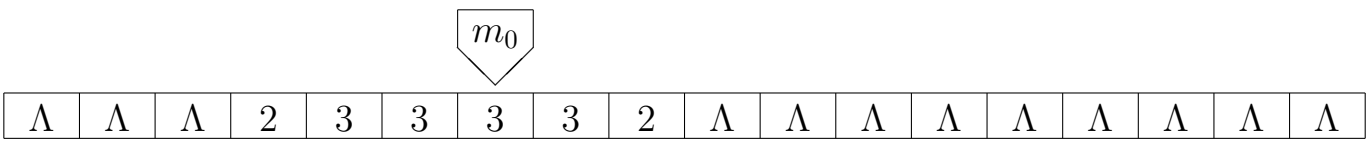
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

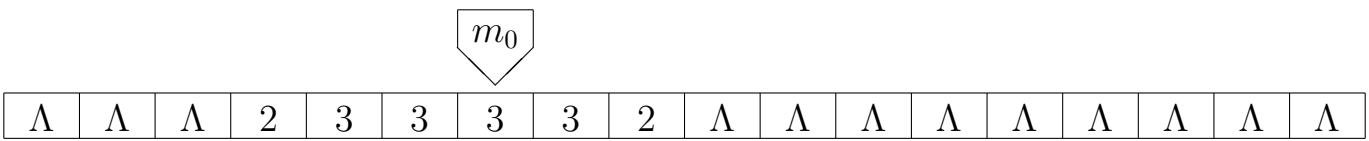
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

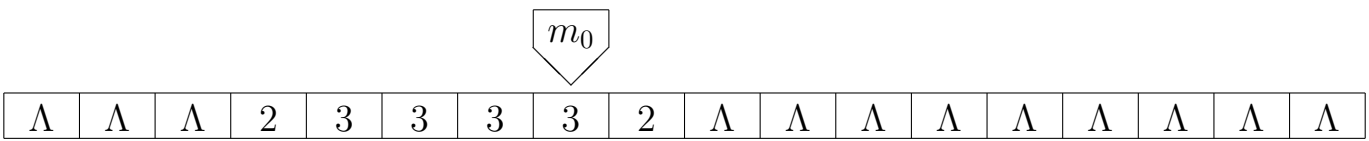
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

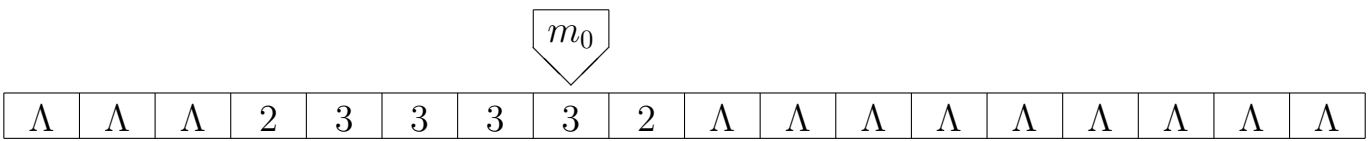
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

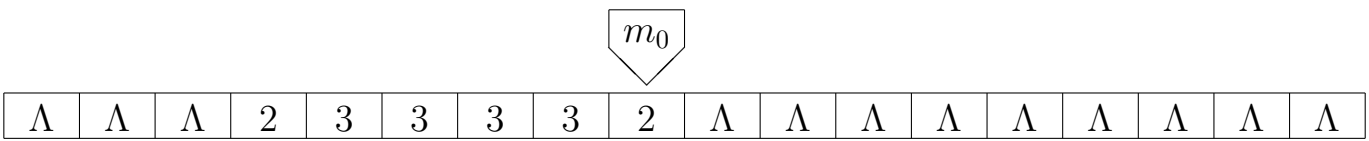
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

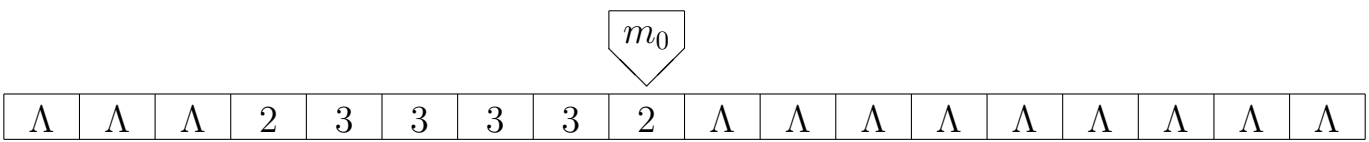
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2							
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

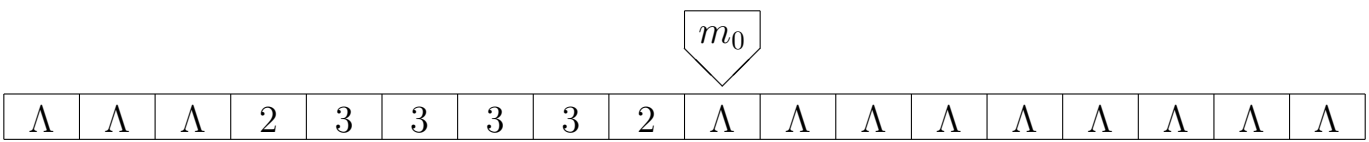
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$						
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

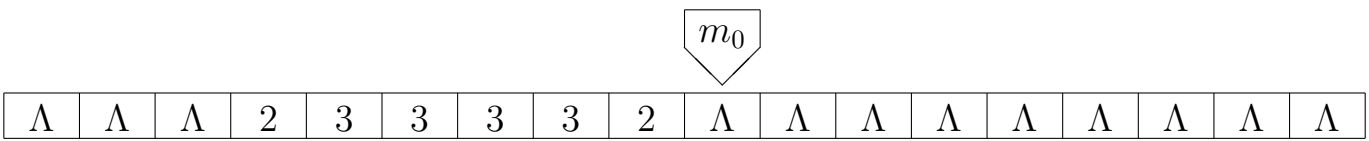
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ			$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$						
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

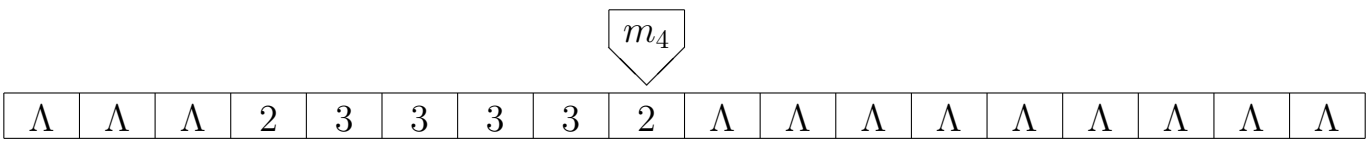
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$						
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

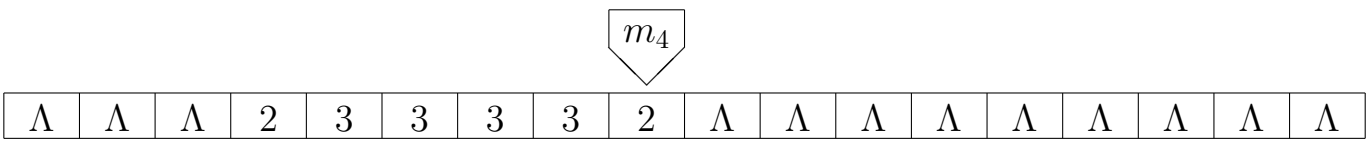
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$						
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

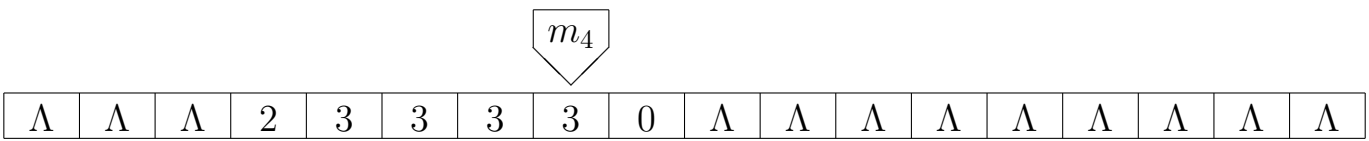
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

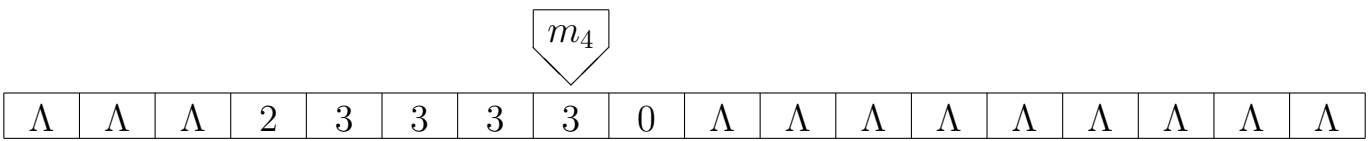
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$			



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

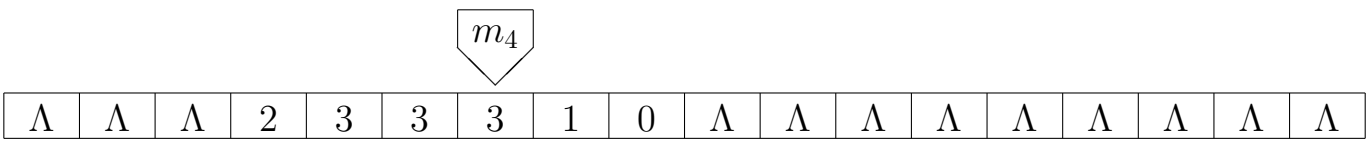
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

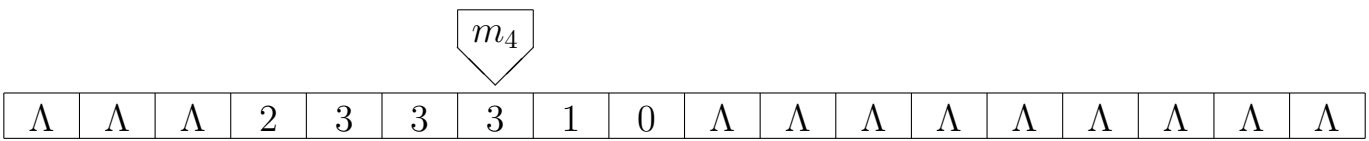
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

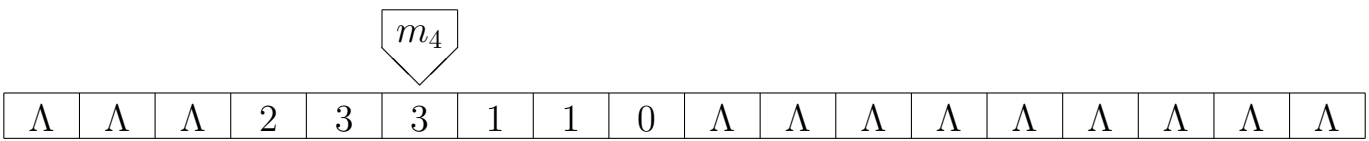
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

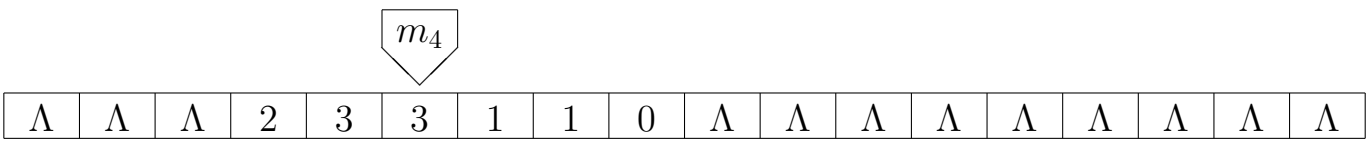
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

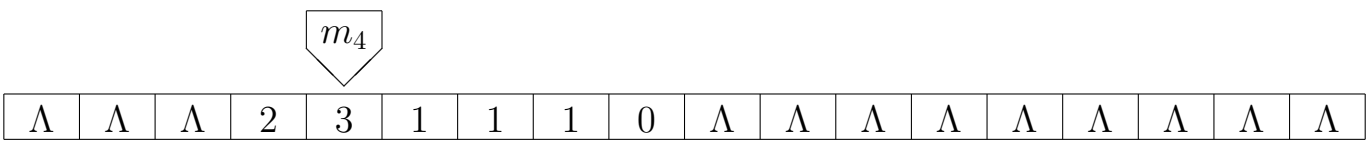
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

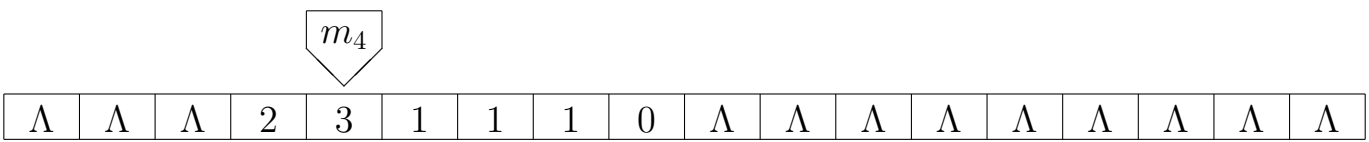
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

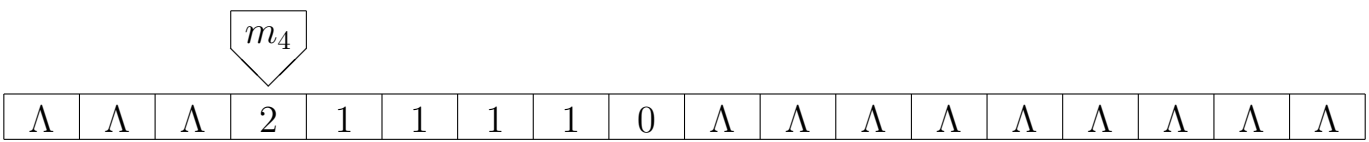
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

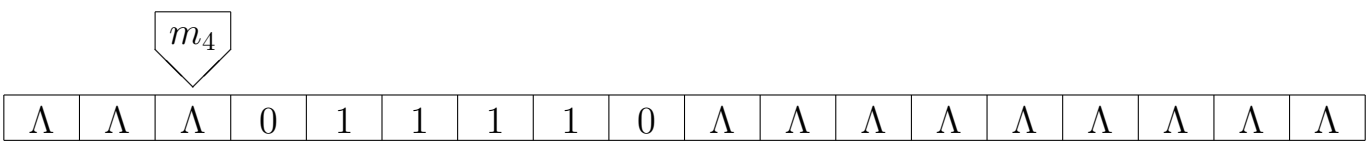
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		

Λ	Λ	Λ	2	1	1	1	1	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

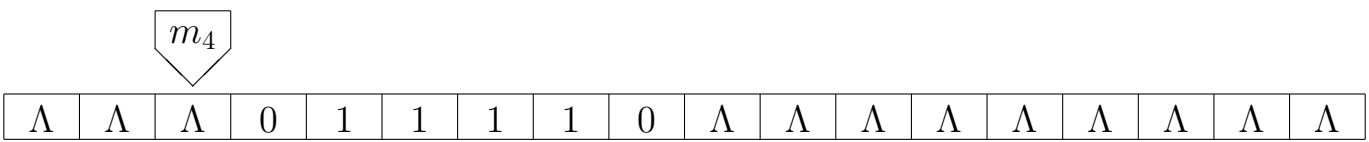
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$			
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

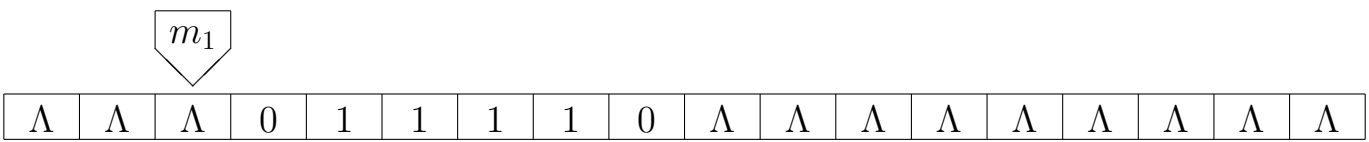
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

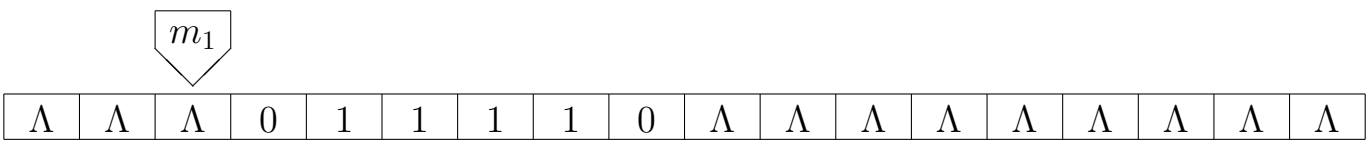
	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$				$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		

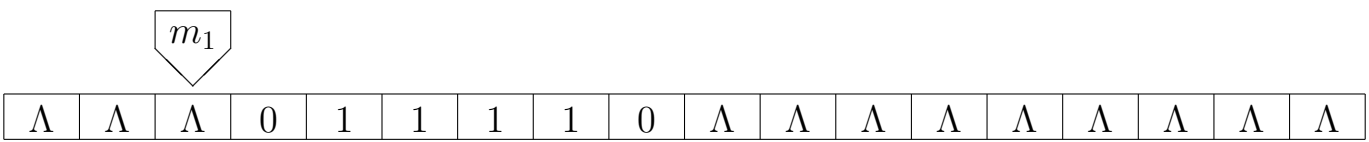


Мы не предусмотрели «перепрыгивание» через 2.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$		$L\,(2, m_2)$		$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		

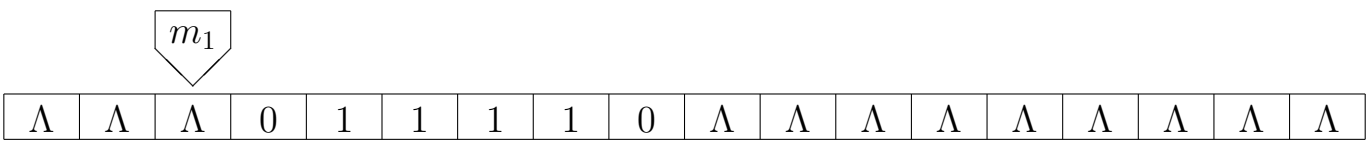


Мы не предусмотрели «перепрыгивание» через 2.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

	m_0	m_1	m_2	m_3	m_4	m_5	m_6
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$		$L\,(2, m_2)$	$L\,(2, m_3)$	$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		

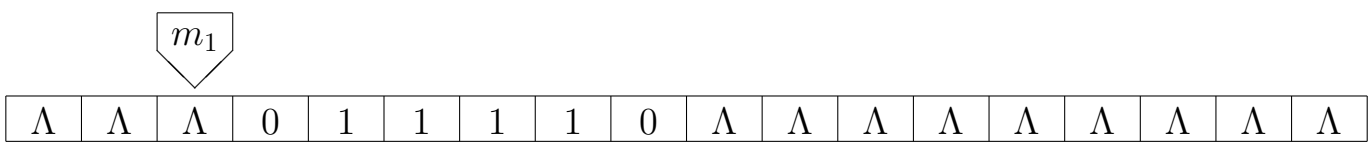


Мы не предусмотрели «перепрыгивание» через 2.

Задача 4. Написать машину Тьюринга, приписывающую слева ко входному слову в алфавите $\{0;\ 1\}$ копию этого входного слова в зеркальном отражении, например: $\underbrace{110} \mapsto \underbrace{011}\,\underbrace{110}$.

Ответ.

	m_0	m_1	m_2	m_3	m_4	~~m_5~~	~~m_6~~
Λ	$L\,(\Lambda, m_4)$		$R\,(2, m_0)$	$R\,(3, m_0)$	(Λ, m_1)		
0	$L\,(2, m_2)$						
1	$L\,(3, m_3)$						
2	$R\,(2, m_0)$		$L\,(2, m_2)$	$L\,(2, m_3)$	$L\,(0, m_4)$		
3	$R\,(3, m_0)$		$L\,(3, m_2)$	$L\,(3, m_3)$	$L\,(1, m_4)$		



Теперь алгоритм завершен.

Решение задачи 5.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle$ $\rangle$
01001000

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle$ $\rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle$ $\rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle$ $\rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть последней, поскольку

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad \rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть последней, поскольку команды, находящиеся после нее выполняться не смогут никогда.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть последней, поскольку команды, находящиеся после нее выполняться не смогут никогда.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$
01001000

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть последней, поскольку команды, находящиеся после нее выполняться не смогут никогда.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$
$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
Установку «курсора» можно сделать командой $(\Lambda; c; 0)$.
В отличие от машины Тьюринга, порядок команд в алгорифме Маркова важен.
Данная команда должна быть последней, поскольку команды, находящиеся после нее выполняться не смогут никогда.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (\Lambda; c; 0)\rangle$
$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0); \qquad (\Lambda; c; 0) \rangle$
$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0); \quad (\Lambda; c; 0) \rangle$
$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.
В обрабатываемом слове нет подслова $c00$, поэтому переходим к следующей команде...

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0); \qquad (\Lambda; c; 0) \rangle$
$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\qquad\qquad (\Lambda; c; 0) \rangle$
$01001000 \to\ c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0) \rangle$
$01001000 \to\ c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to c01001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);$ $(c0; 0c; 0);$ $(c1; 1c; 0);$ $(\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$
$\to\ 0101c00$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0)\rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$
$\to\ 0101c00 \to\ 0101c0$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\quad (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\qquad (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (c; \Lambda; 1);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$
$\to\ 0101c00 \to\ 0101c0 \to\ 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (c; \Lambda; 1);\ (\Lambda; c; 0) \rangle$

$01001000 \to c01001000 \to 0c1001000 \to 01c001000 \to 01c01000 \to 010c1000 \to 0101c000 \to$
$\to 0101c00 \to 0101c0 \to 01010c$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (c; \Lambda; 1);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$
$\to\ 0101c00 \to\ 0101c0 \to\ 01010c \to\bullet\, 01010$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. $\mathcal{A} = \langle (c00; c0; 0);\ (c0; 0c; 0);\ (c1; 1c; 0);\ (c; \Lambda; 1);\ (\Lambda; c; 0) \rangle$

$01001000 \to\ c01001000 \to\ 0c1001000 \to\ 01c001000 \to\ 01c01000 \to\ 010c1000 \to\ 0101c000 \to$
$\to\ 0101c00 \to\ 0101c0 \to\ 01010c \to\bullet\, 01010$

Описание алгоритма. 1) Сначала на левом конце слова ставим «курсор» c.
2) «Курсор» c, встретив справа 00, «убивает» один из нулей.
3) «Курсор» c, встретив справа одиночный 0 или 1, «перепрыгивает» через него.
4) «Курсор» c, встретив справа «пустоту», «исчезает», останавливая программу.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										
d										

Λ	0	1	0	0	1	0	0	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

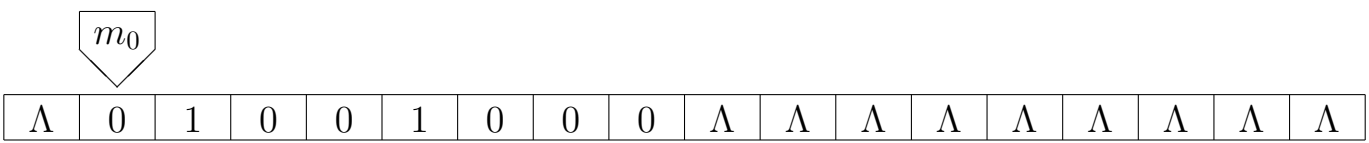
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										
c										
d										



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1										
c										
d										

Λ	0	1	0	0	1	0	0	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

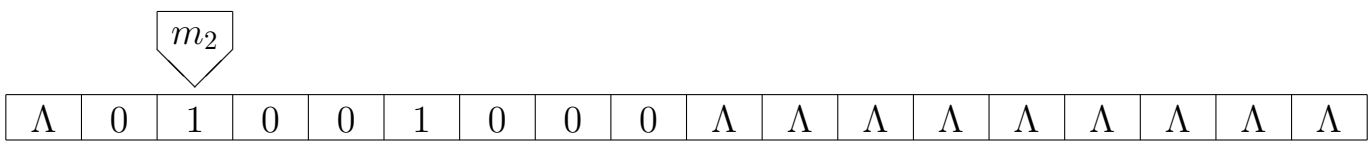
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1										
c										
d										



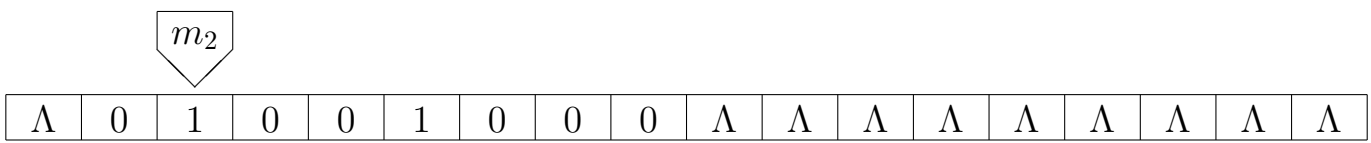
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1		$R(1, m_0)$								
c										
d										



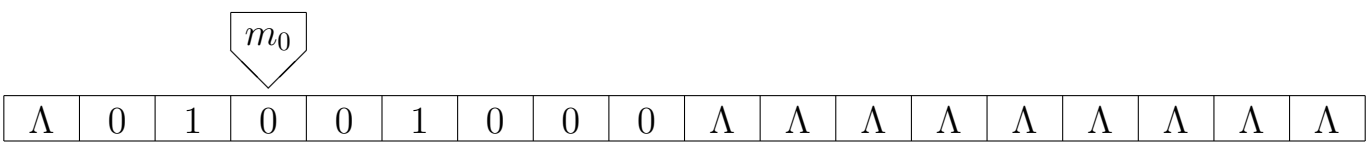
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1		$R(1, m_0)$								
c										
d										

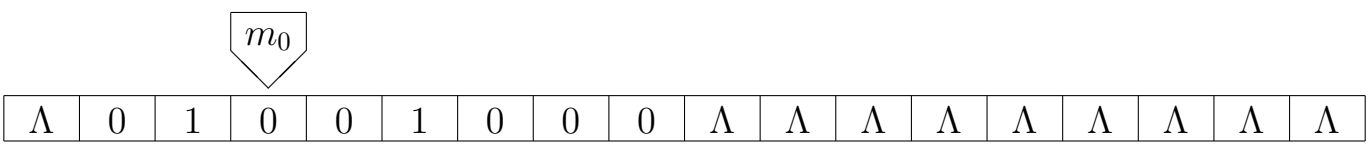


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1		$R(1, m_0)$								
c										
d										

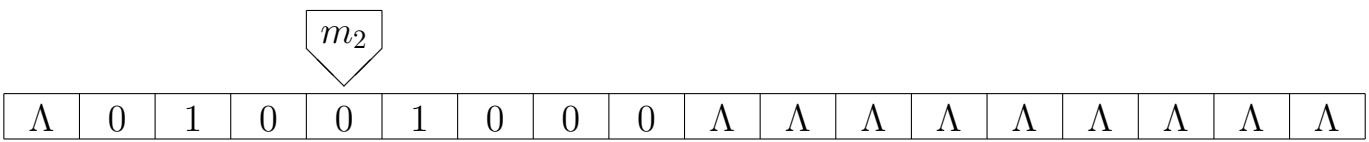


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$									
1		$R(1, m_0)$								
c										
d										



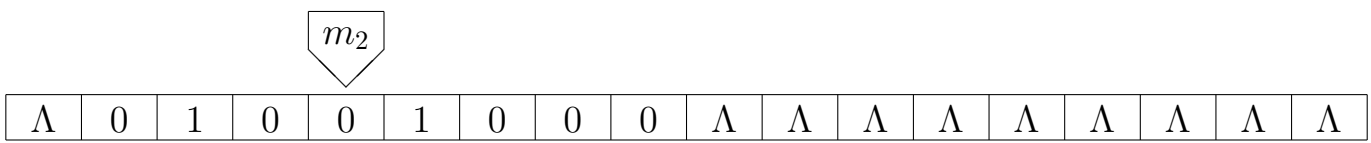
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$								
c										
d										

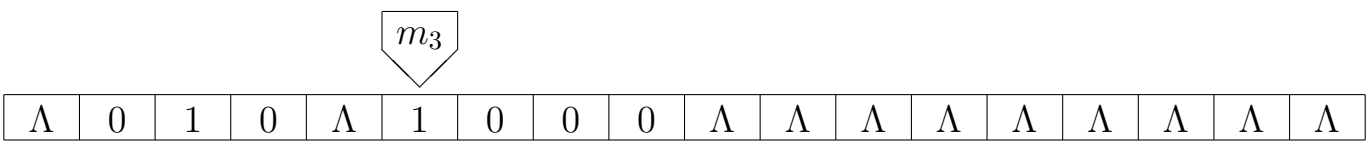


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$								
c										
d										



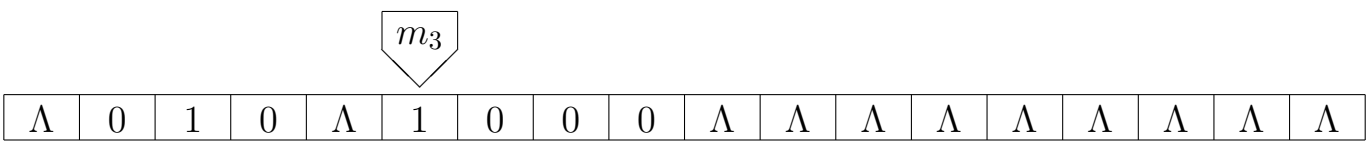
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$	$L(c, m_5)$							
c										
d										



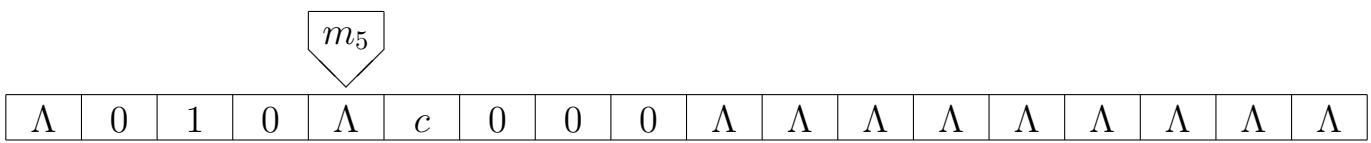
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$	$L(c, m_5)$							
c										
d										



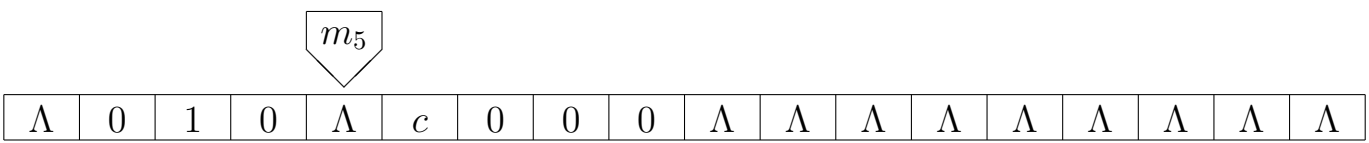
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R(1, m_0)$					
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$	$L(c, m_5)$							
c										
d										



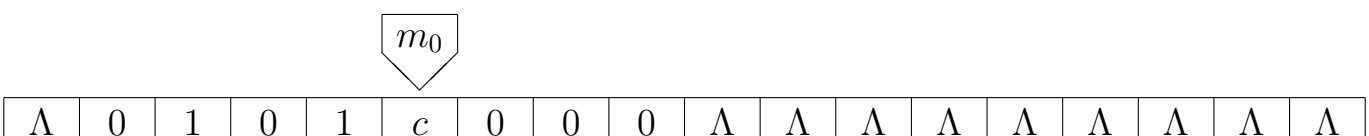
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R(1, m_0)$					
0	$R(0, m_2)$	$R(\Lambda, m_3)$								
1		$R(1, m_0)$	$L(c, m_5)$							
c										
d										



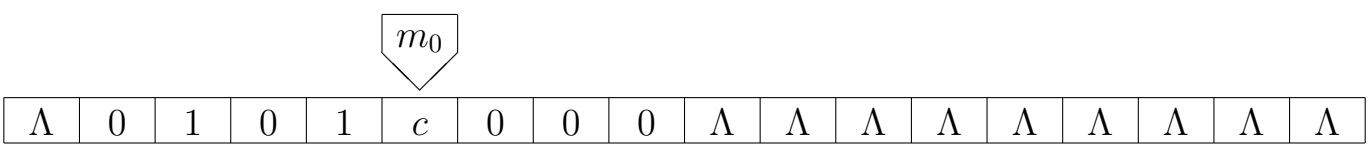
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$								
1		$R\,(1, m_0)$	$L\,(c, m_5)$							
c	$R(c, m_6)$									
d										



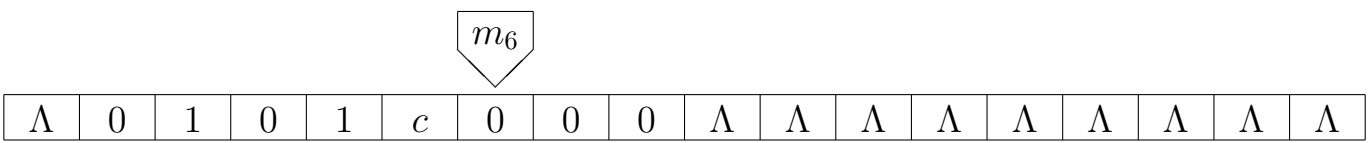
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$								
1		$R\,(1, m_0)$	$L\,(c, m_5)$							
c	$R(c, m_6)$									
d										



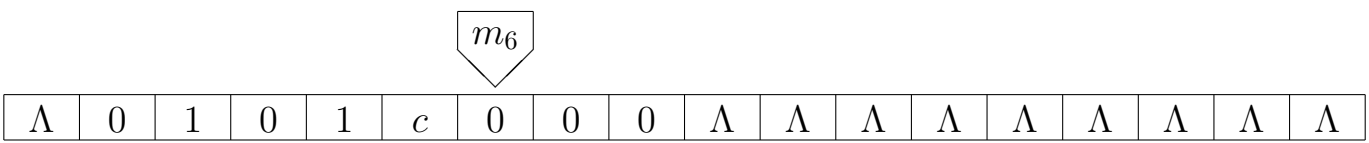
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										



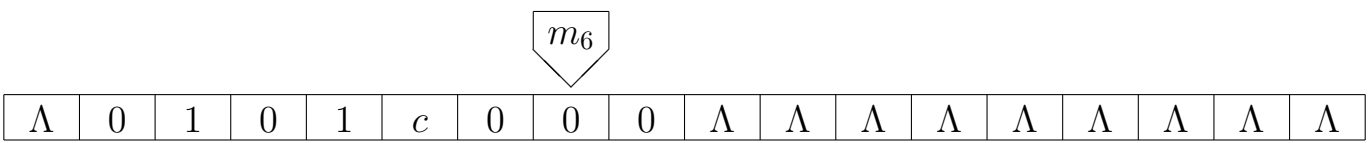
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										



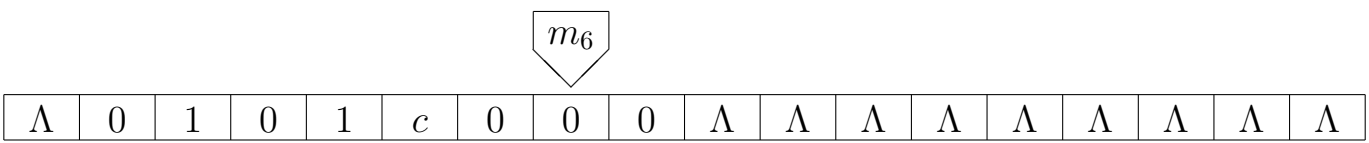
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										

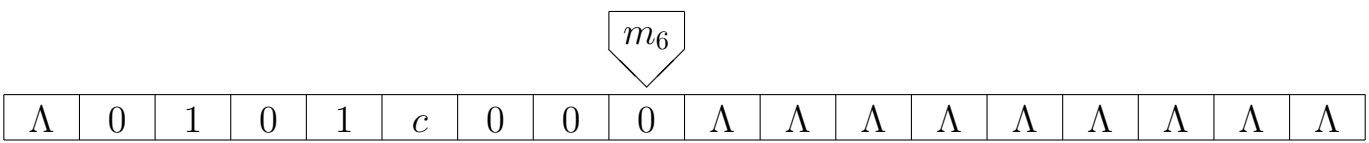


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										

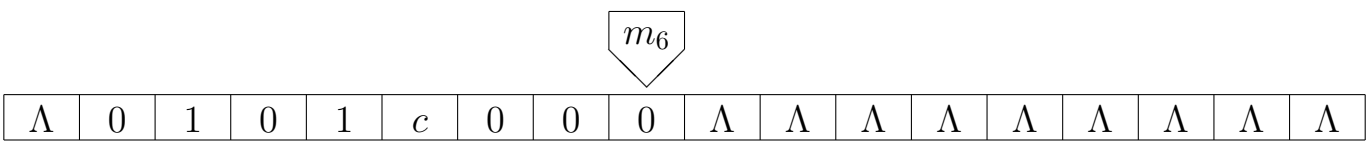


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										

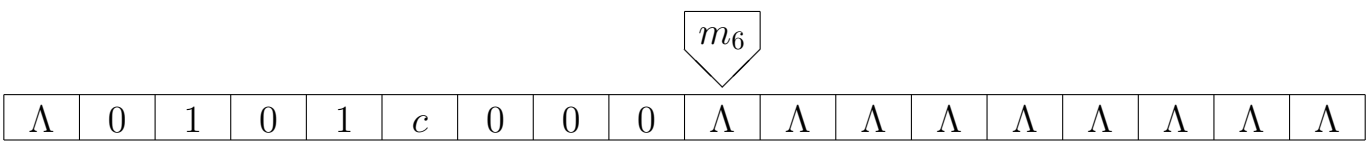


Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;
б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$					
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										



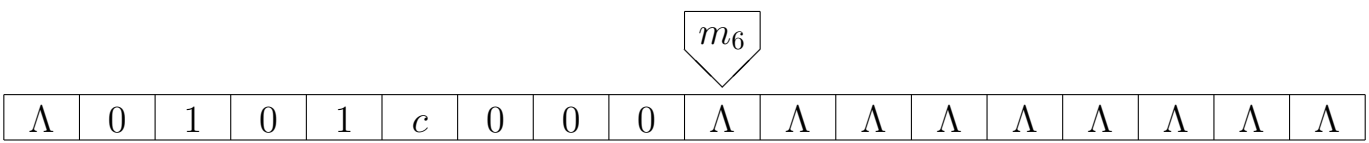
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										



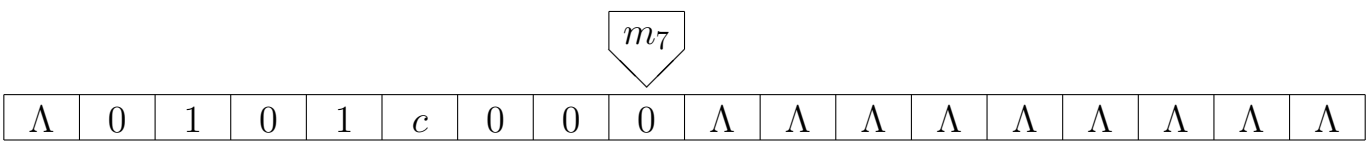
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$				
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$				
c	$R(c, m_6)$									
d										



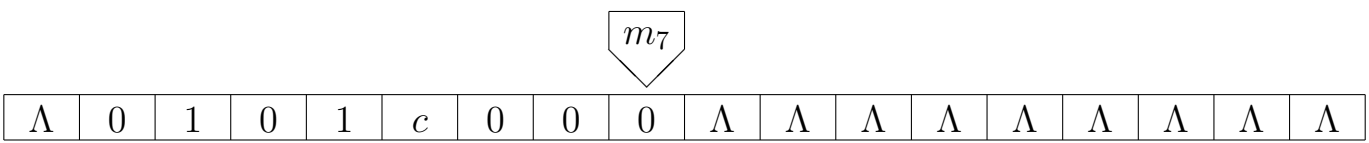
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



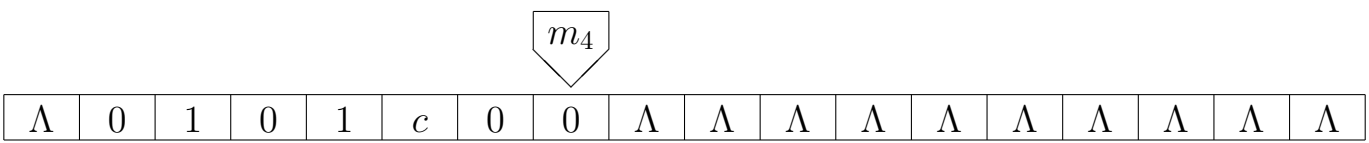
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$				$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$			$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



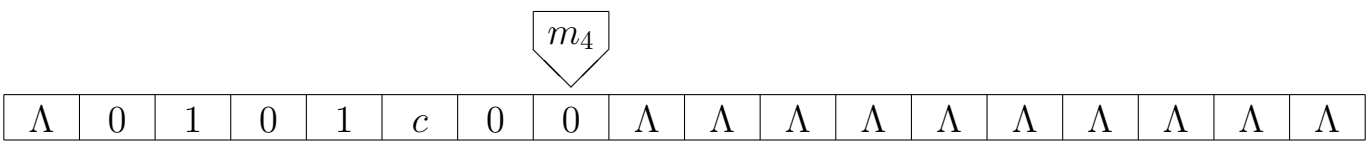
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



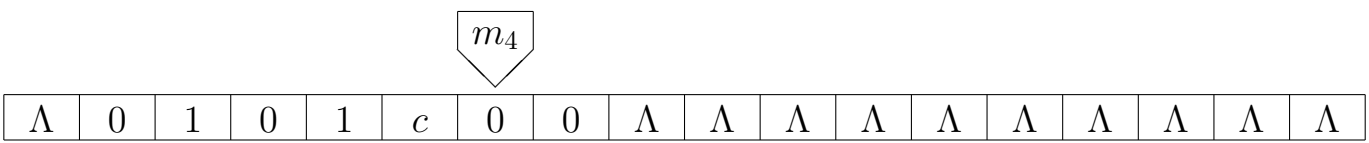
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



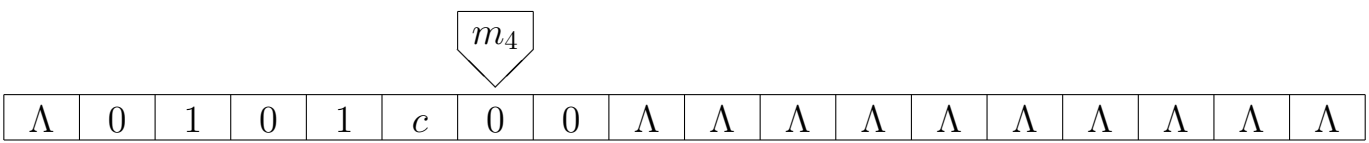
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



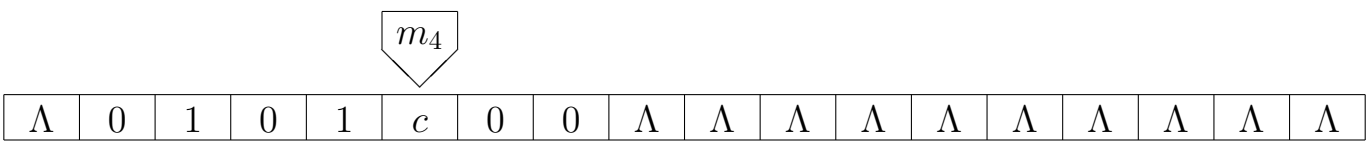
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$									
d										



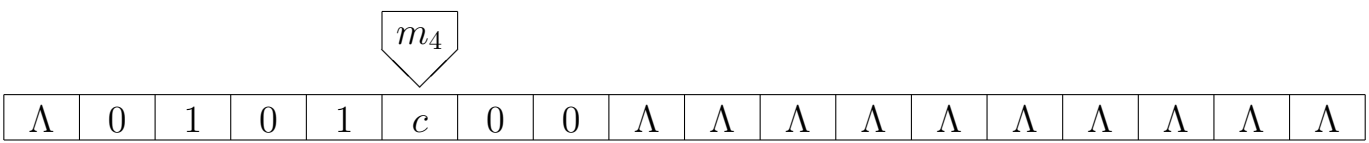
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



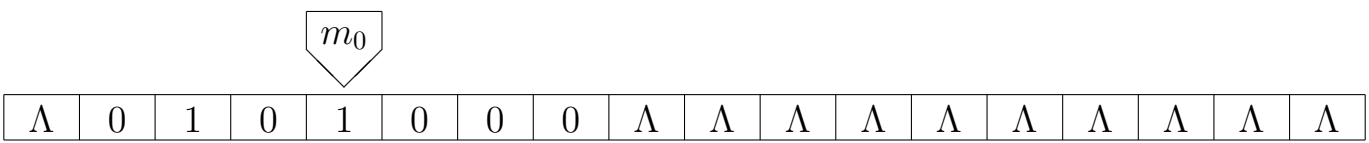
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1		$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



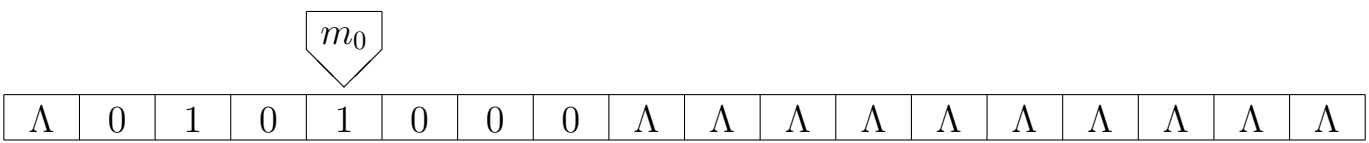
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



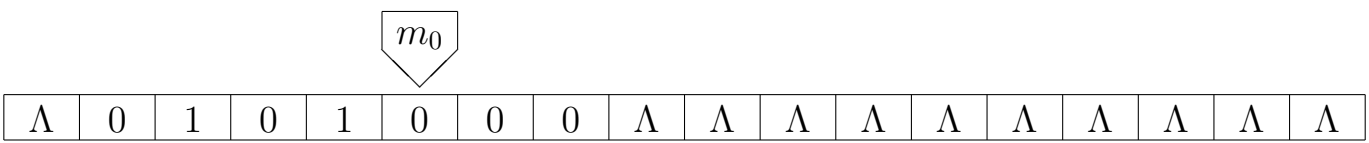
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



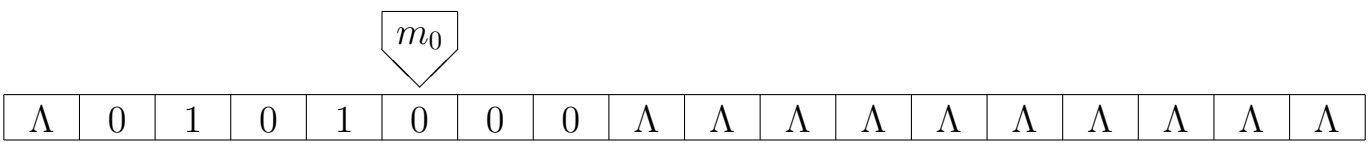
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



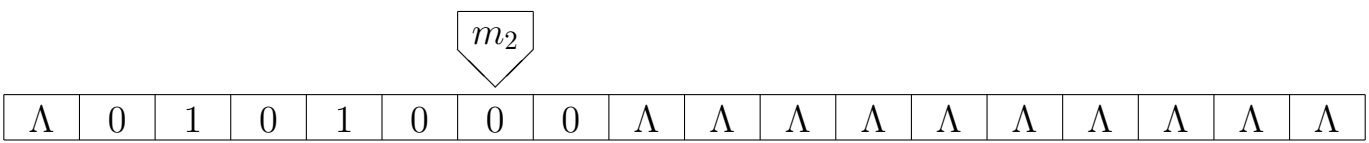
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



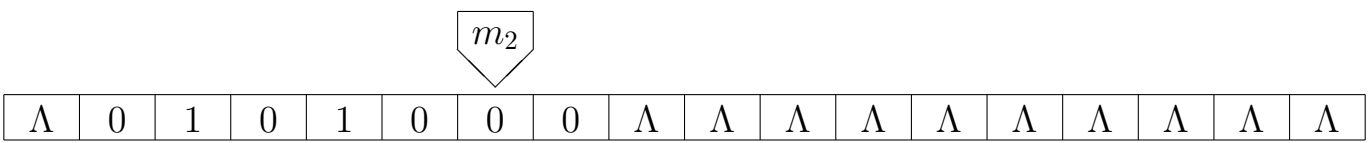
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



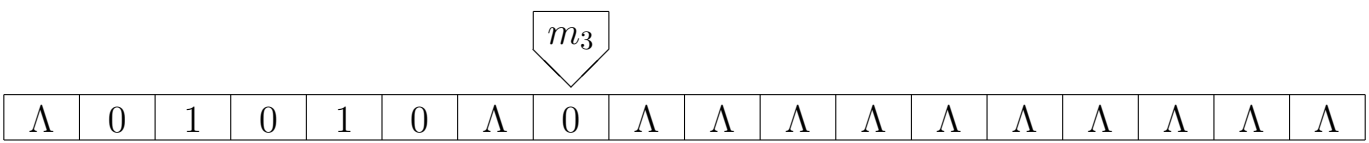
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$		$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



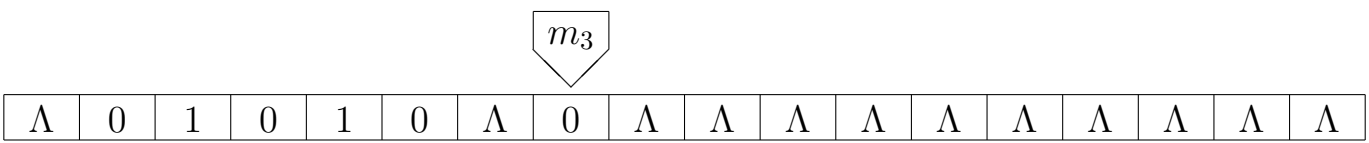
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



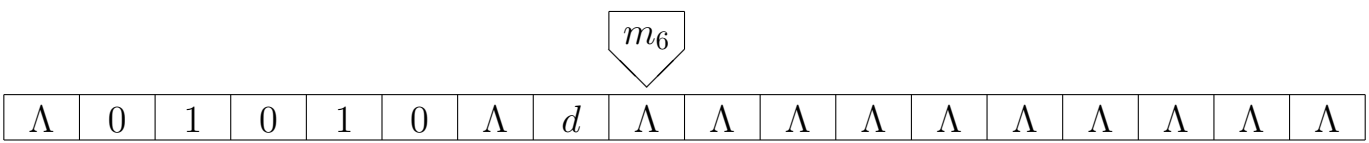
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



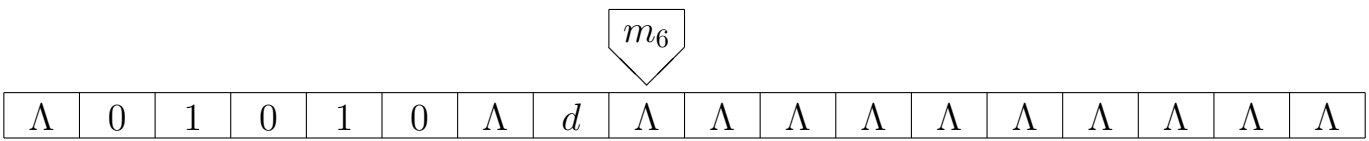
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



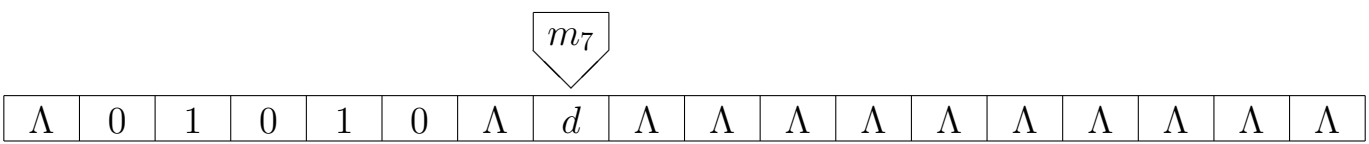
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d										



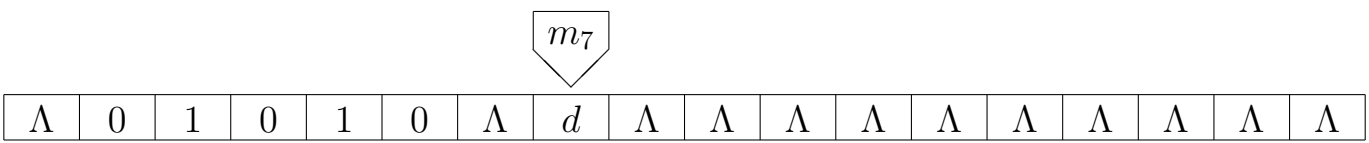
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d							$L\,(\Lambda, m_4)$			



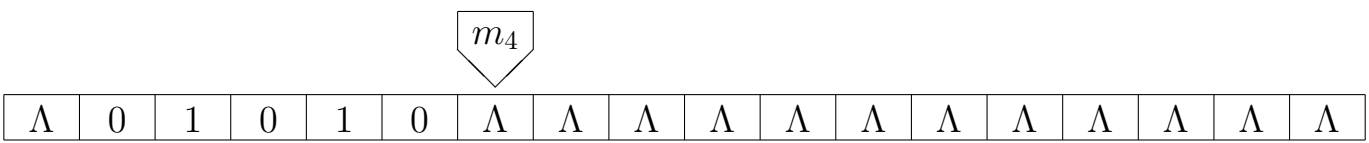
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ					$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d							$L\,(\Lambda, m_4)$			



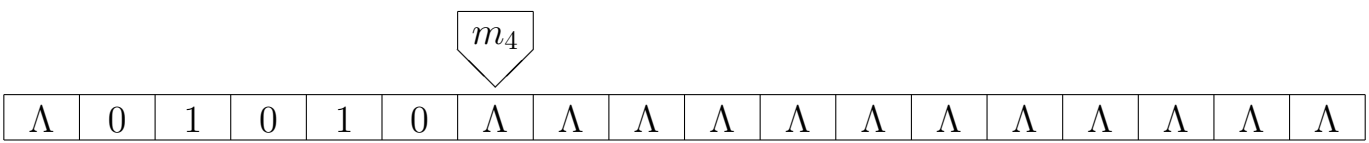
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d							$L\,(\Lambda, m_4)$			



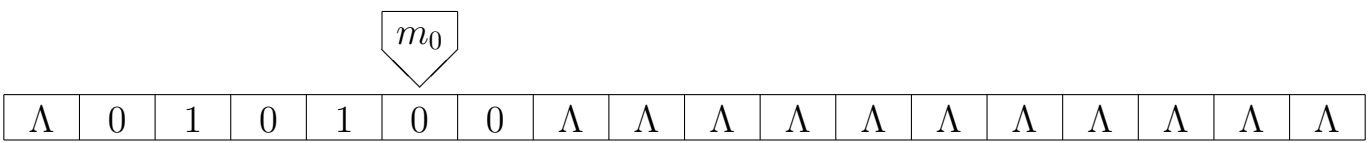
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$				
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$			
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$			
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$					
d							$L\,(\Lambda, m_4)$			



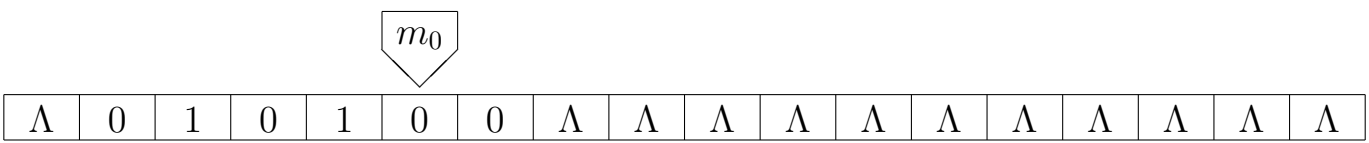
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



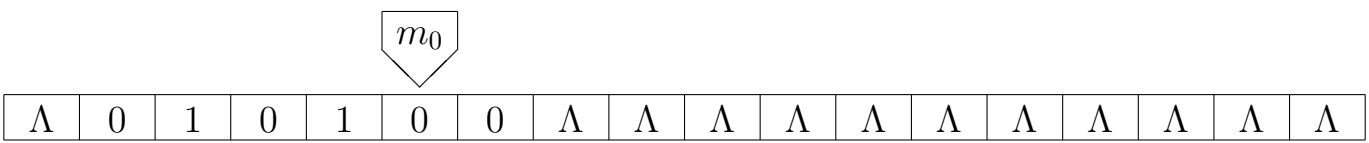
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



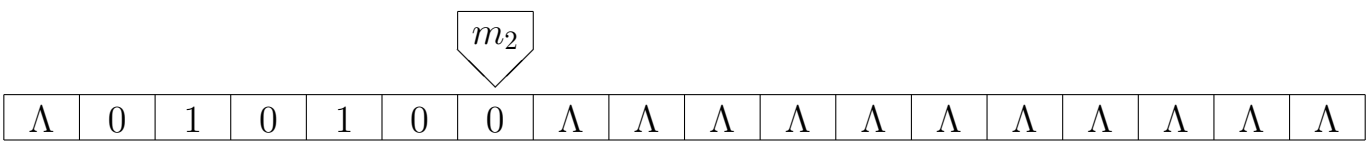
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



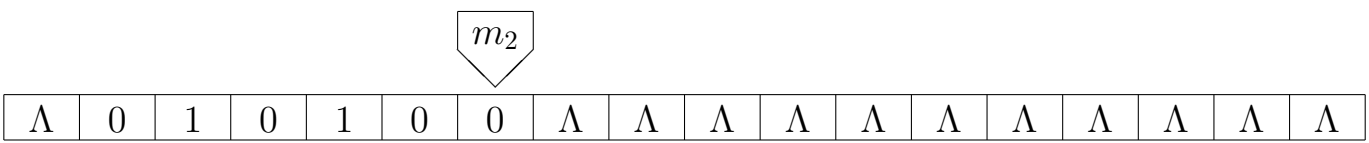
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



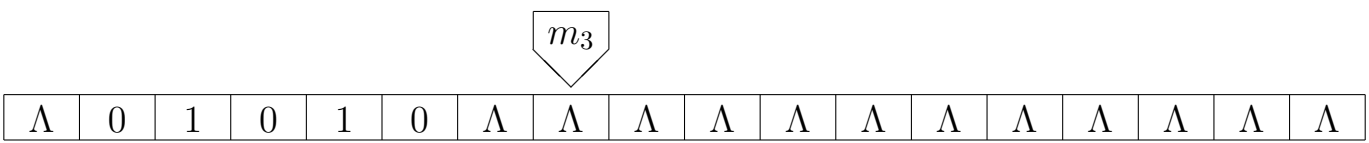
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ				$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



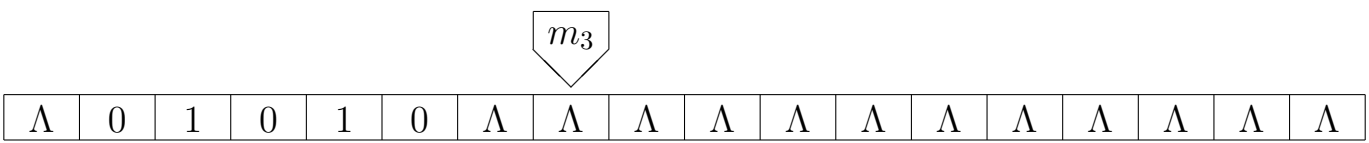
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



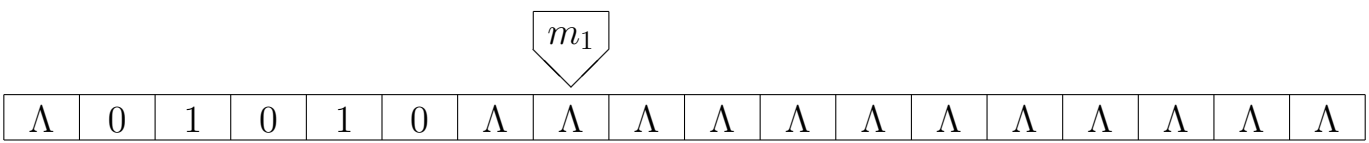
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			

Λ	1	0	0	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

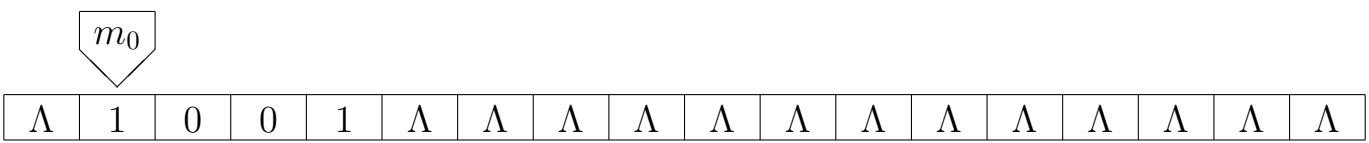
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



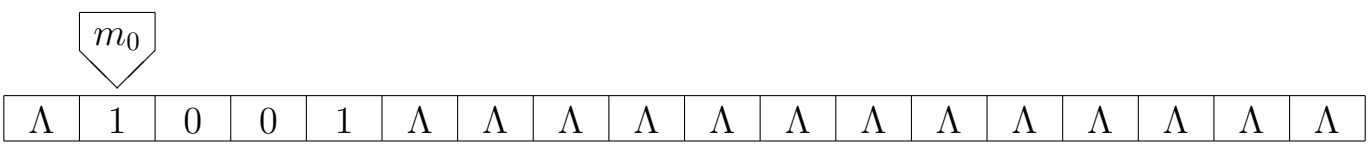
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



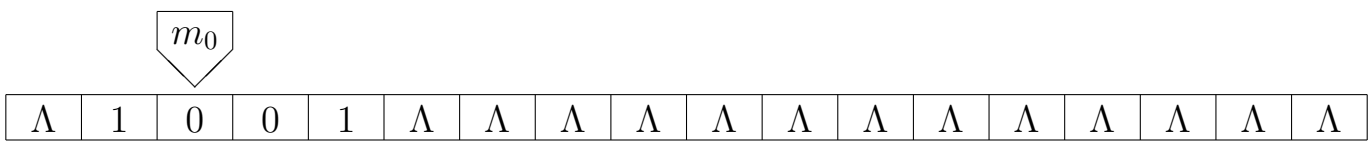
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



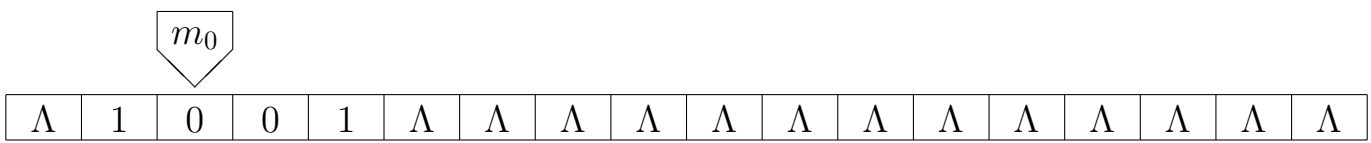
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



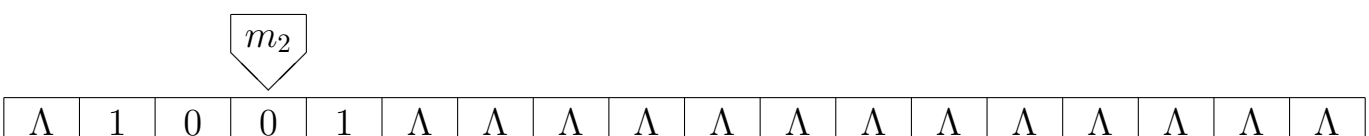
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



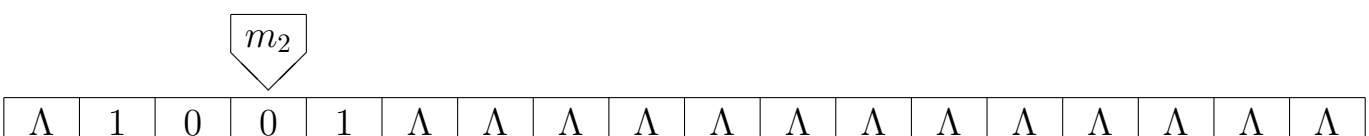
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



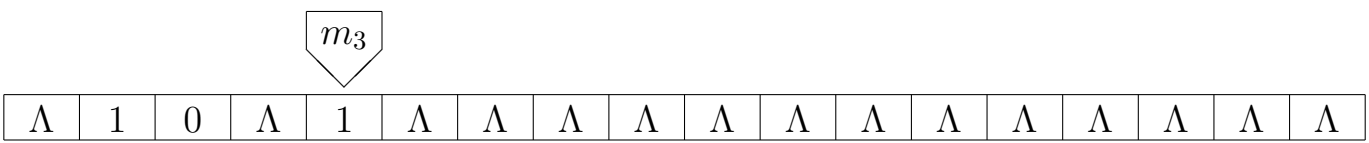
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



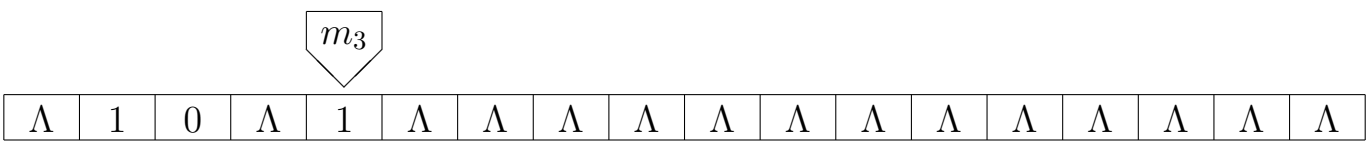
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



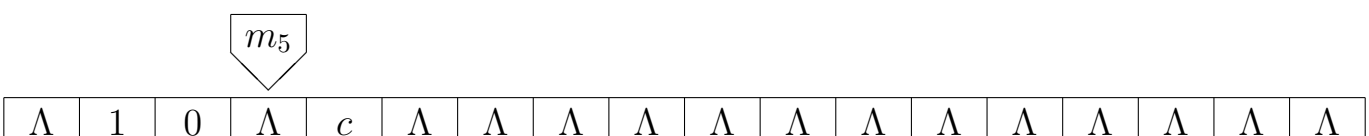
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



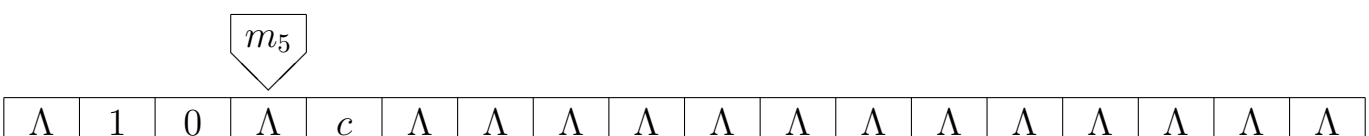
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			

Λ	1	0	1	c	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

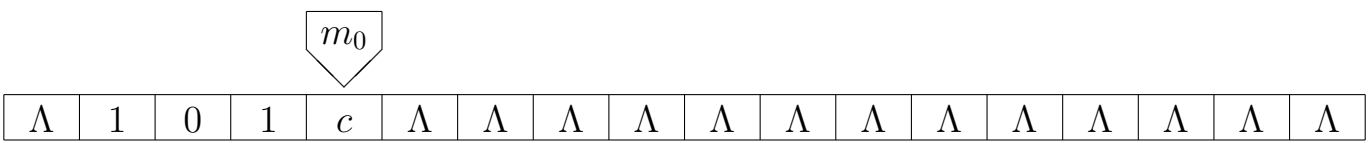
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



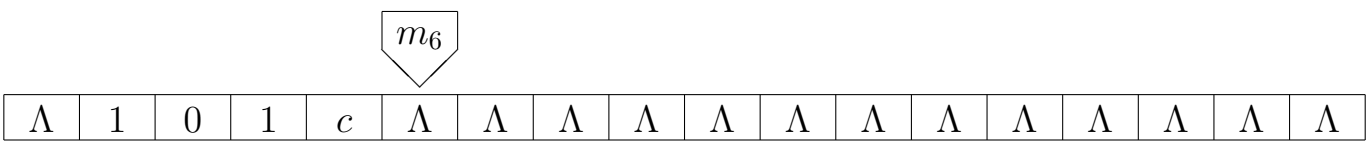
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



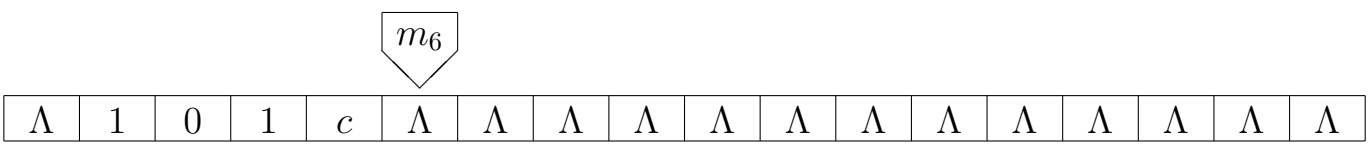
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



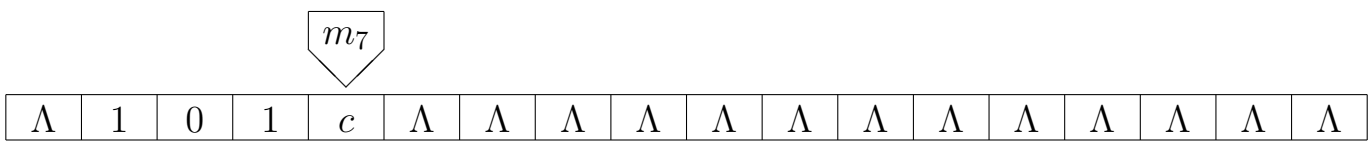
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$			(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



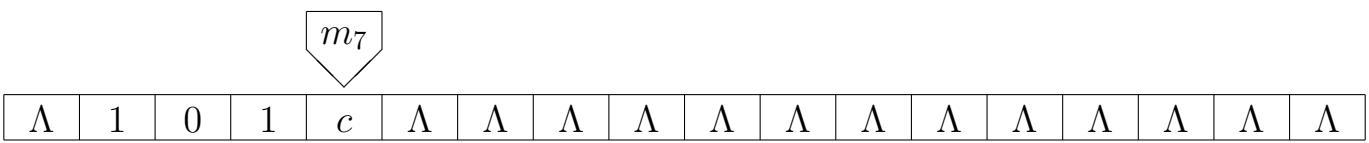
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



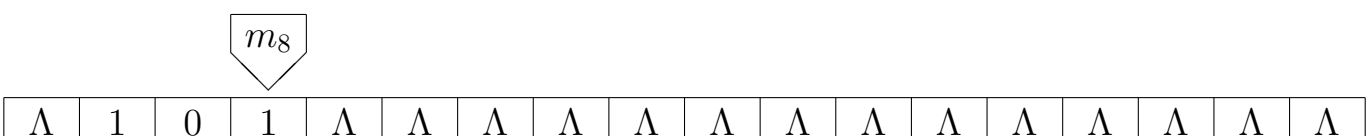
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



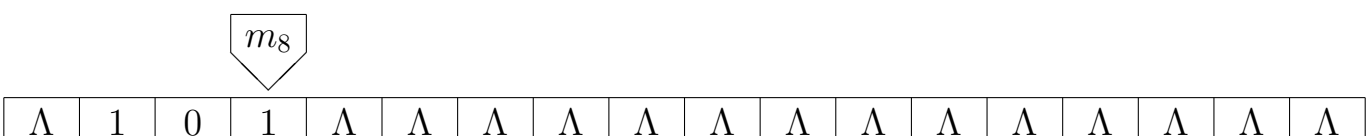
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



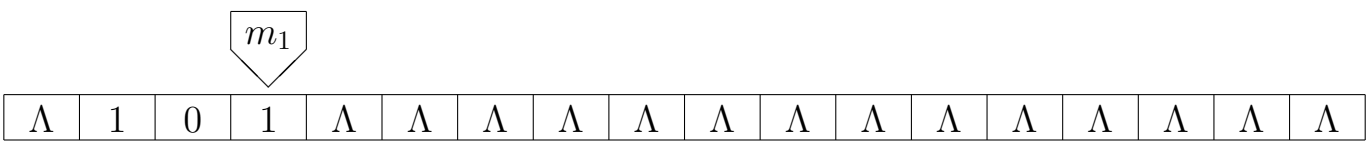
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



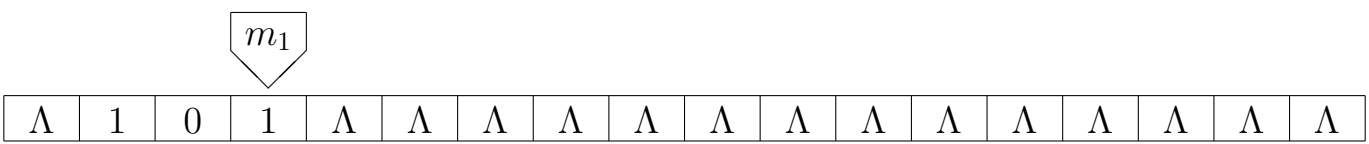
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			

Λ	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

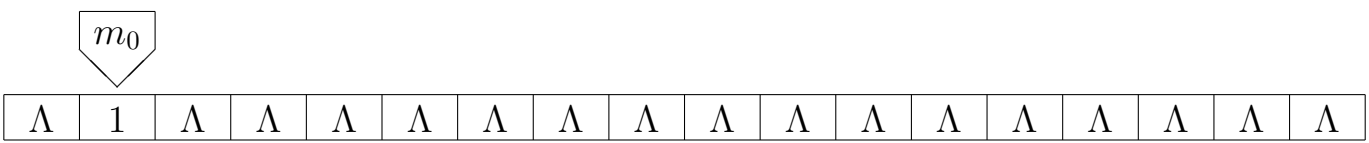
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



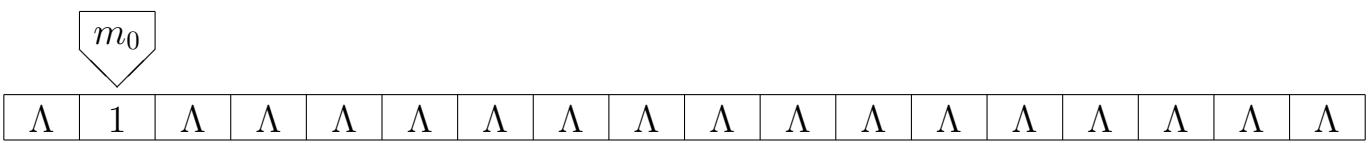
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



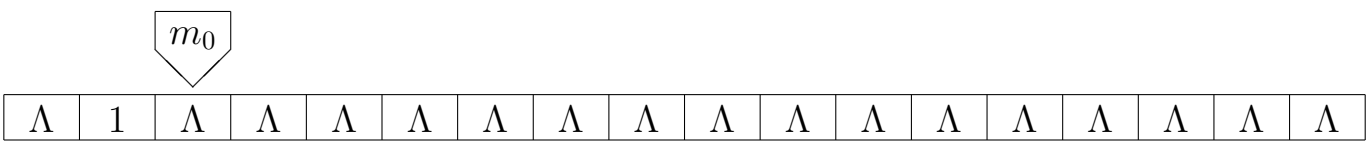
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ			(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



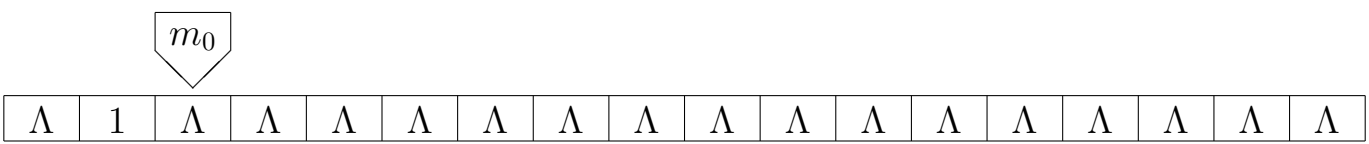
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



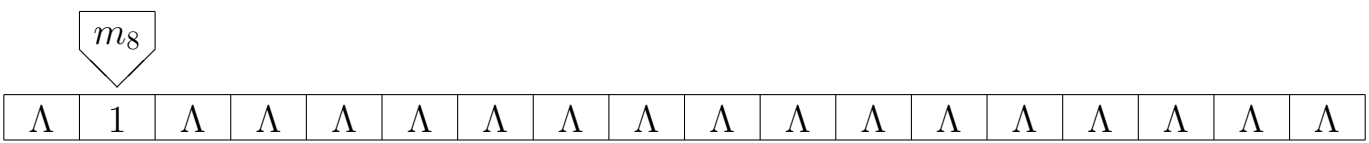
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



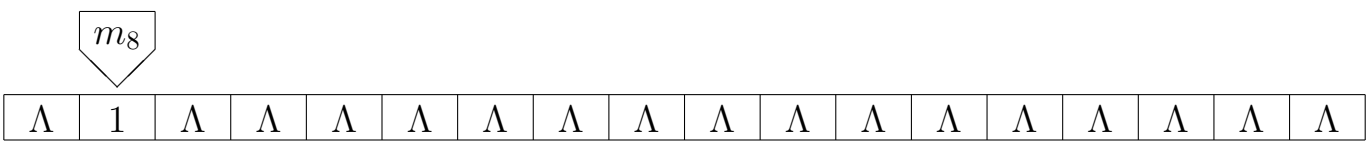
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



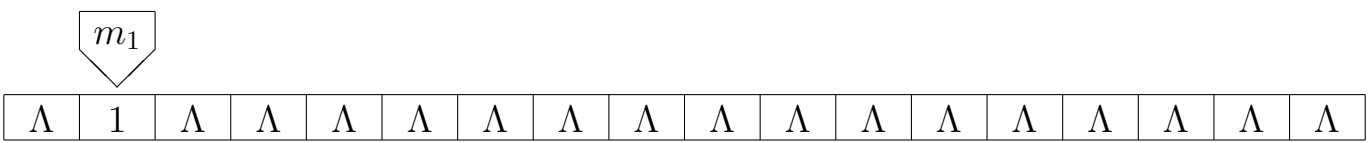
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L(\Lambda, m_8)$		(Λ, m_1)	$L(0, m_0)$	$R(1, m_0)$	$L(\Lambda, m_7)$		(Λ, m_1)		
0	$R(0, m_2)$	$R(\Lambda, m_3)$	$R(d, m_6)$	$L(0, m_4)$	$L(1, m_4)$	$R(0, m_6)$	$L(\Lambda, m_4)$	$(0, m_1)$		
1	$R(1, m_0)$	$R(1, m_0)$	$L(c, m_5)$	$L(0, m_5)$	$L(1, m_5)$	$R(1, m_6)$	$L(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L(0, m_0)$	$L(1, m_0)$		$L(\Lambda, m_8)$	(Λ, m_1)		
d							$L(\Lambda, m_4)$			



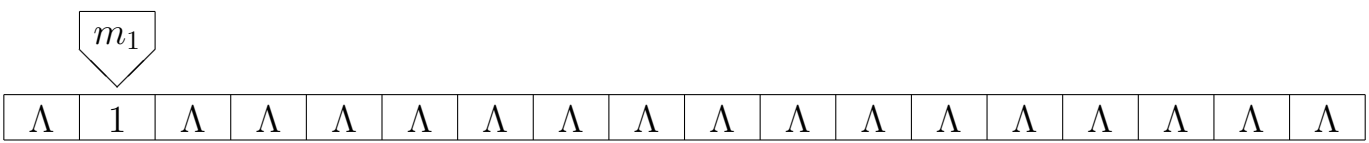
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			

Λ	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

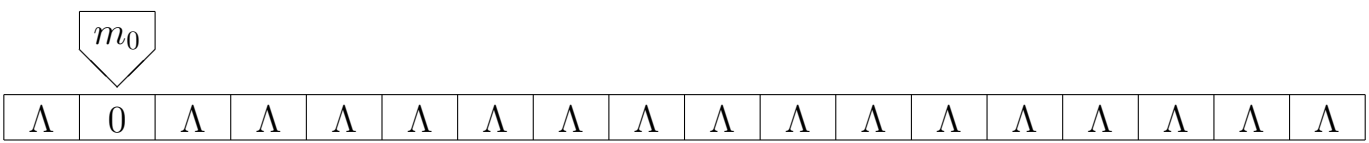
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



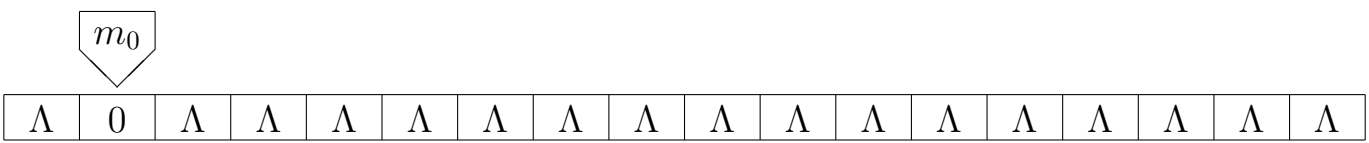
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



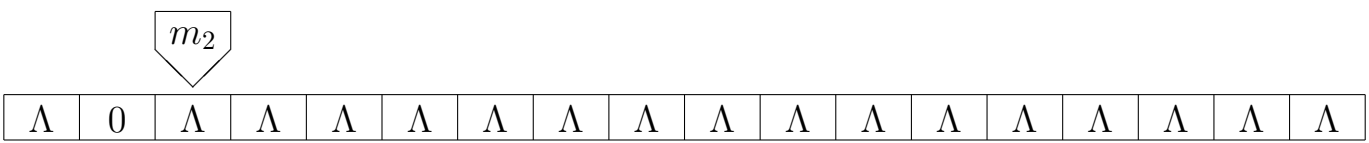
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$		(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



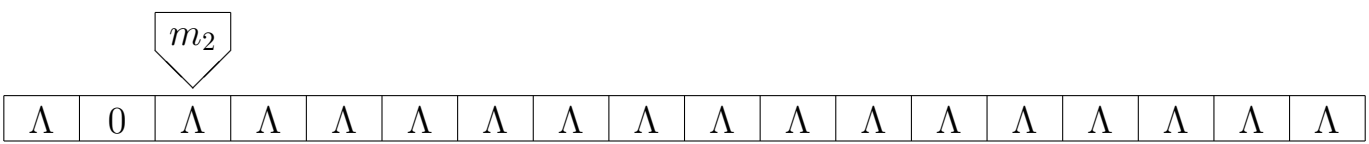
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$	(Λ, m_1)	(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



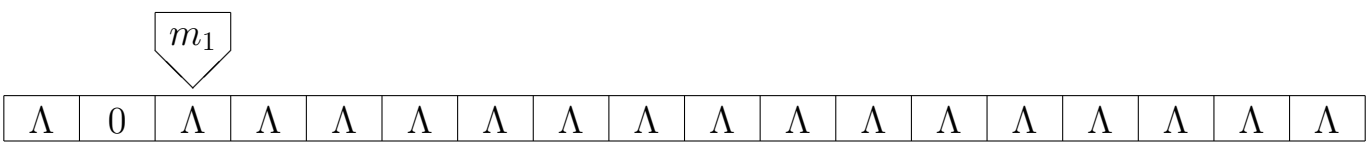
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

a)	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	$L\,(\Lambda, m_8)$	(Λ, m_1)	(Λ, m_1)	$L\,(0, m_0)$	$R\,(1, m_0)$	$L\,(\Lambda, m_7)$		(Λ, m_1)		
0	$R\,(0, m_2)$	$R\,(\Lambda, m_3)$	$R\,(d, m_6)$	$L\,(0, m_4)$	$L\,(1, m_4)$	$R\,(0, m_6)$	$L\,(\Lambda, m_4)$	$(0, m_1)$		
1	$R\,(1, m_0)$	$R\,(1, m_0)$	$L\,(c, m_5)$	$L\,(0, m_5)$	$L\,(1, m_5)$	$R\,(1, m_6)$	$L\,(\Lambda, m_5)$	$(1, m_1)$		
c	$R(c, m_6)$			$L\,(0, m_0)$	$L\,(1, m_0)$		$L\,(\Lambda, m_8)$	(Λ, m_1)		
d							$L\,(\Lambda, m_4)$			



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										

Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										

Λ	0	1	0	0	1	0	0	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

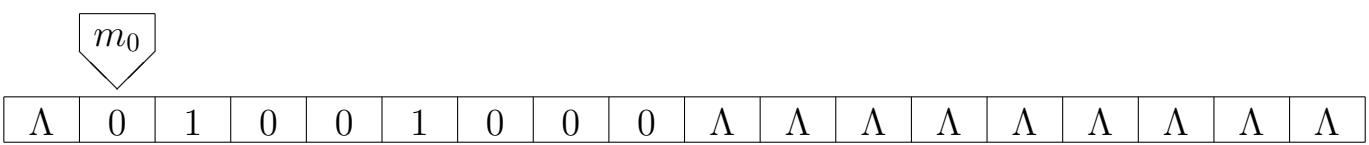
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0										
1										



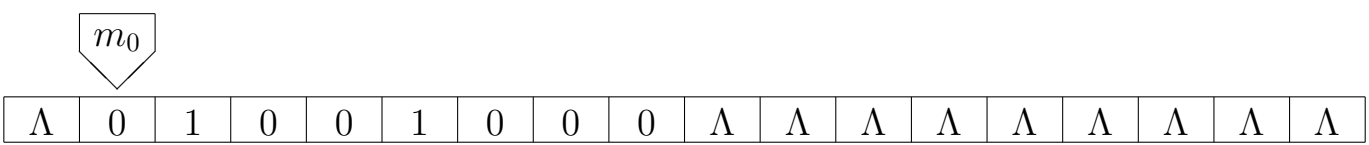
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$									
1										



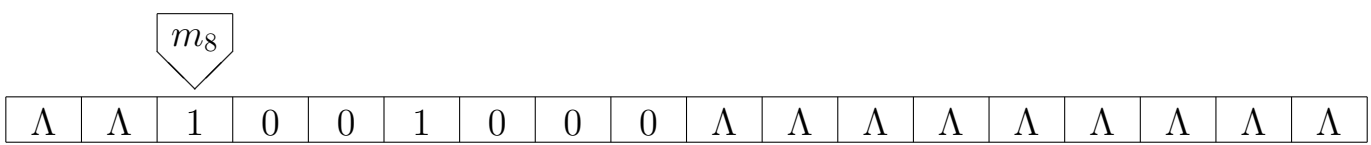
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$									
1										



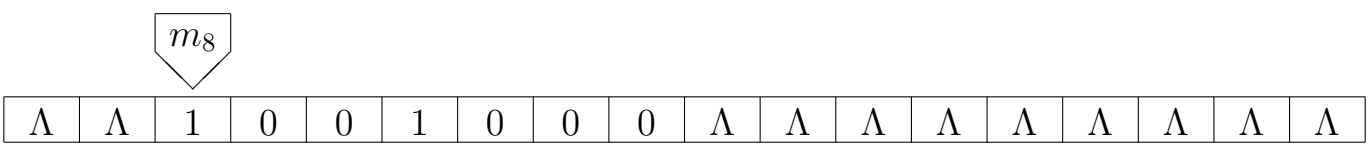
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$									
1								$R(1, m_2)$		



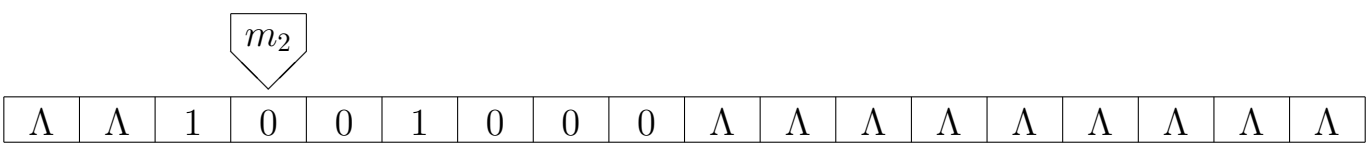
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$									
1								$R(1, m_2)$		



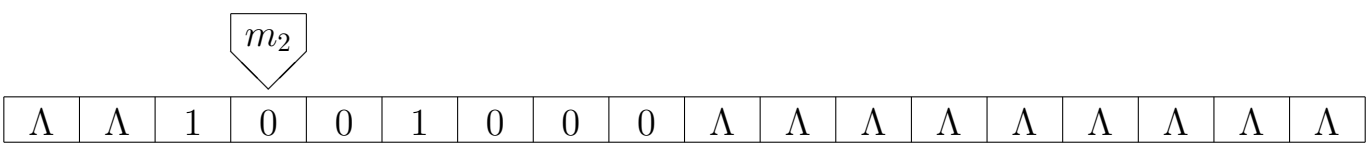
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1								$R(1, m_2)$		



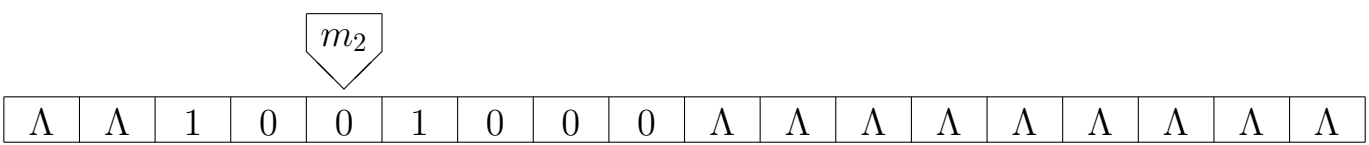
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1								$R(1, m_2)$		



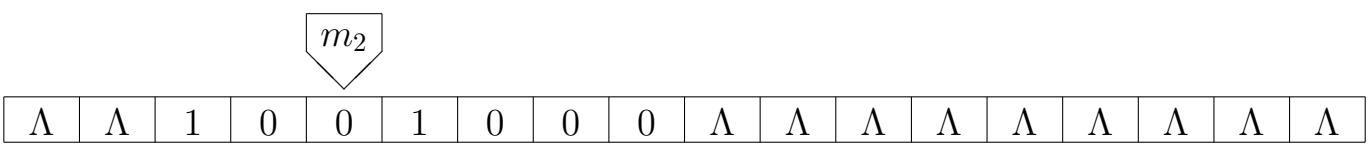
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1								$R(1, m_2)$		



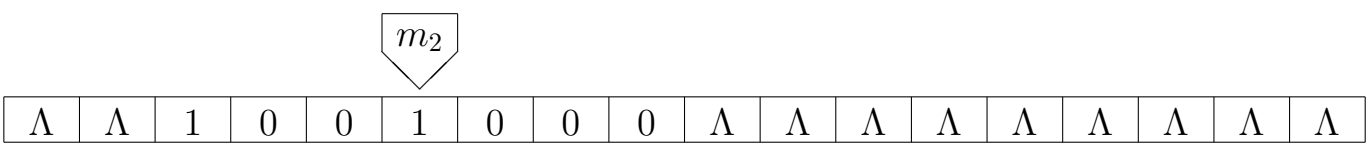
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1								$R(1, m_2)$		



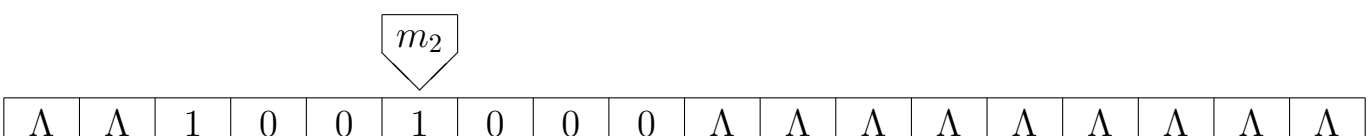
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



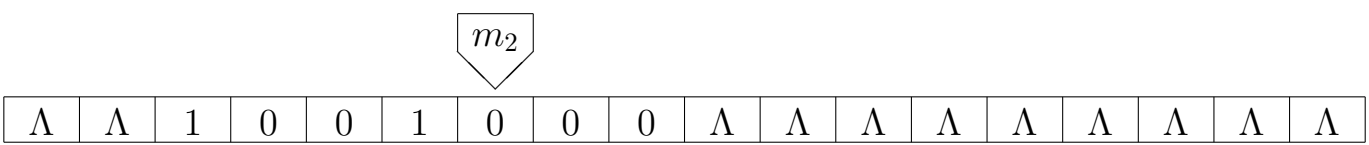
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



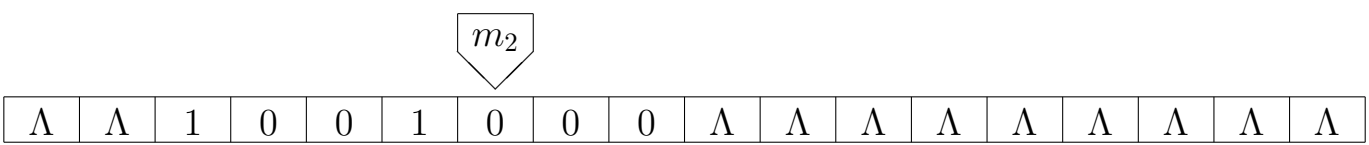
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



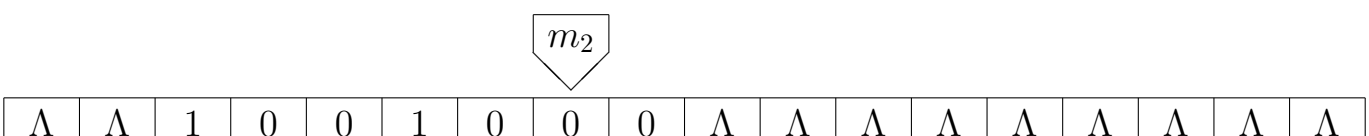
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



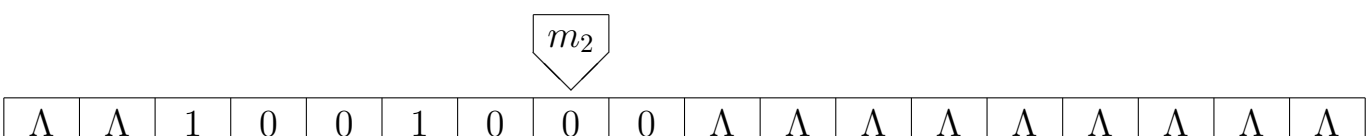
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



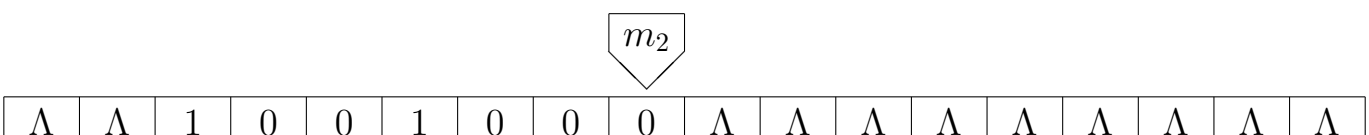
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



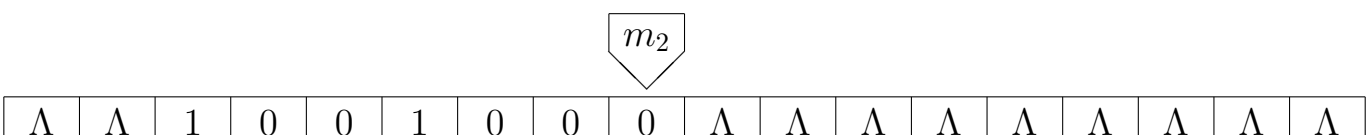
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



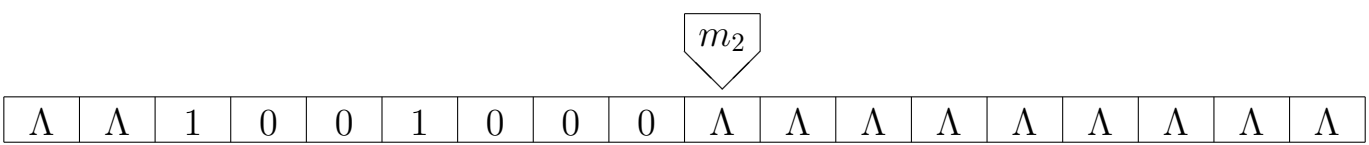
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ										
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



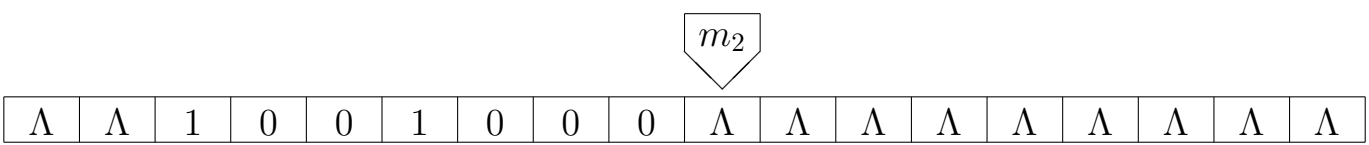
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$								
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



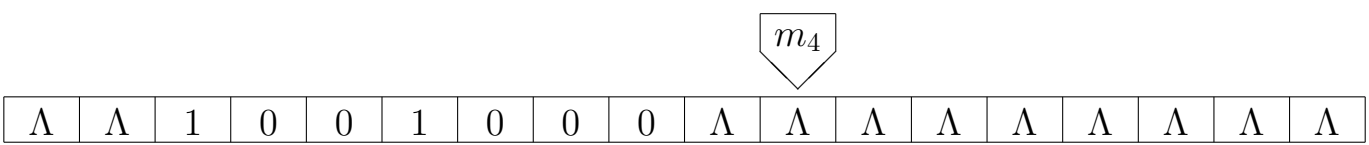
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$								
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



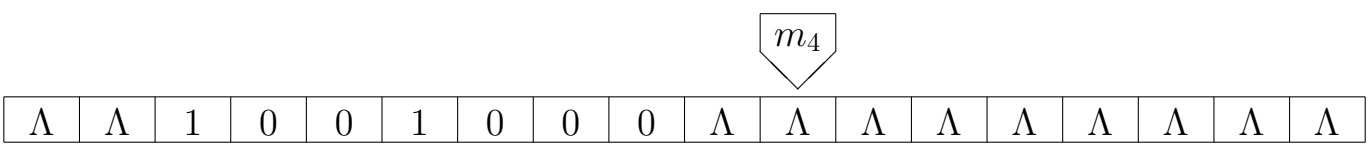
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$						
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



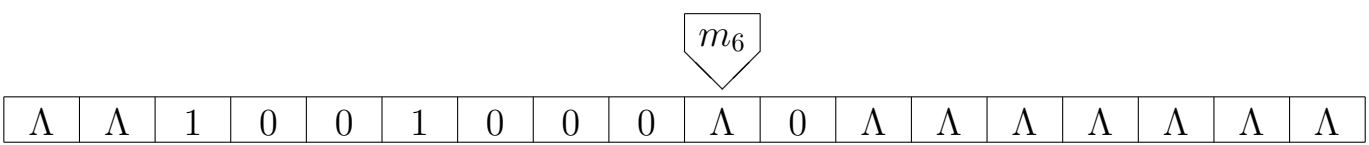
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$						
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



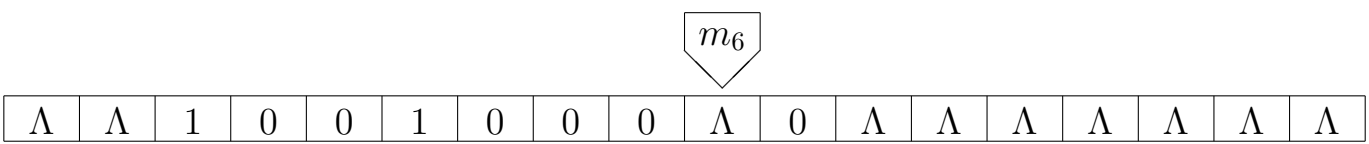
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



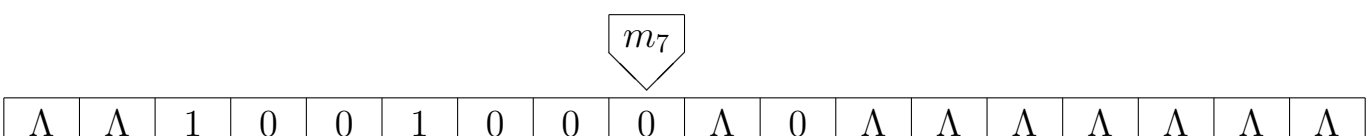
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$								
1		$R(1, m_2)$						$R(1, m_2)$		



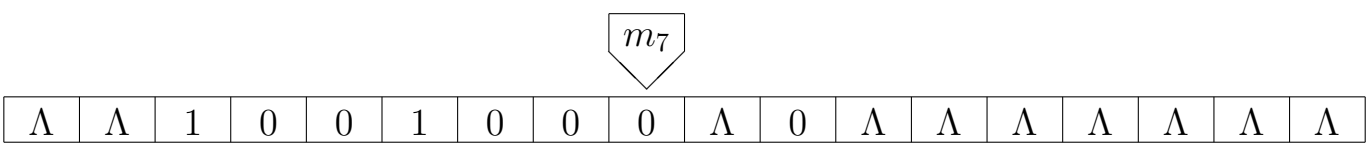
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



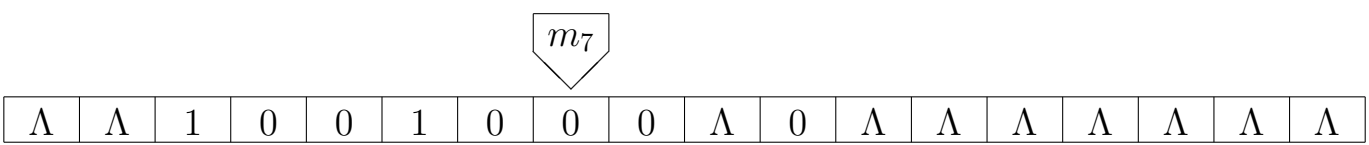
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



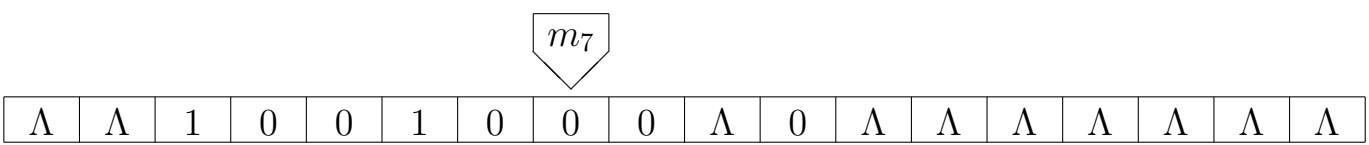
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



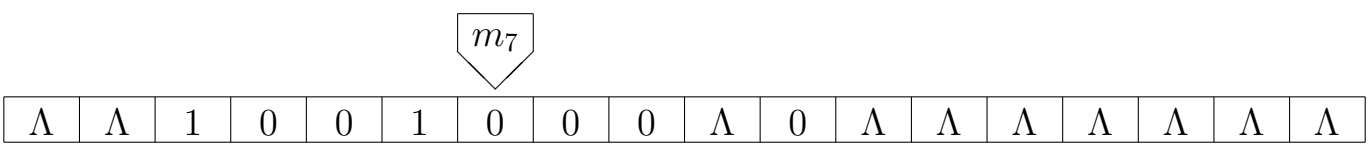
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



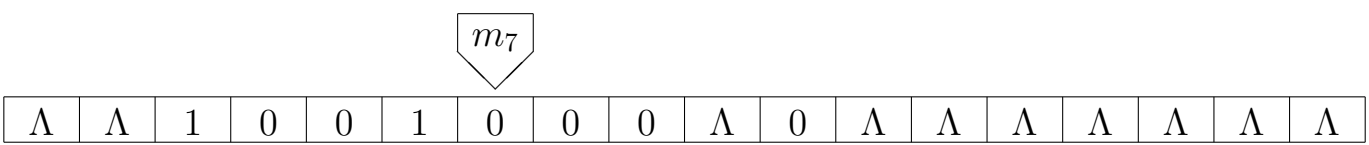
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



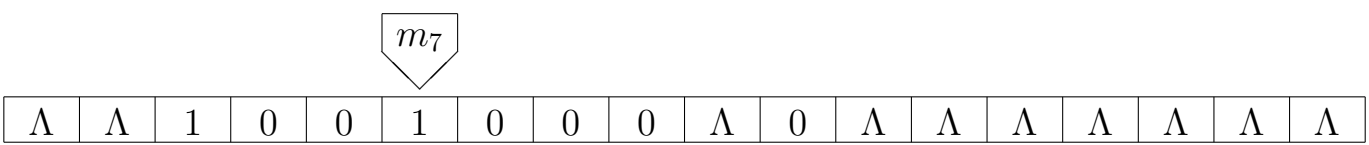
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$						$R(1, m_2)$		



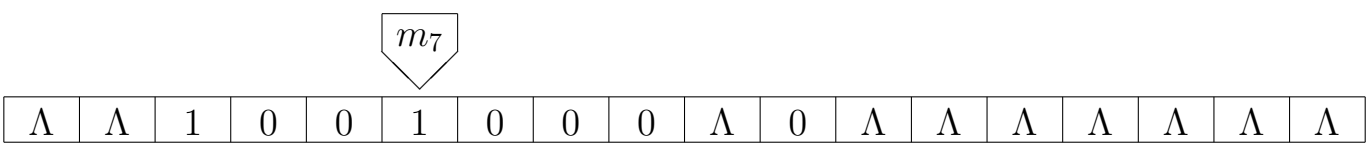
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



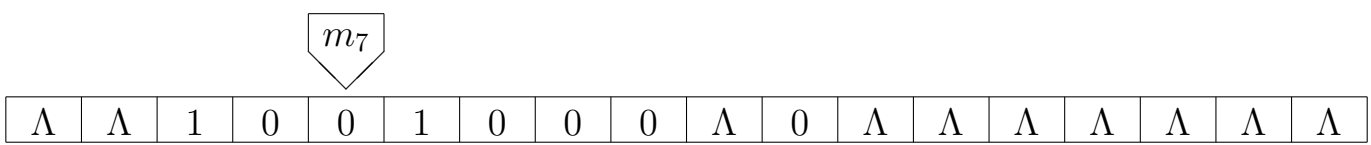
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



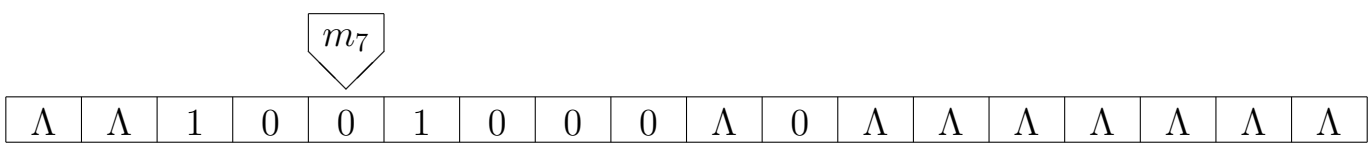
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



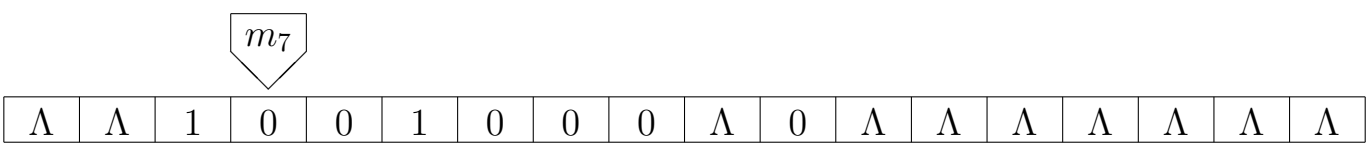
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



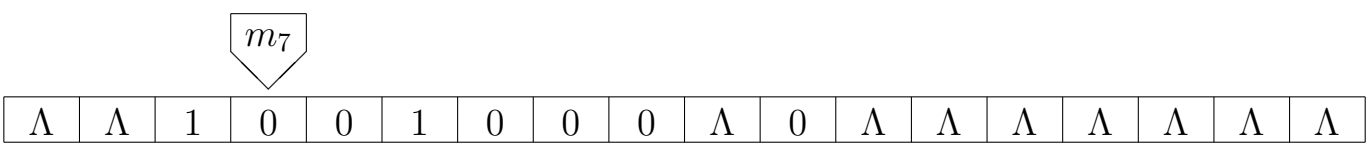
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



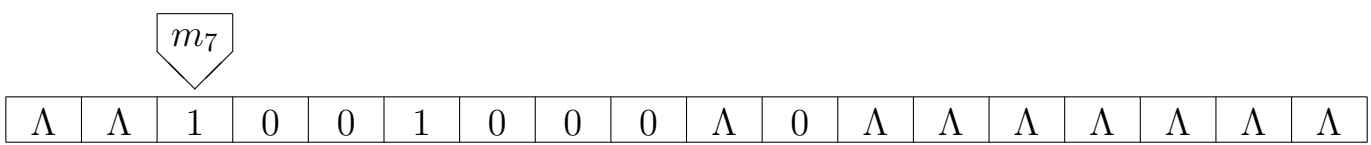
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



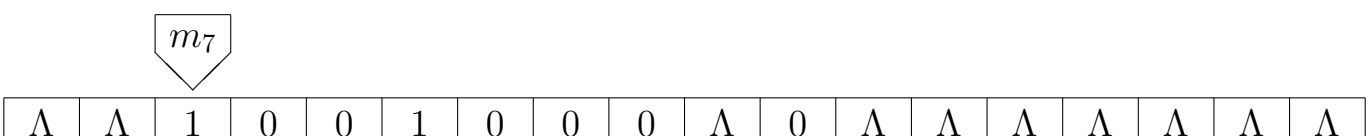
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



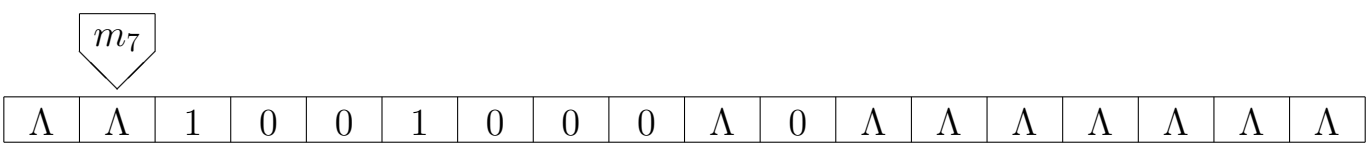
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$				
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



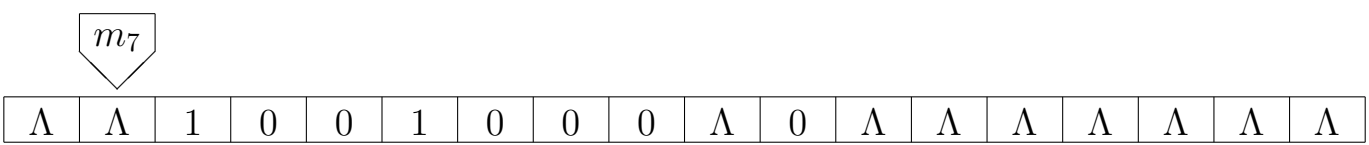
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



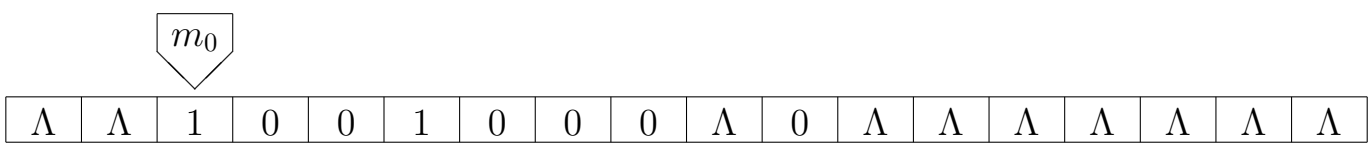
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1		$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



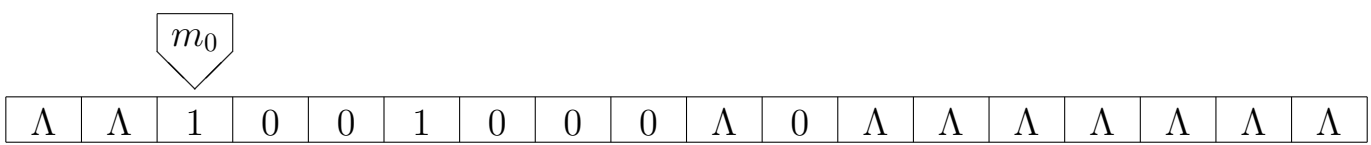
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



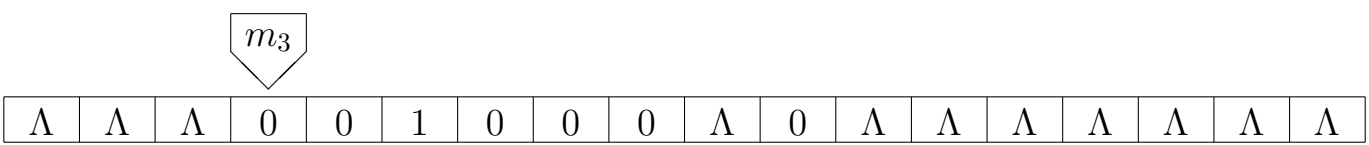
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$					$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



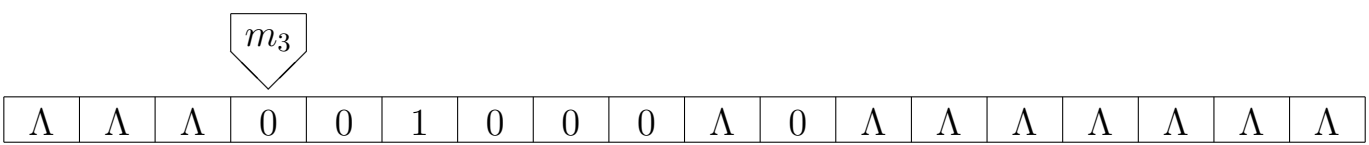
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



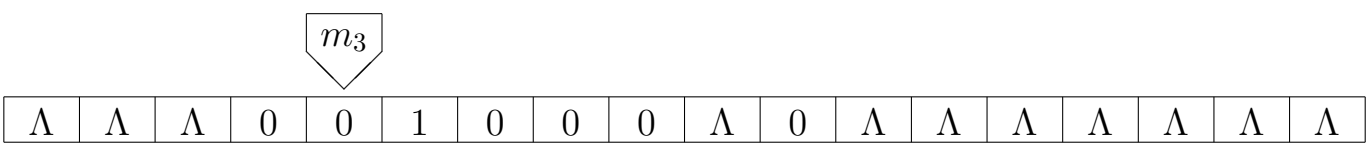
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



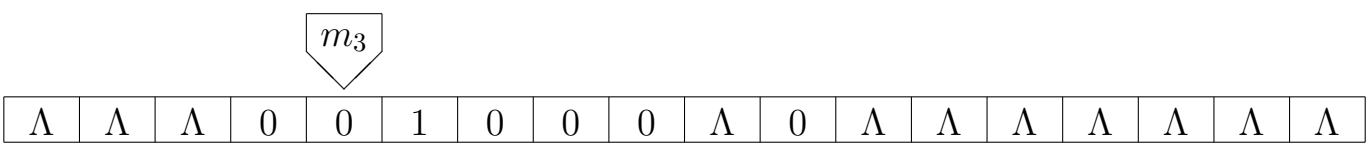
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



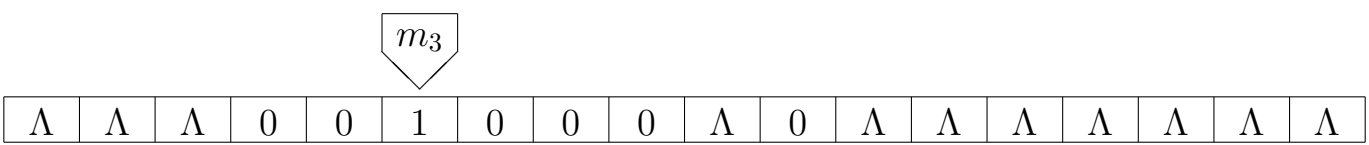
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$					$L(1, m_7)$	$R(1, m_2)$		



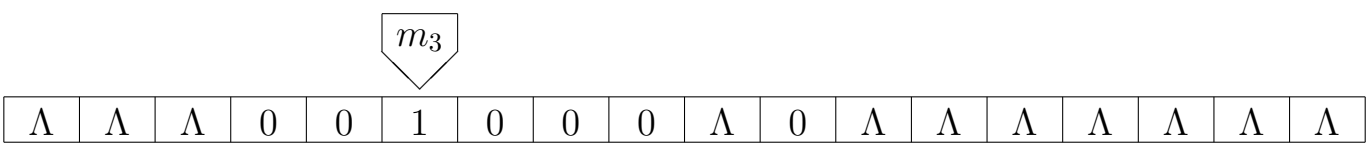
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



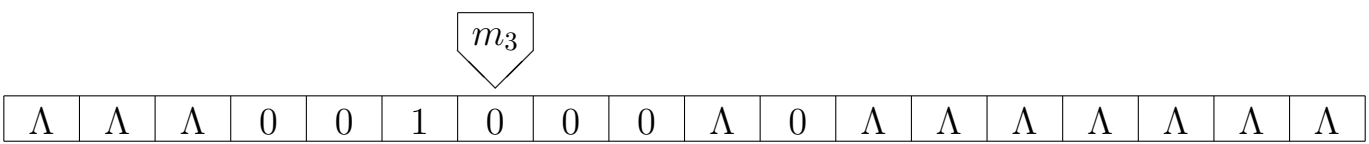
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



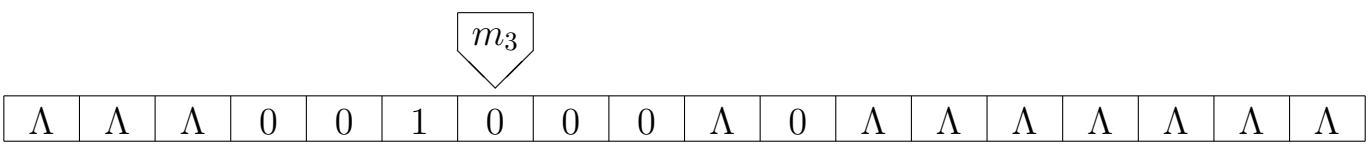
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



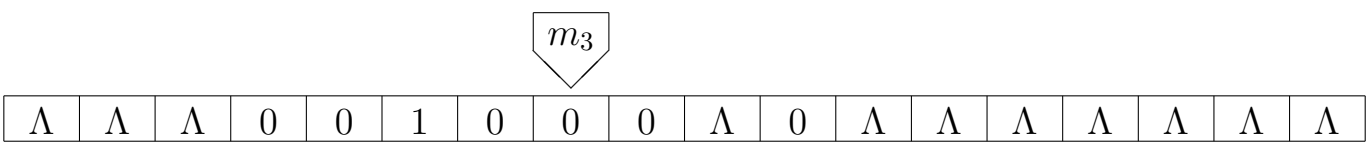
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



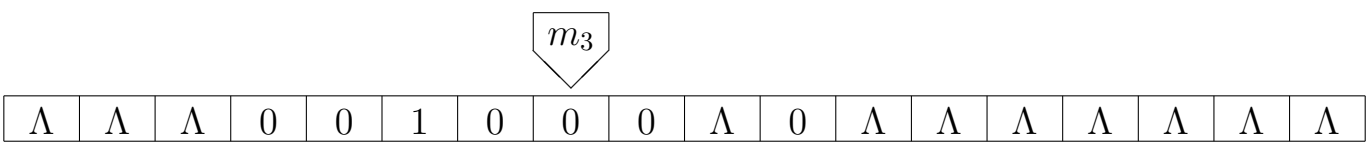
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



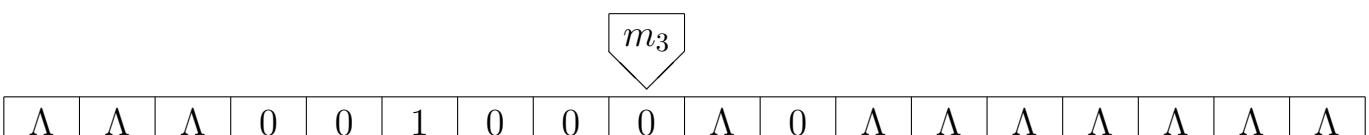
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



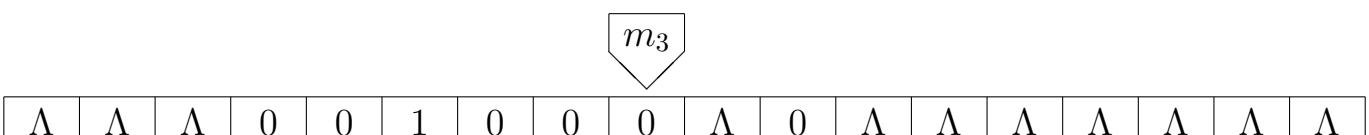
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



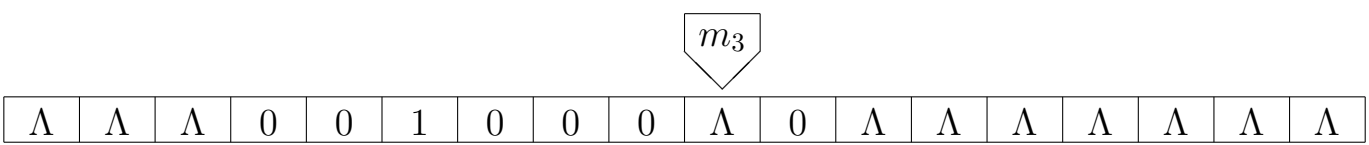
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$		$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



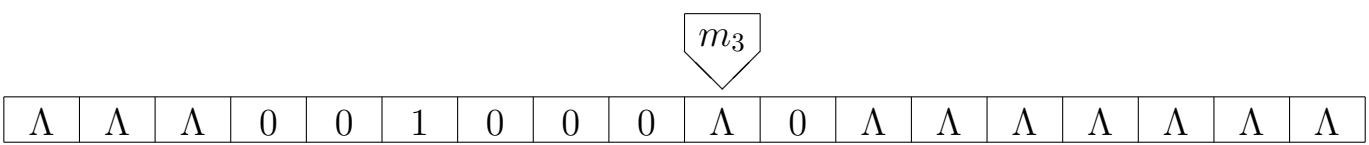
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



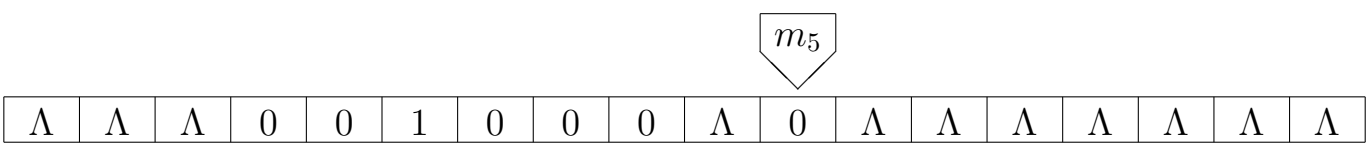
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$				$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



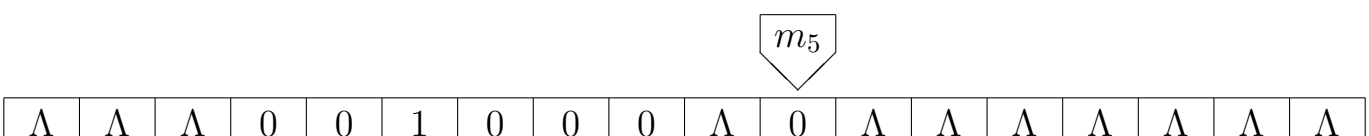
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$		$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



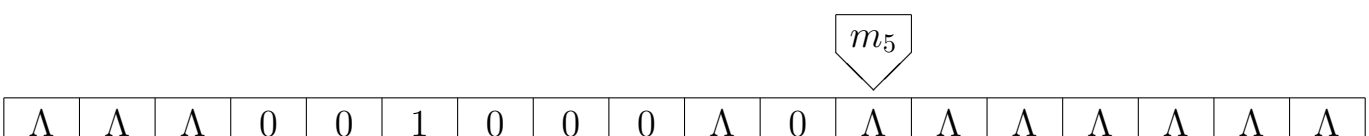
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$		$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$		$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



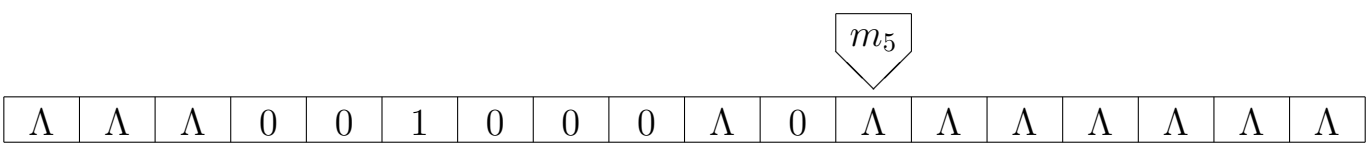
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$		$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



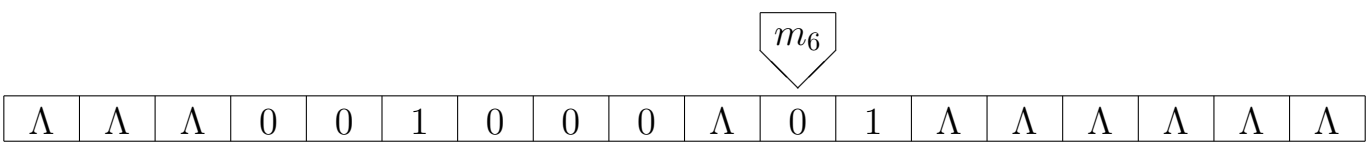
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$		$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



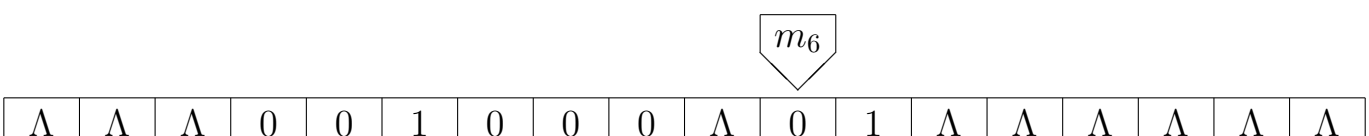
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



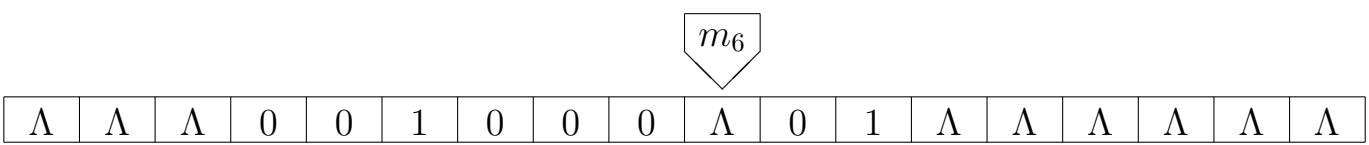
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



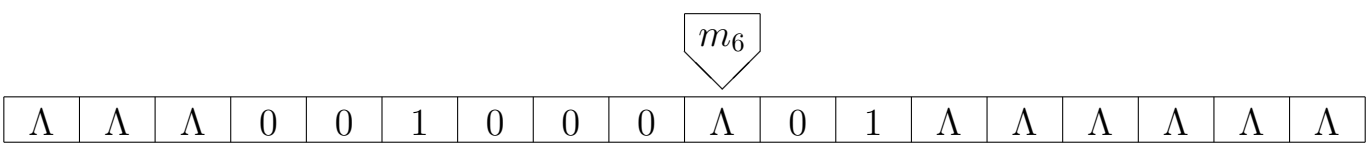
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



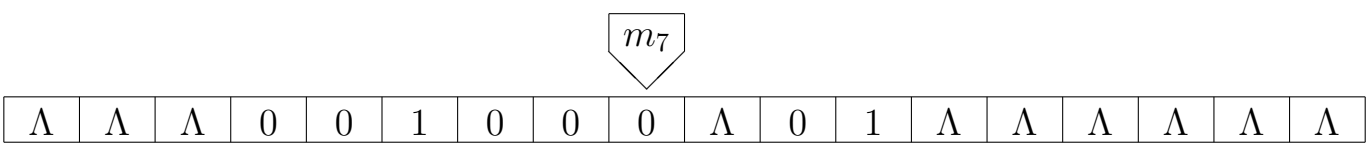
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



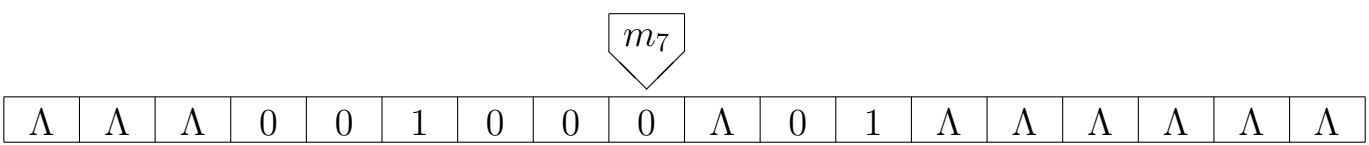
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



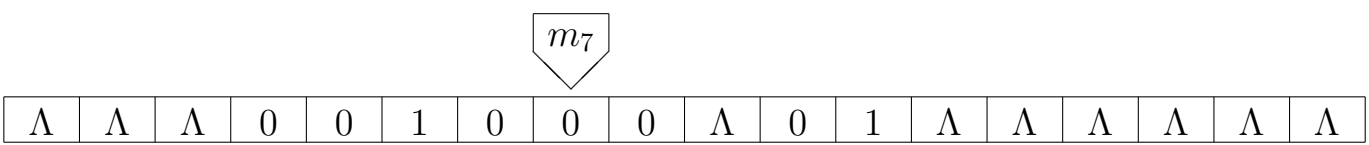
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



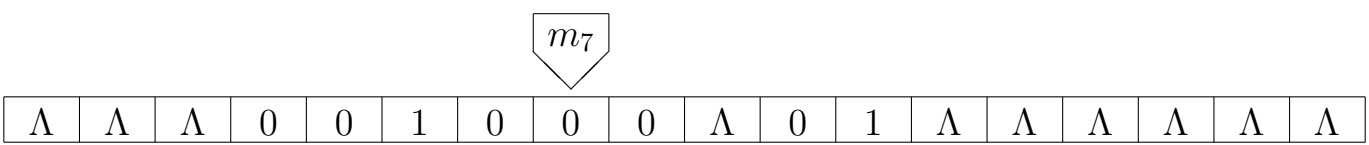
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



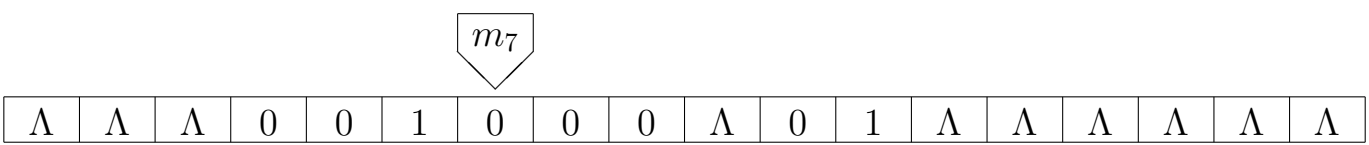
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



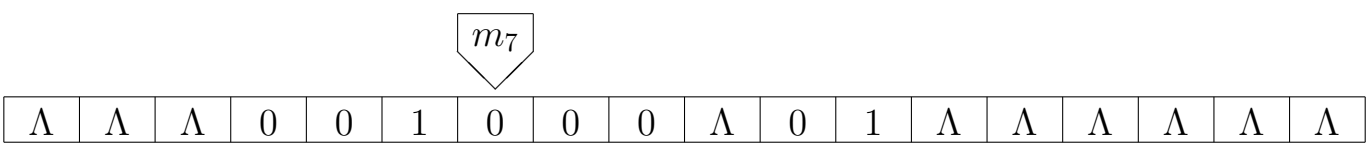
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



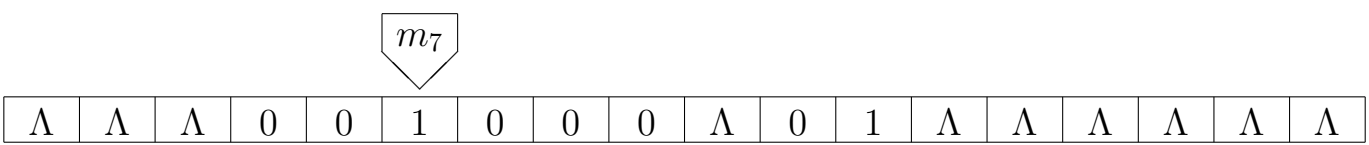
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



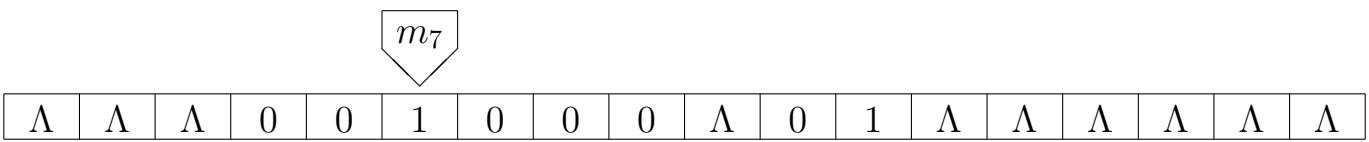
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



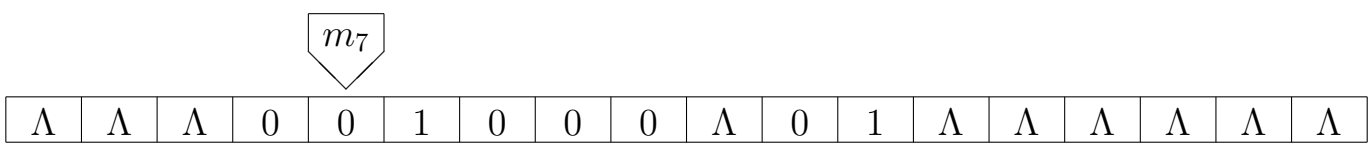
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



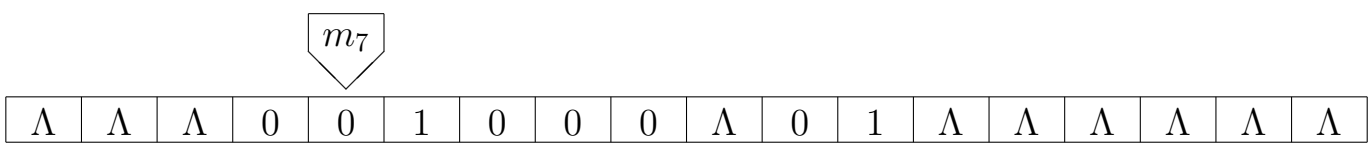
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



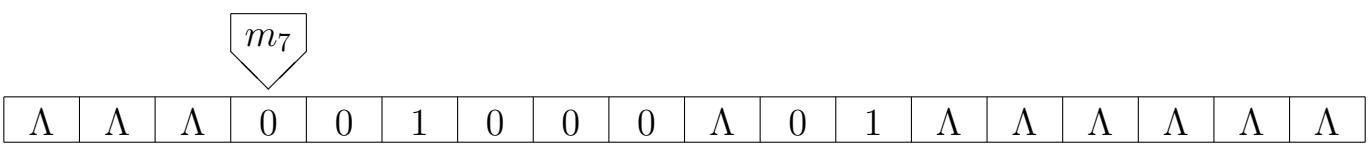
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



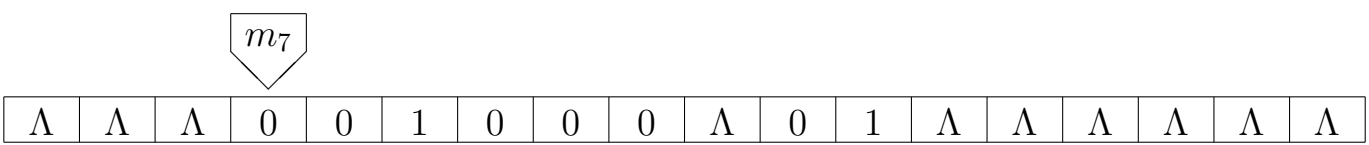
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



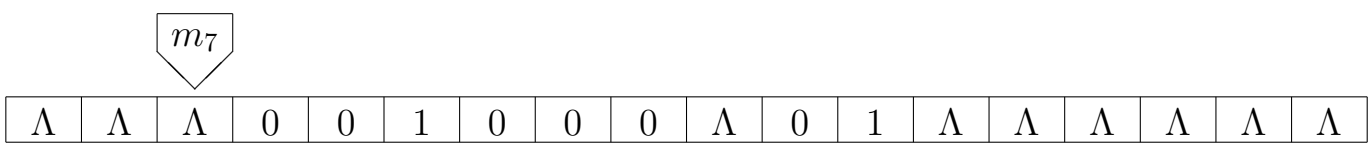
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



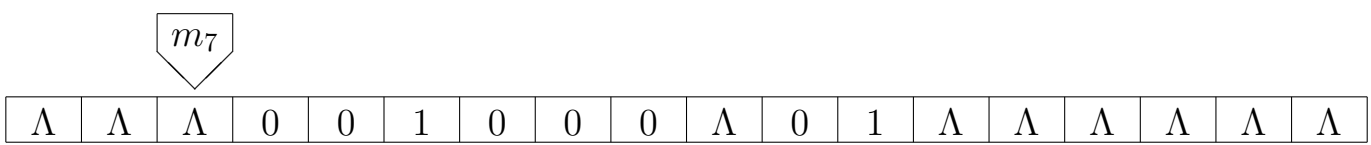
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



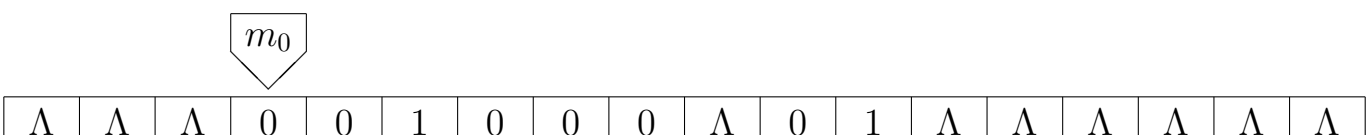
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



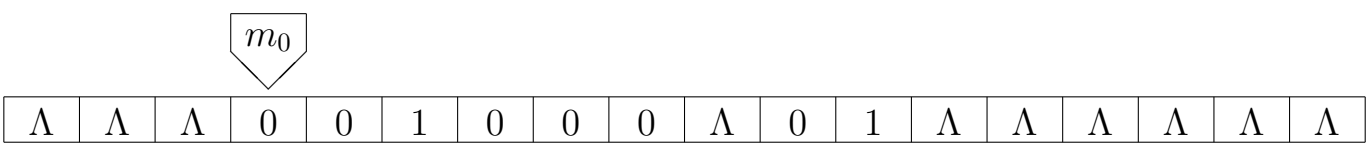
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



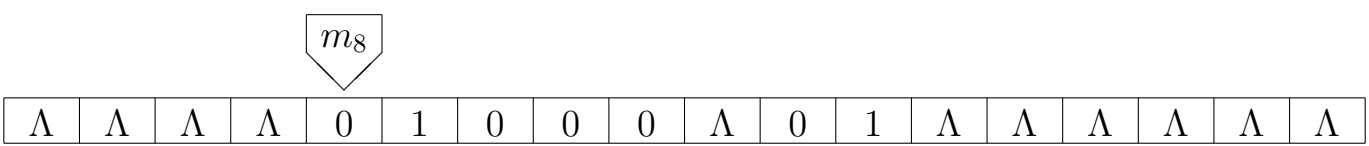
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$			
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



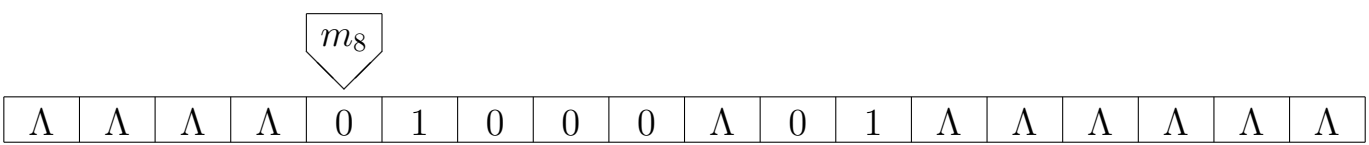
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



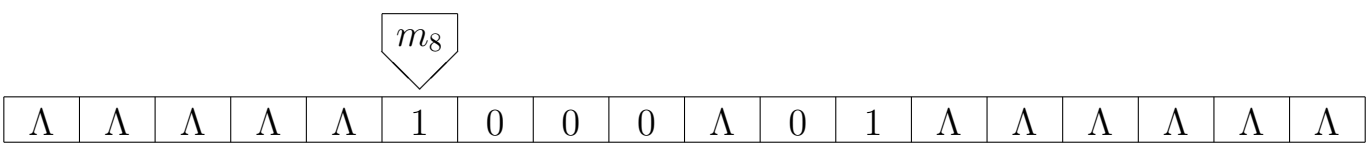
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



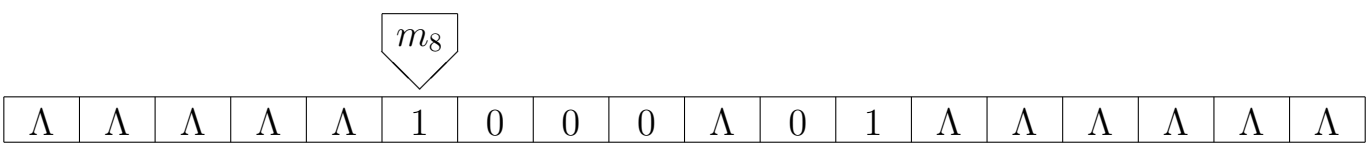
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



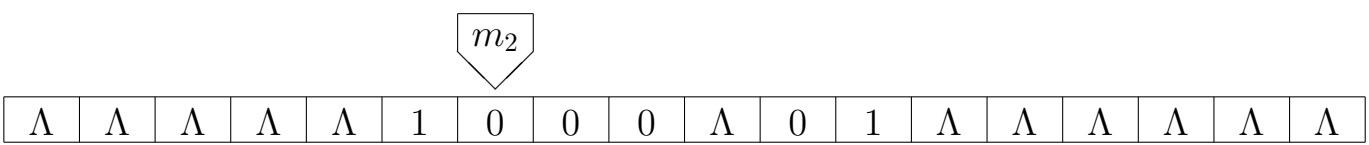
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



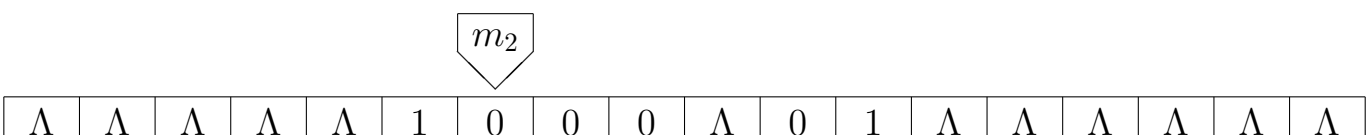
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



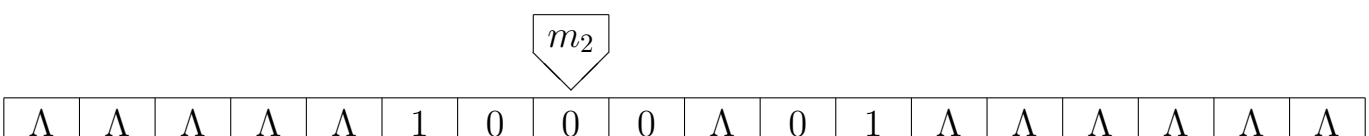
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



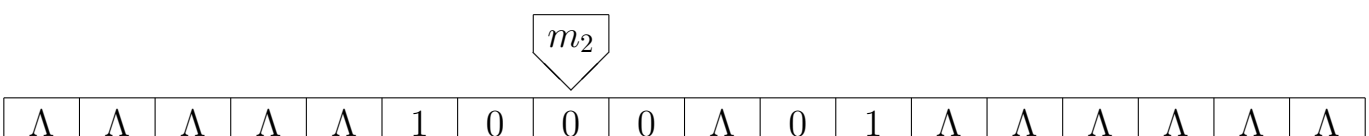
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



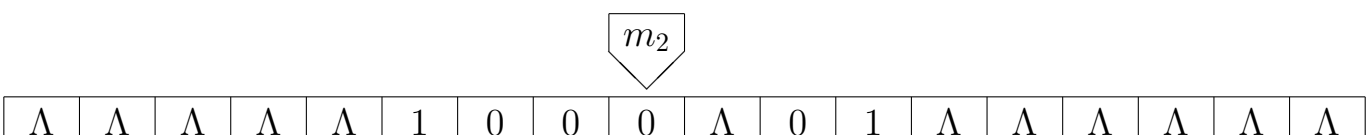
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



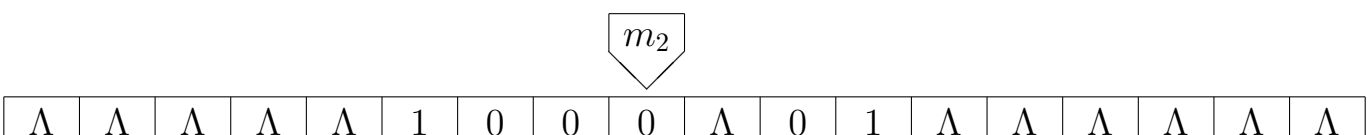
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



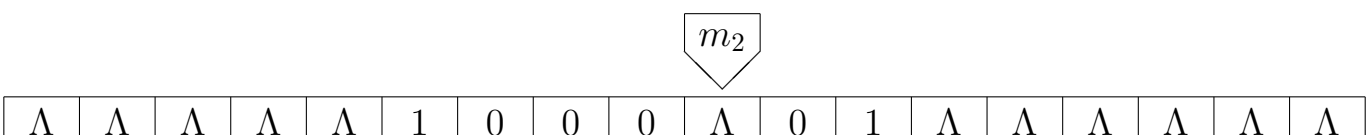
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



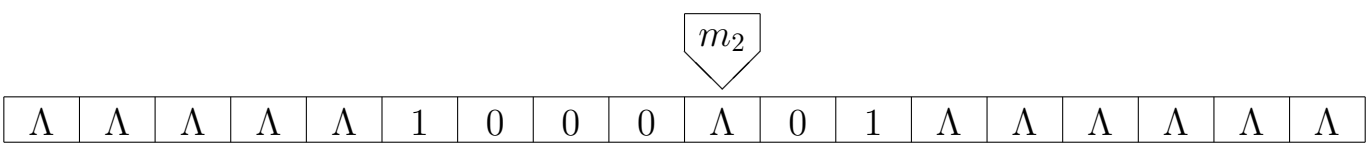
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



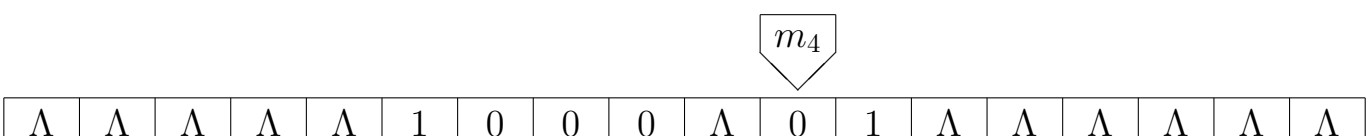
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$		$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



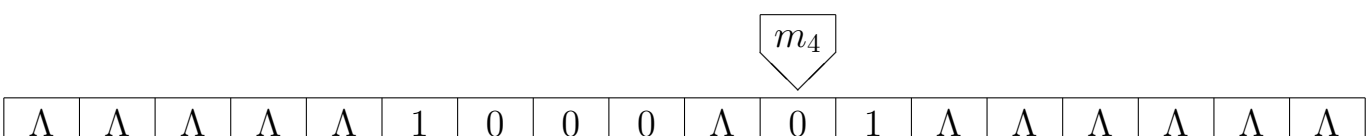
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



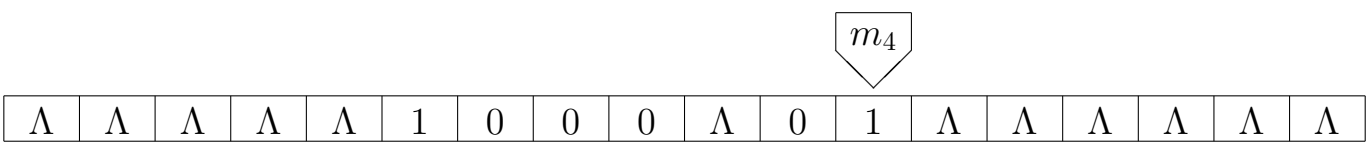
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$				$L(1, m_7)$	$R(1, m_2)$		



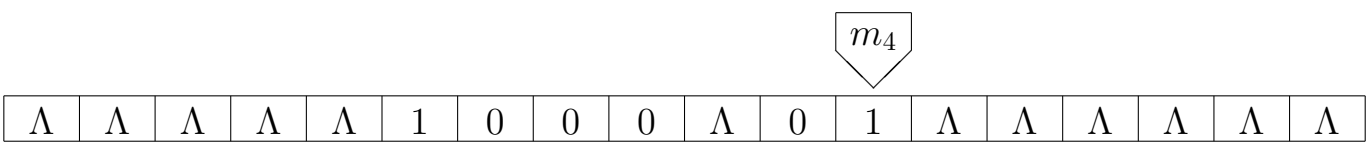
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$			$L(1, m_7)$	$R(1, m_2)$		



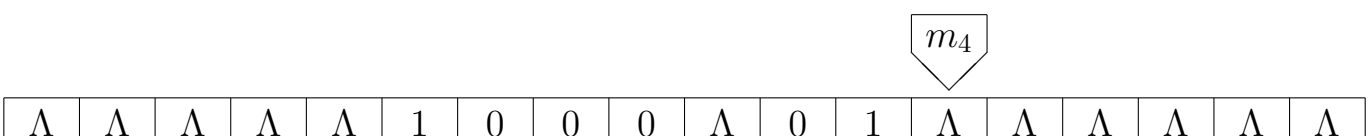
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$			$L(1, m_7)$	$R(1, m_2)$		



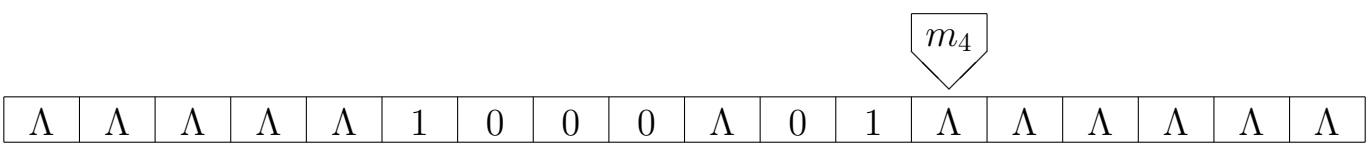
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$			$L(1, m_7)$	$R(1, m_2)$		



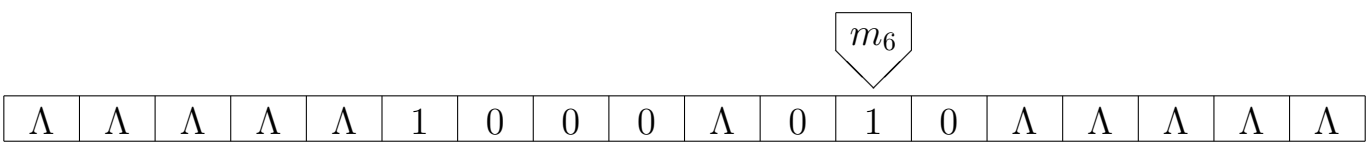
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$			$L(1, m_7)$	$R(1, m_2)$		



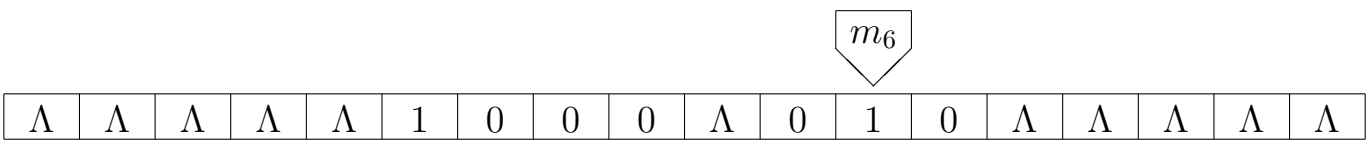
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



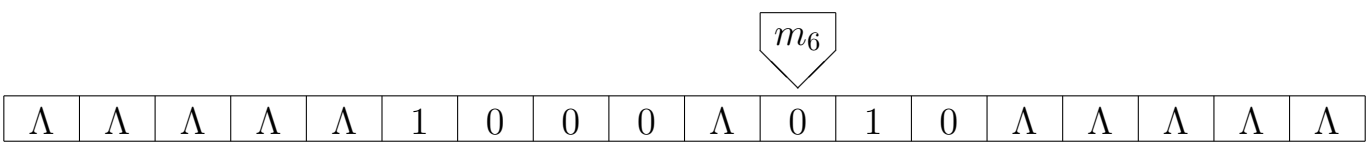
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



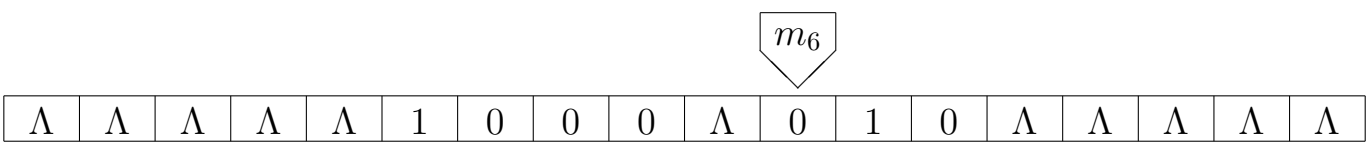
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



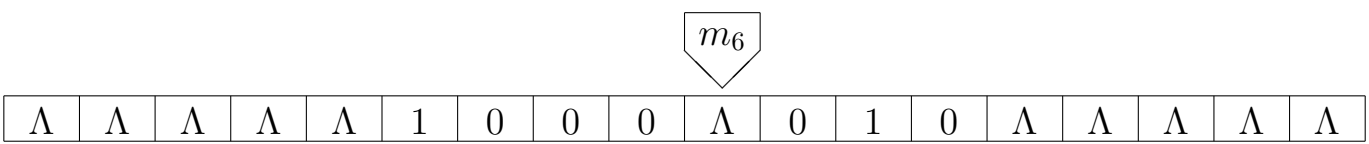
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



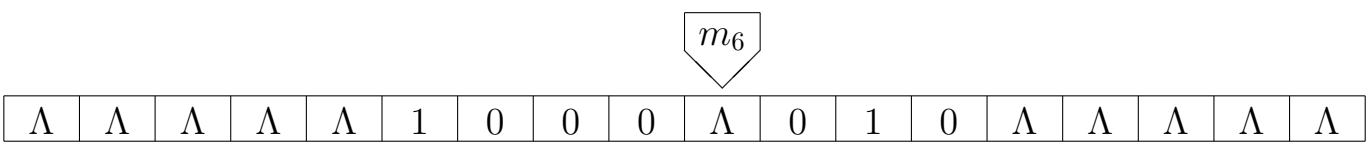
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



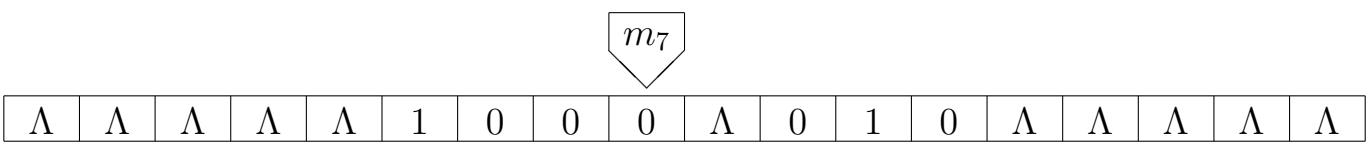
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



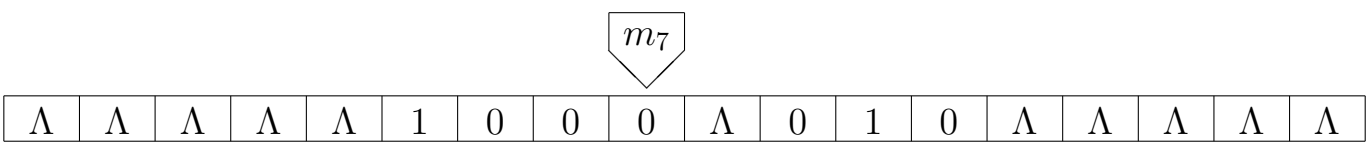
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



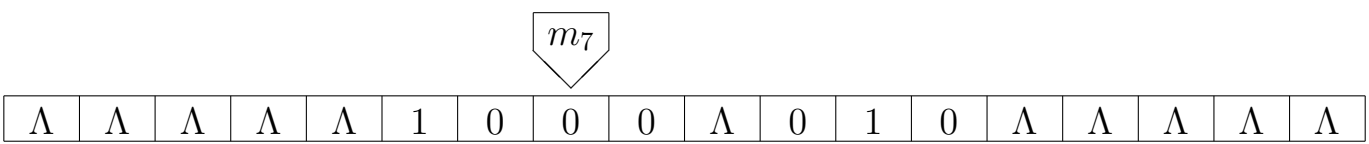
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



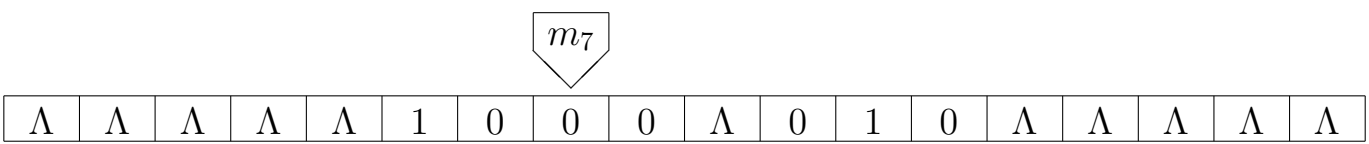
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



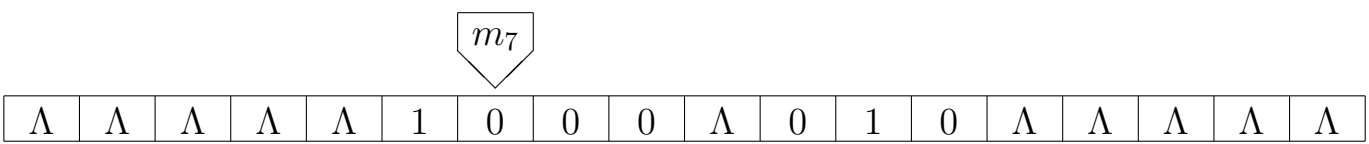
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



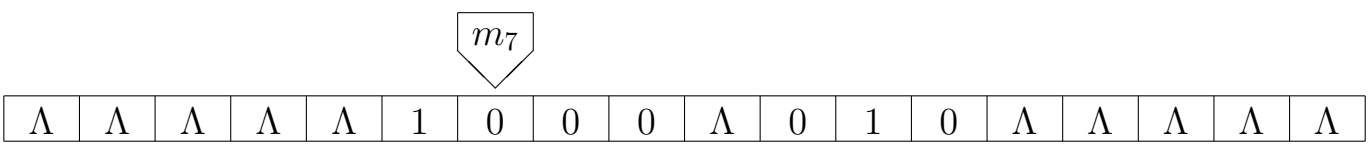
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



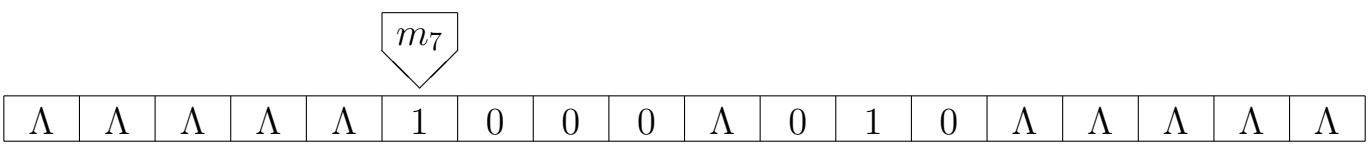
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



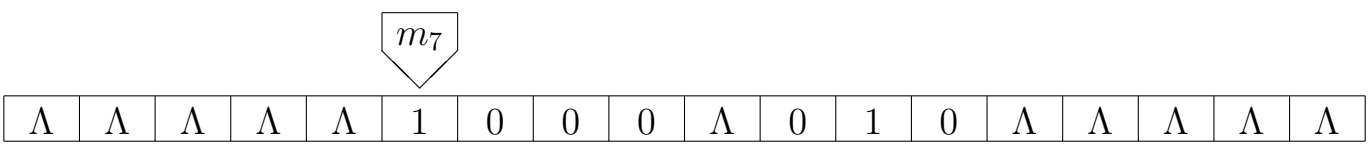
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



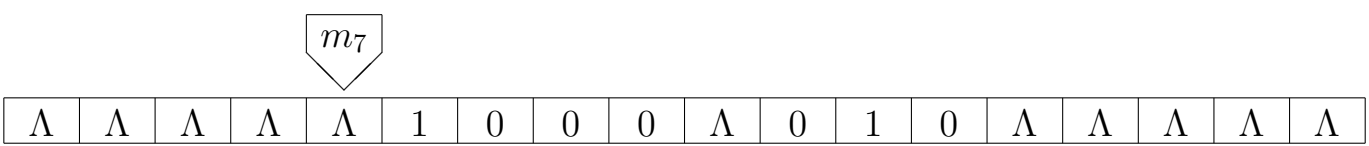
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



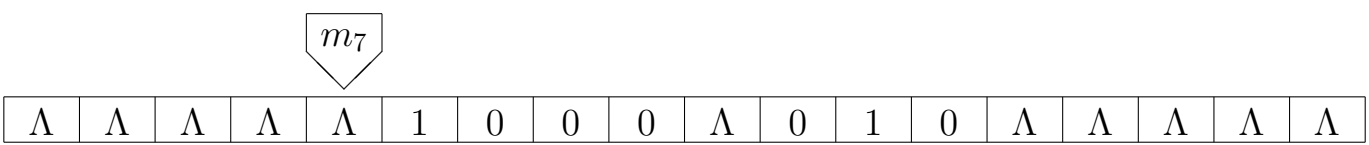
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



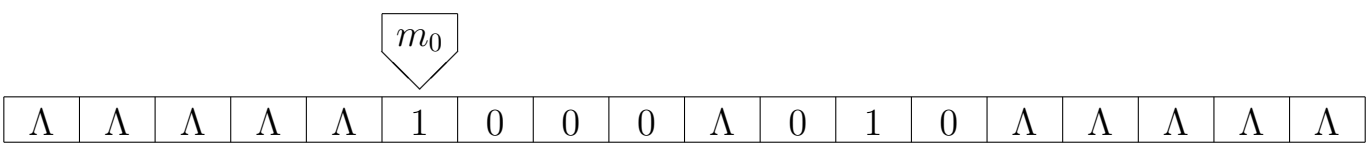
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



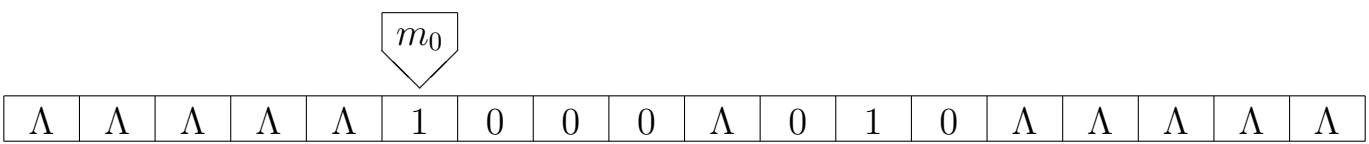
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



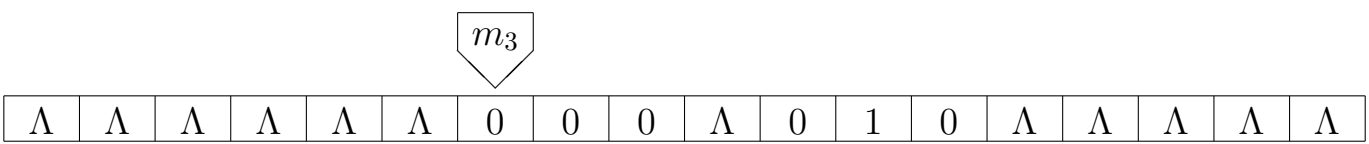
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



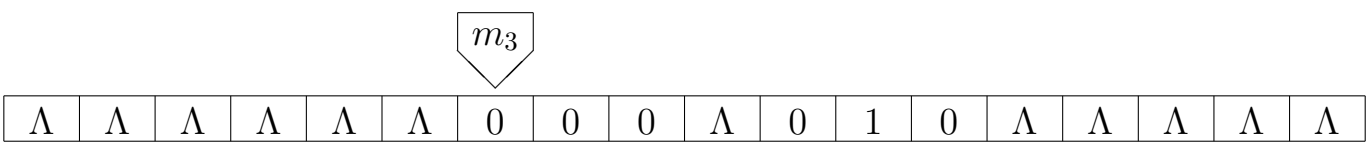
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



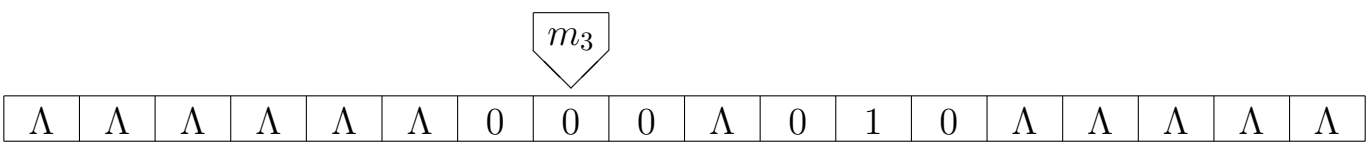
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



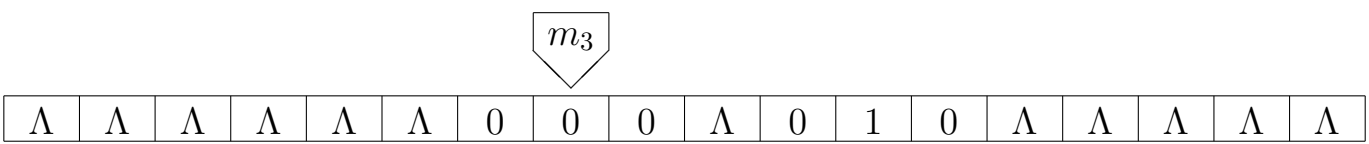
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



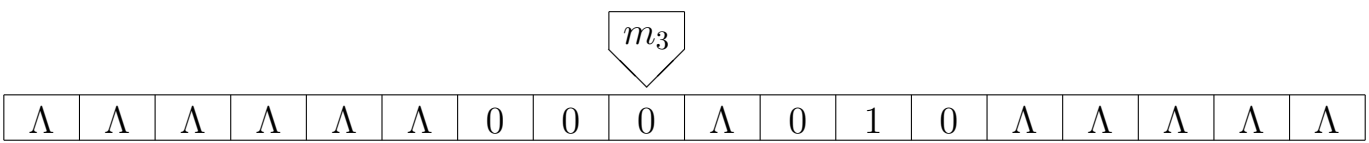
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



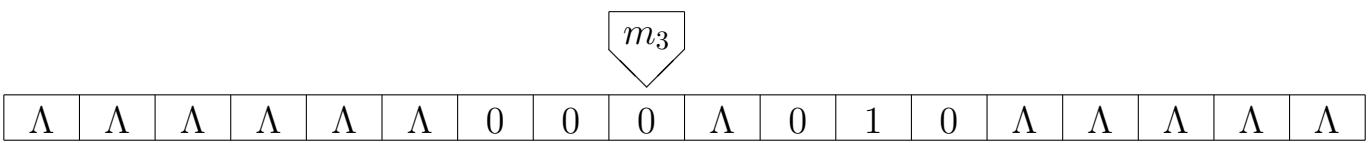
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



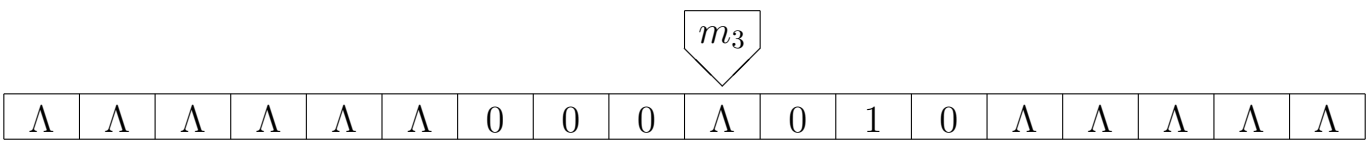
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



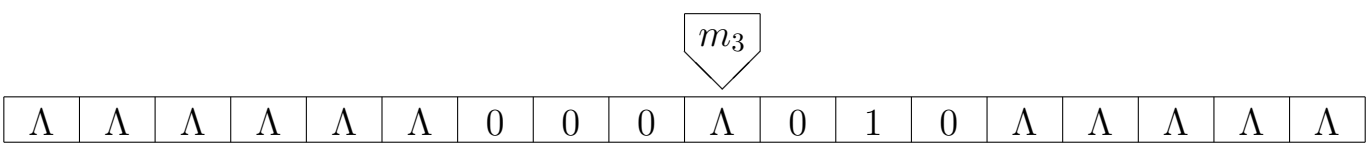
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



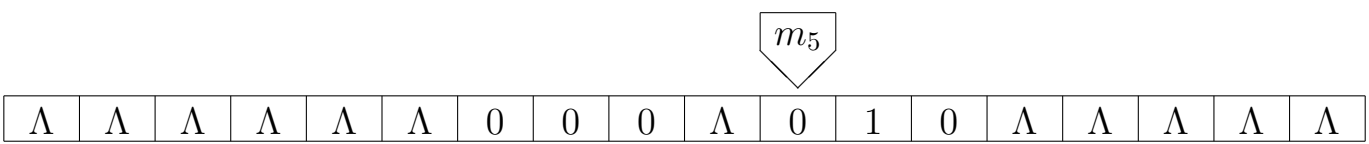
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



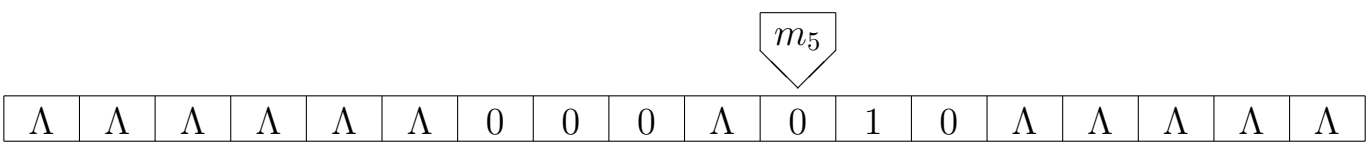
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



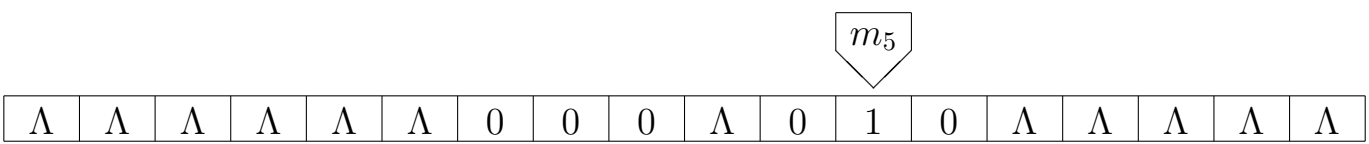
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$		$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



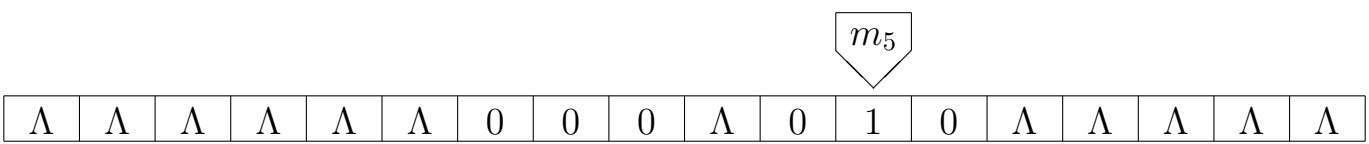
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



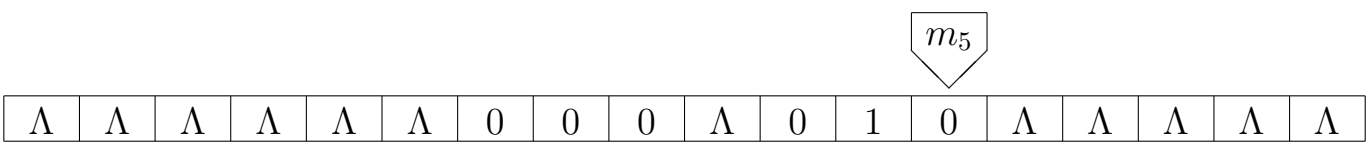
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



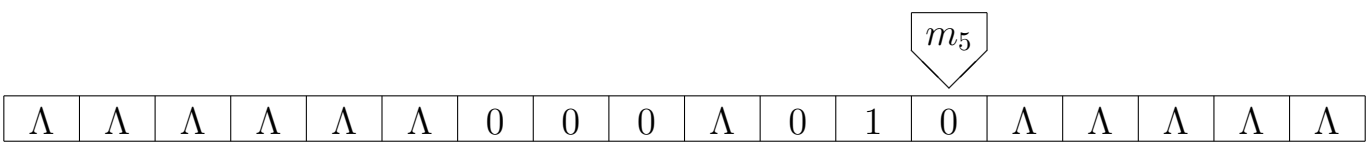
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



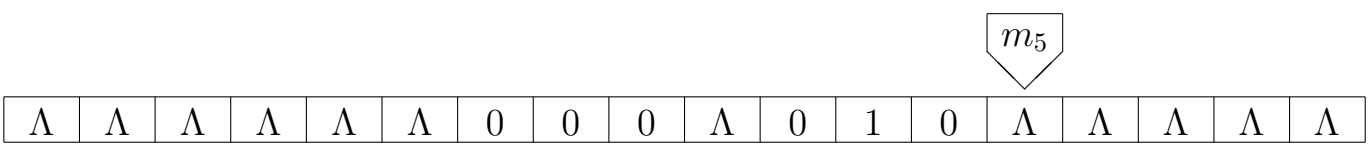
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



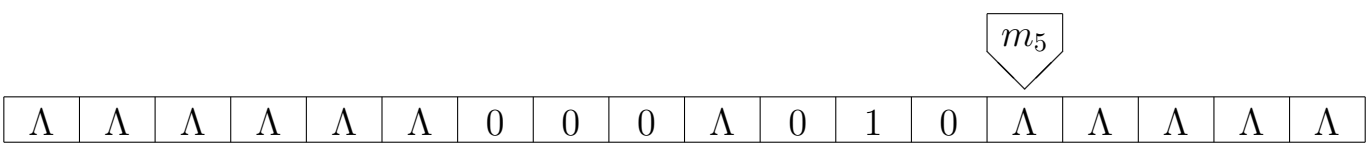
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



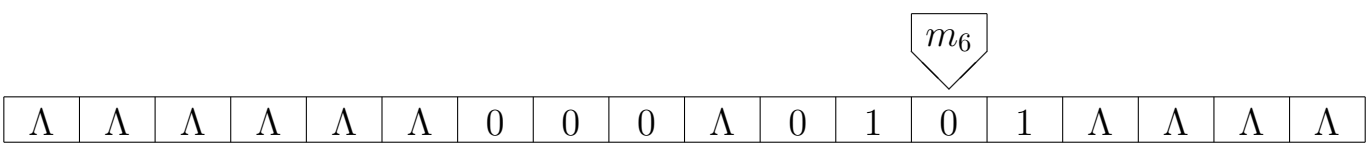
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



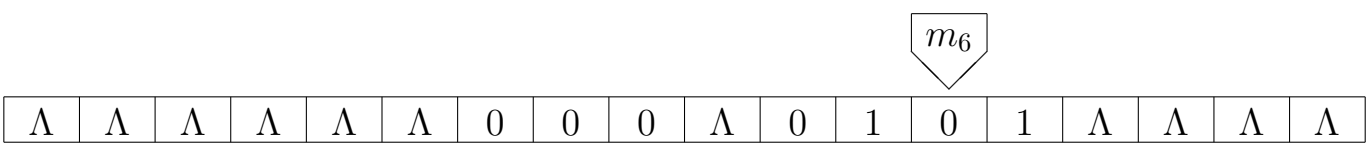
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



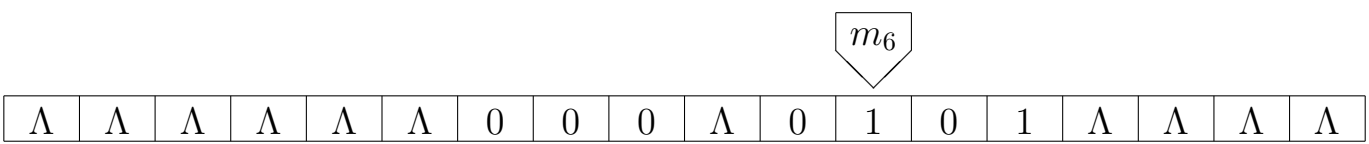
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



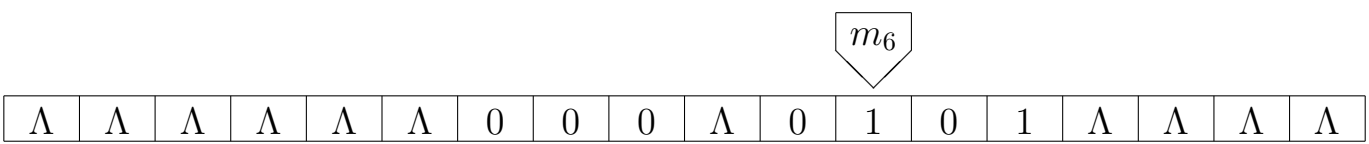
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



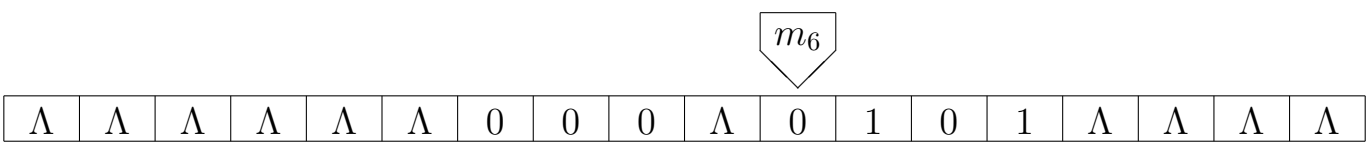
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



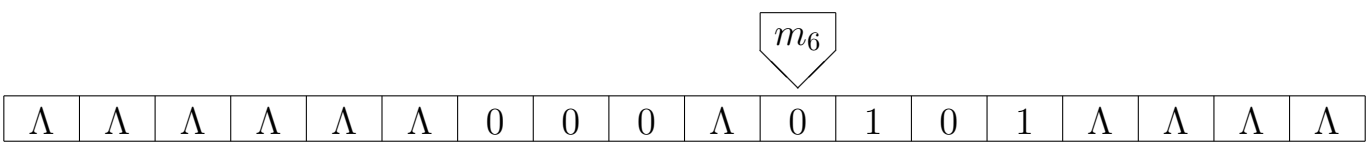
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



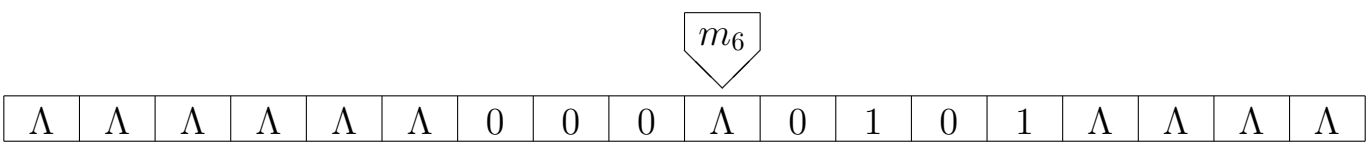
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



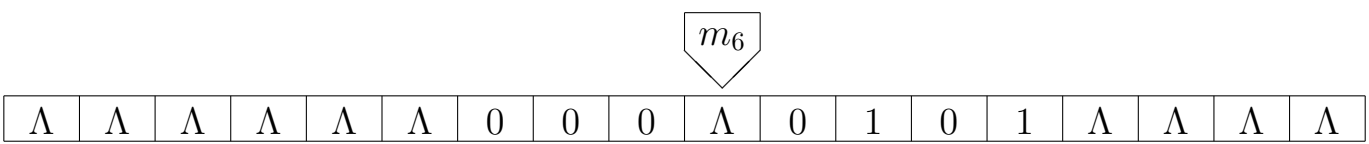
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



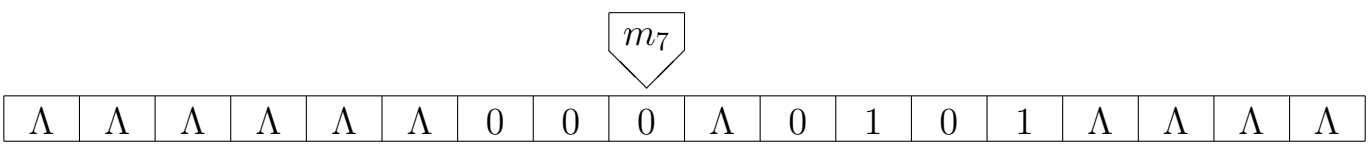
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



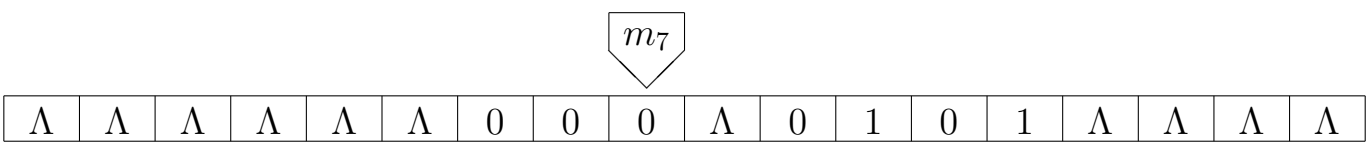
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



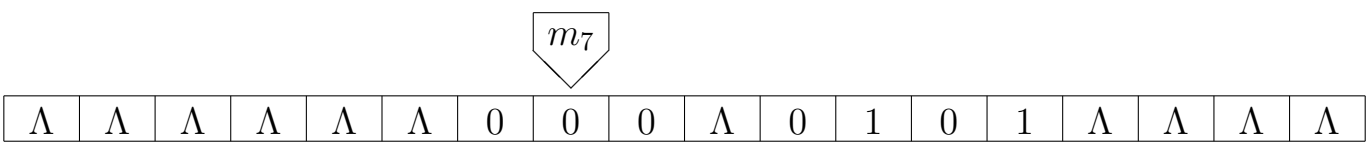
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



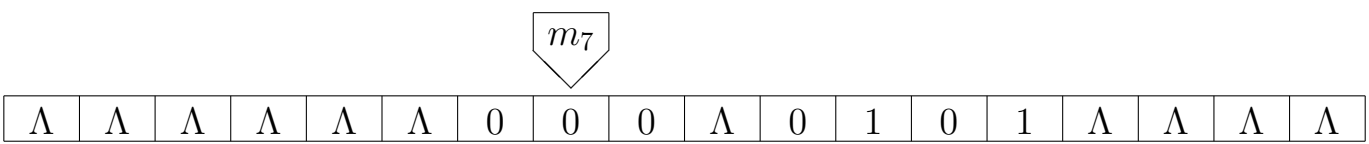
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



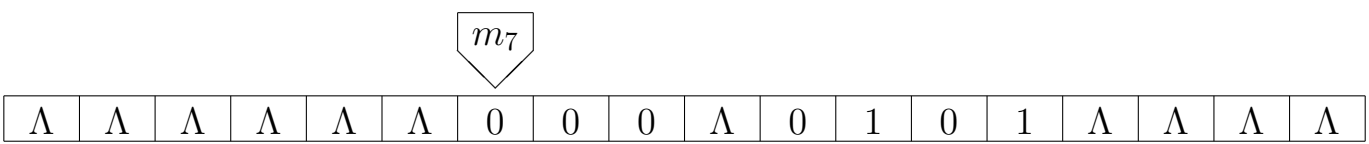
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



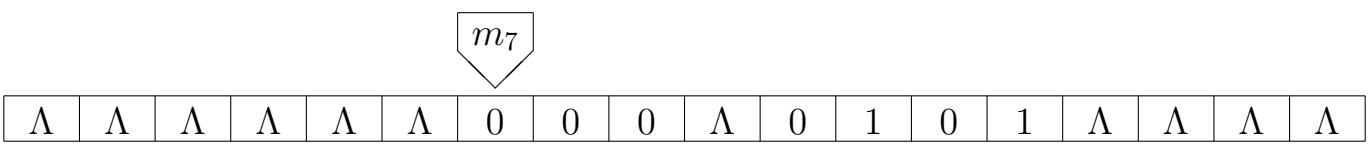
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



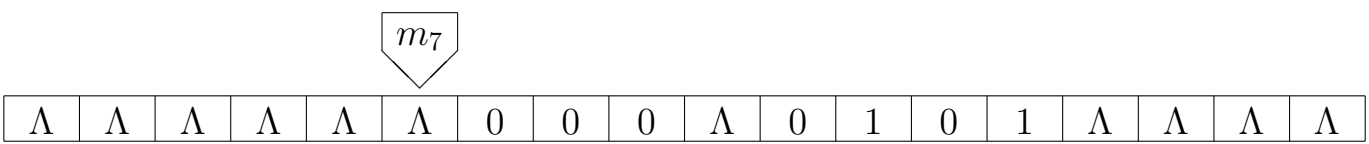
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



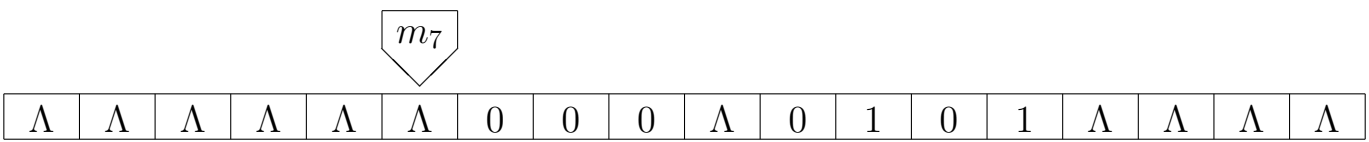
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



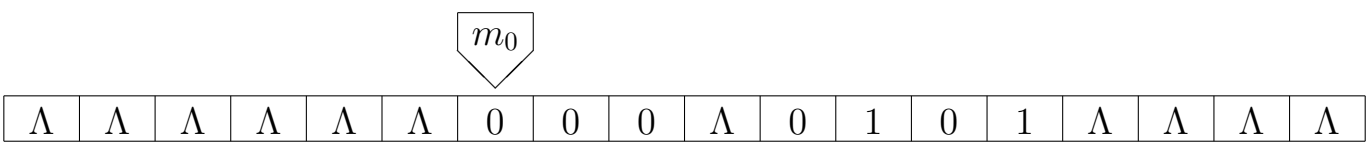
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



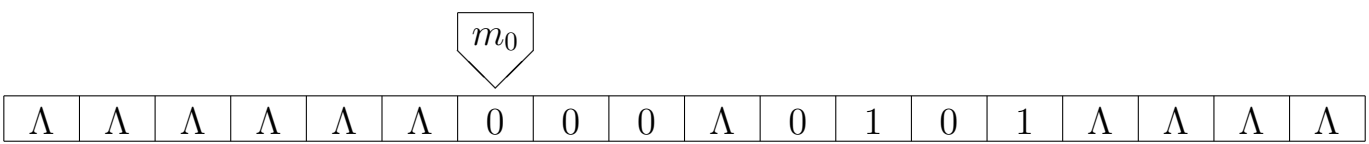
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



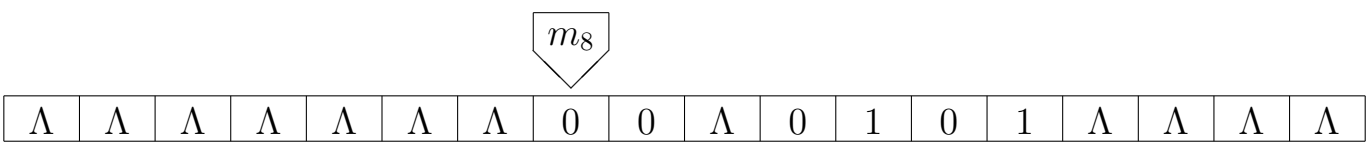
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



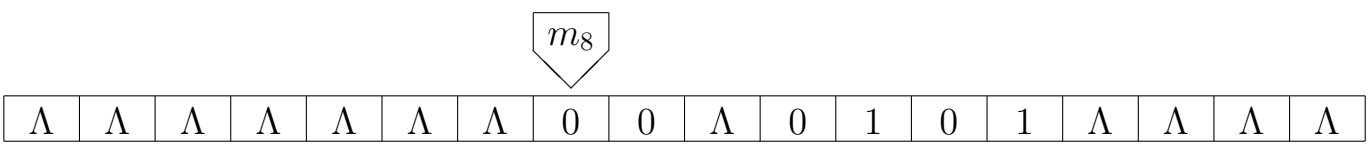
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



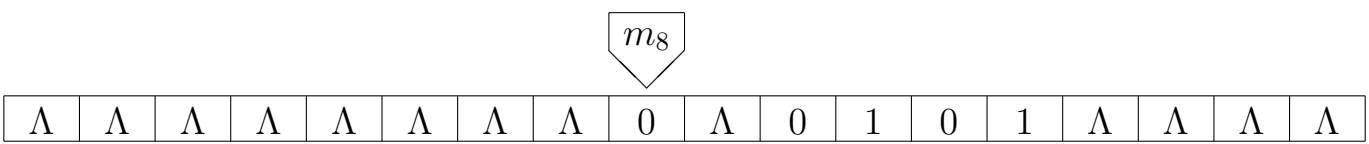
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



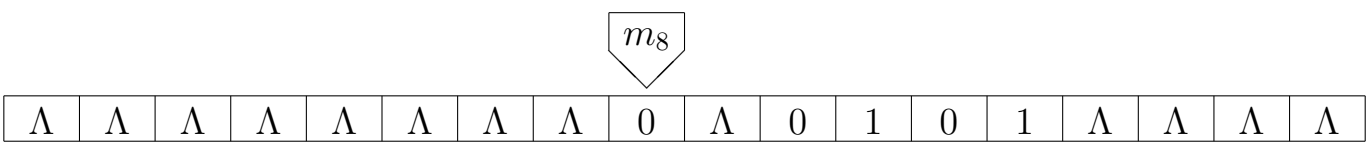
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



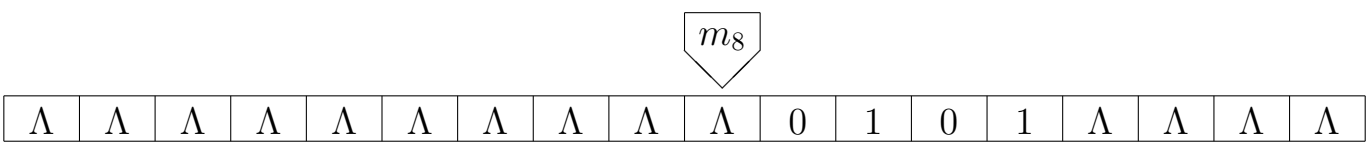
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$			
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



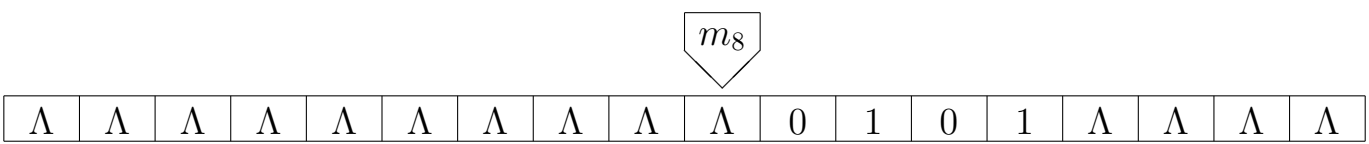
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



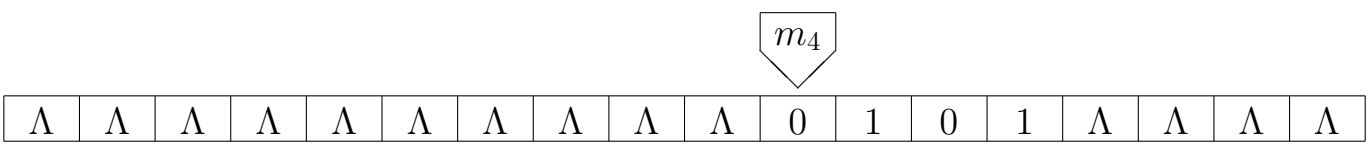
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



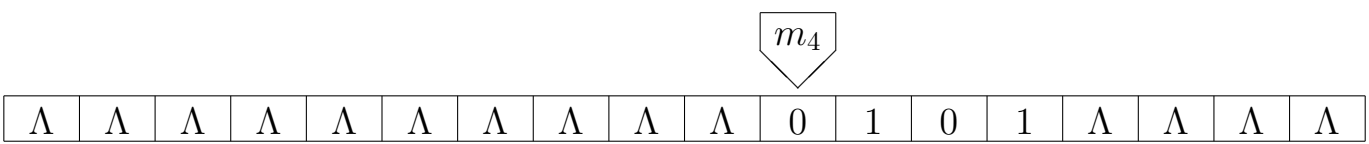
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



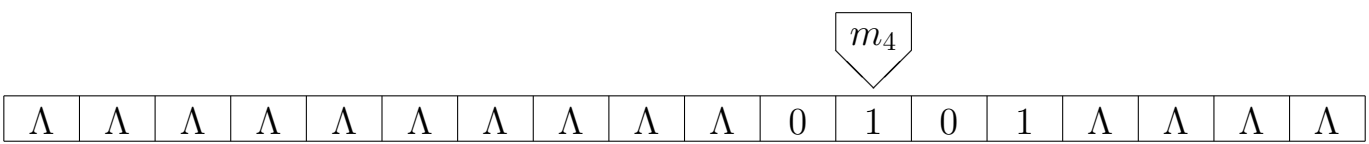
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



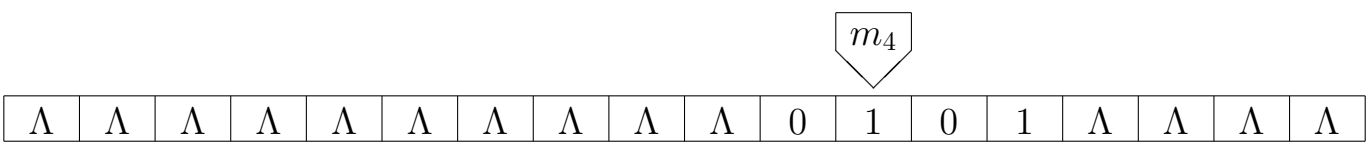
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



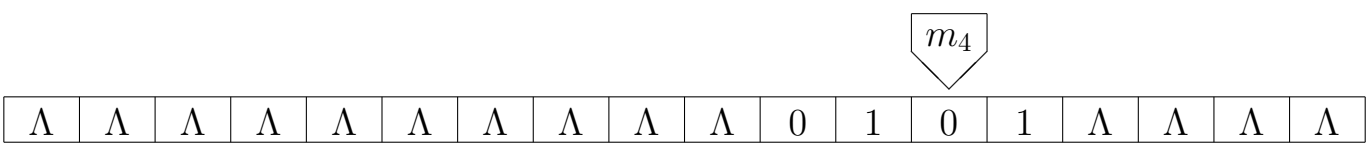
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



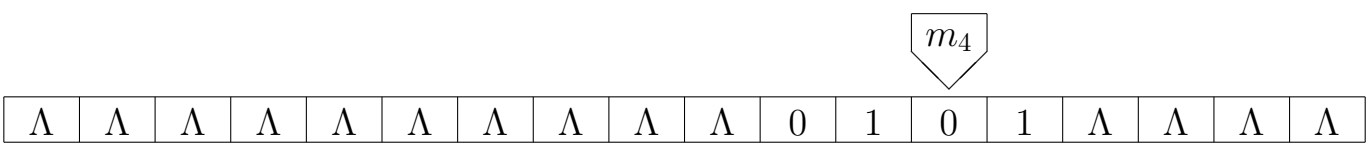
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



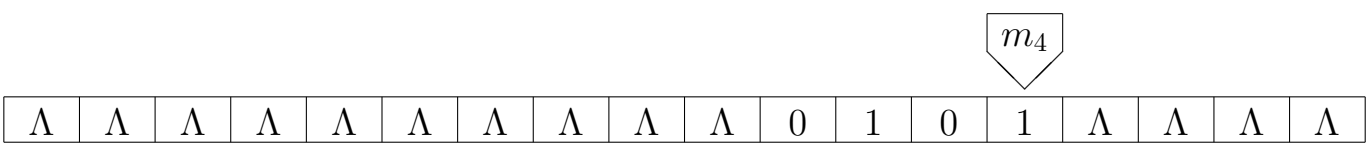
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



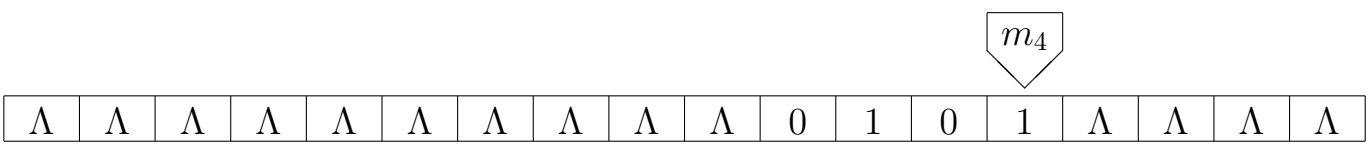
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



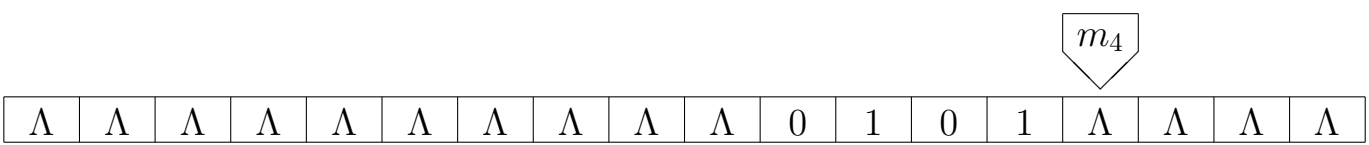
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



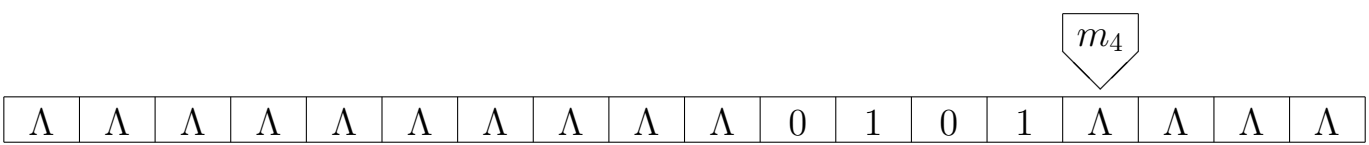
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



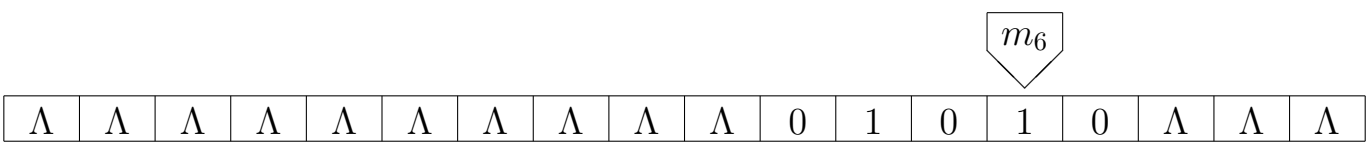
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



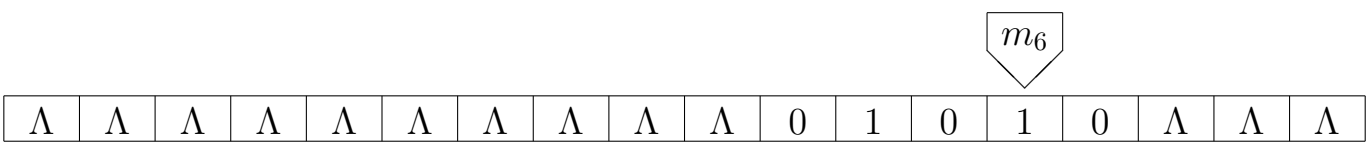
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



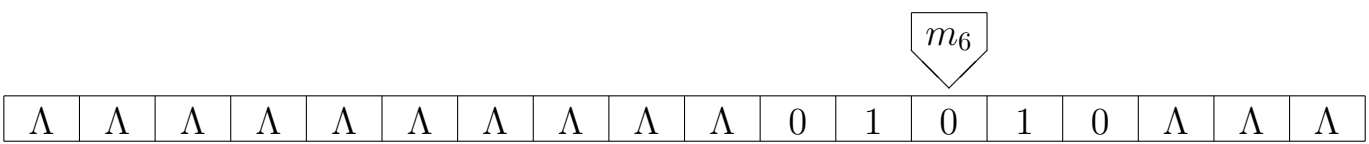
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



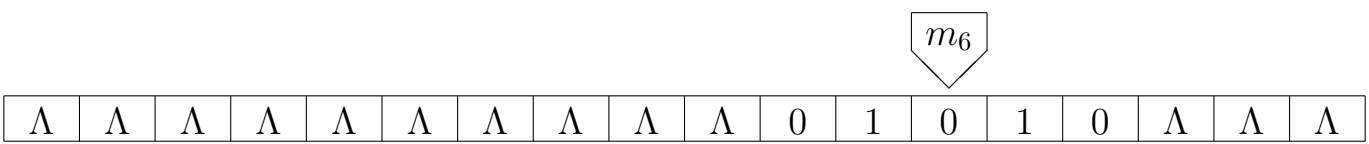
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



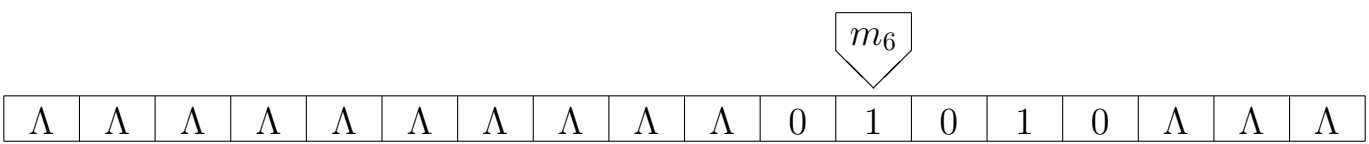
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



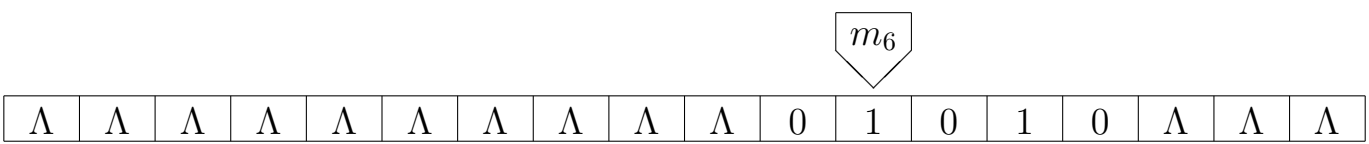
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



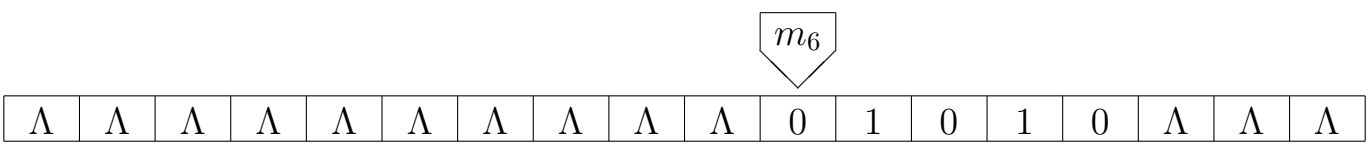
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



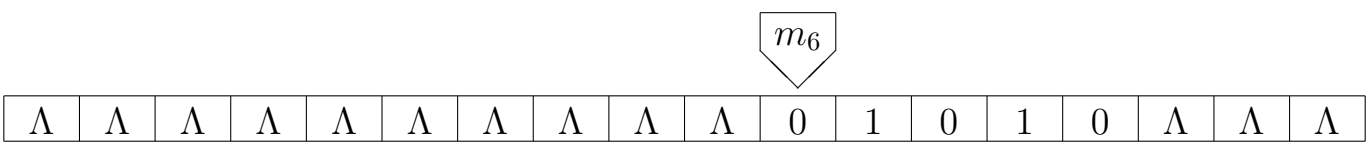
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



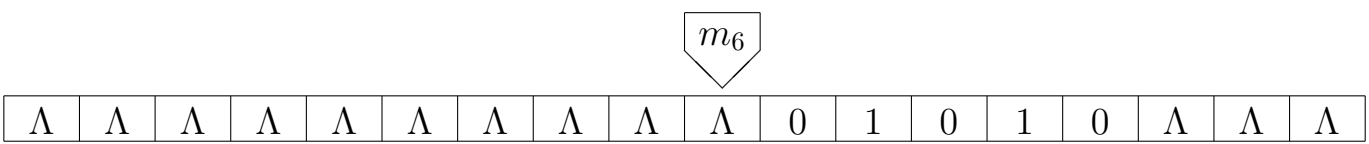
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



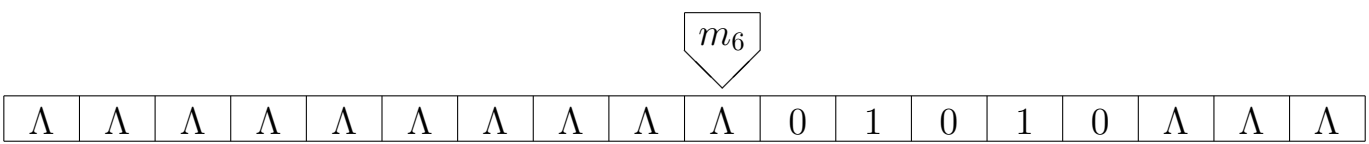
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



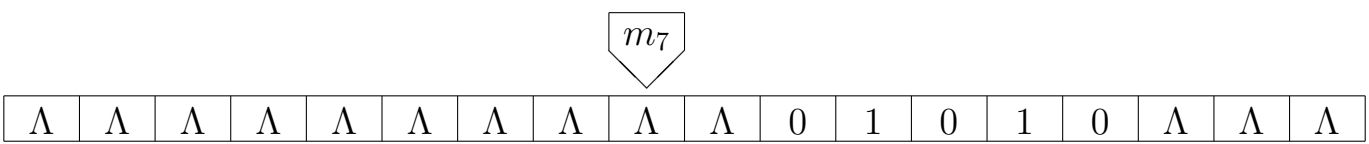
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



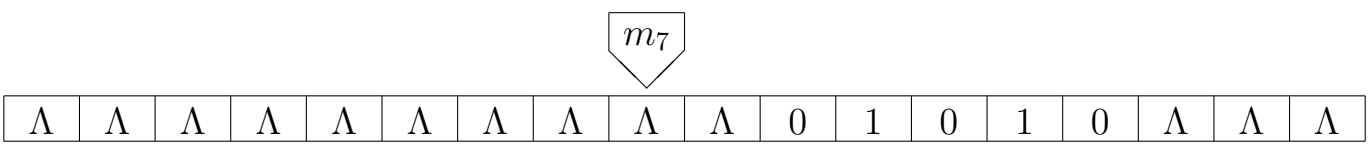
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



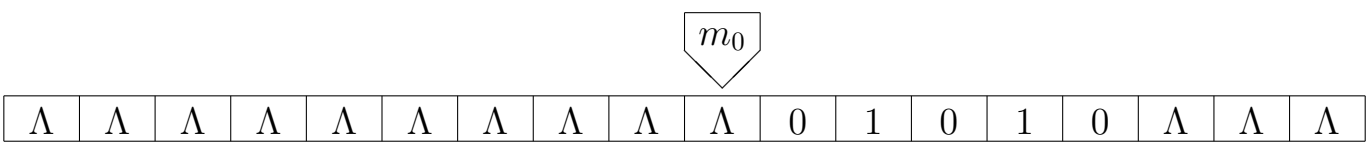
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ		$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



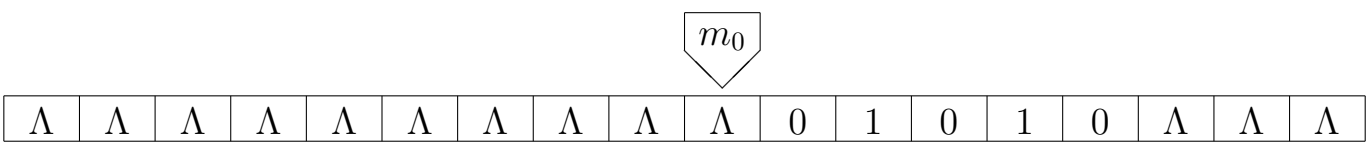
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



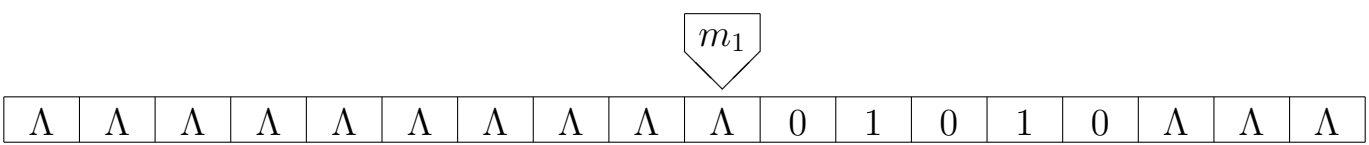
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



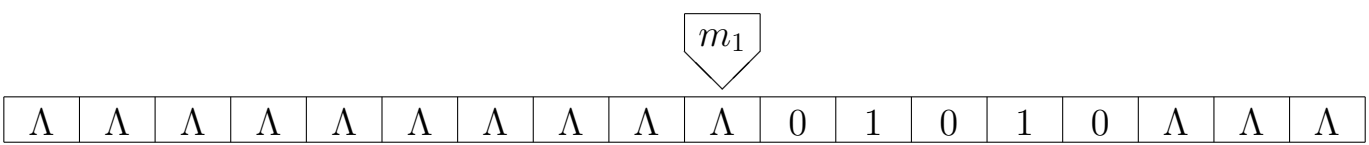
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		

Λ	0	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

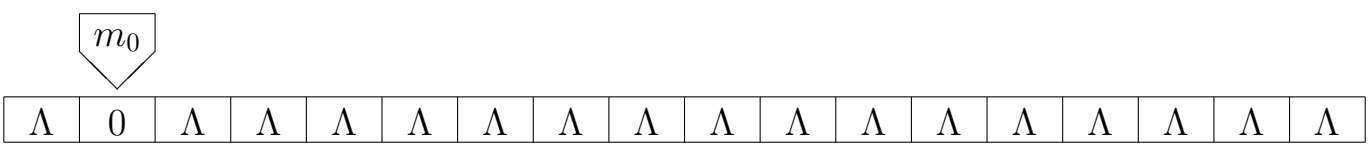
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



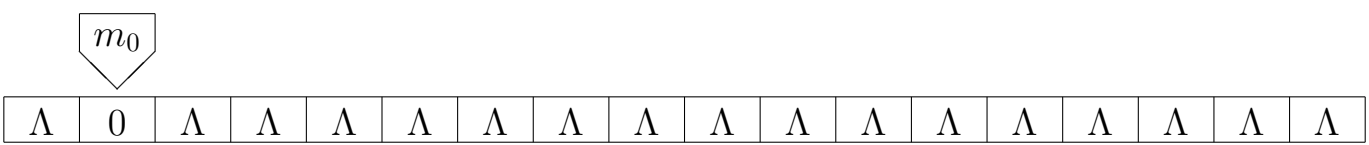
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



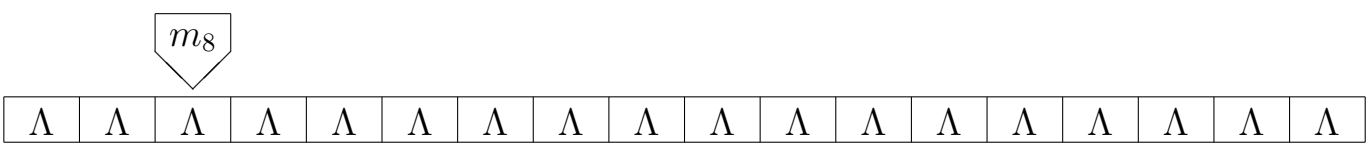
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



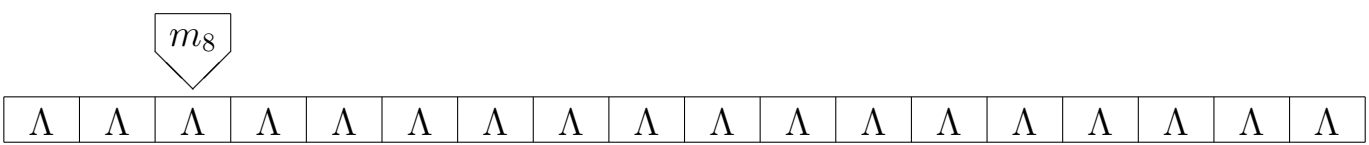
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



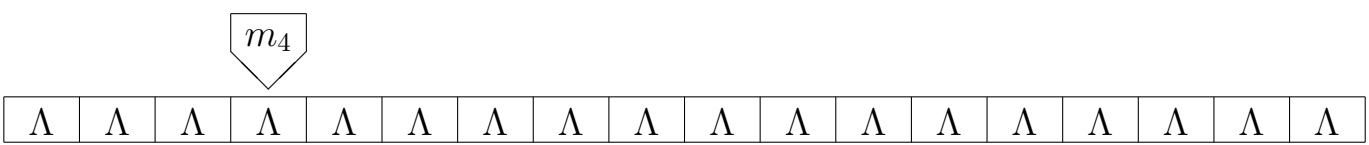
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



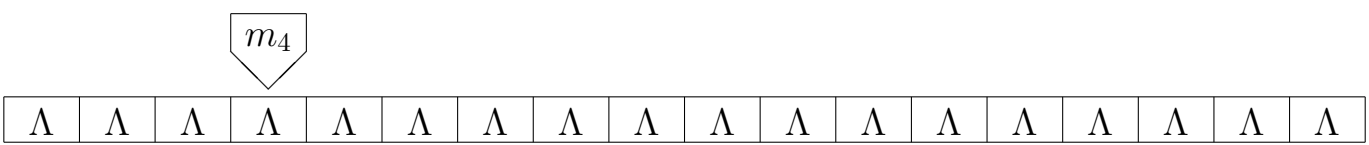
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



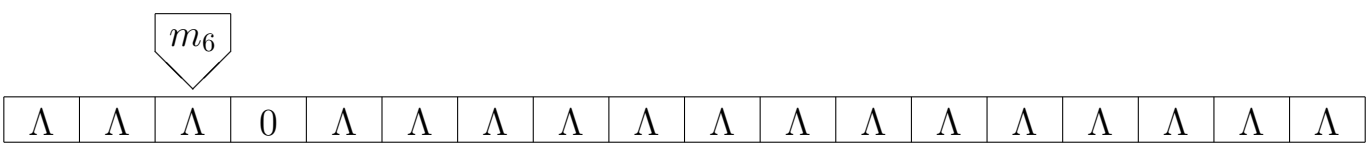
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



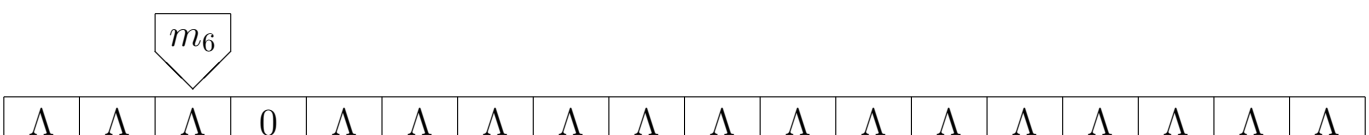
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



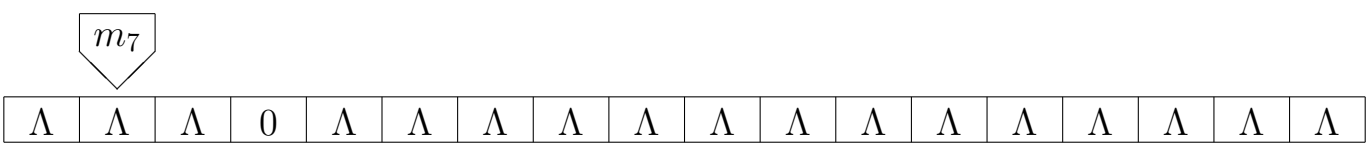
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



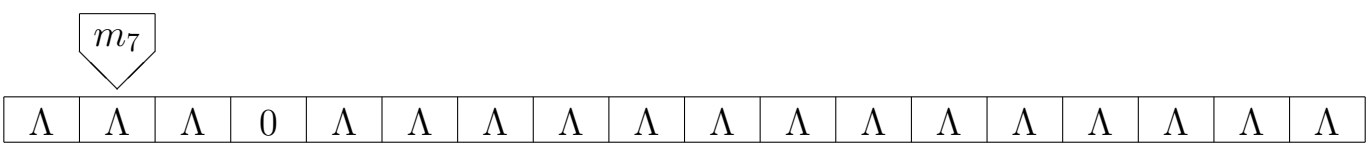
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



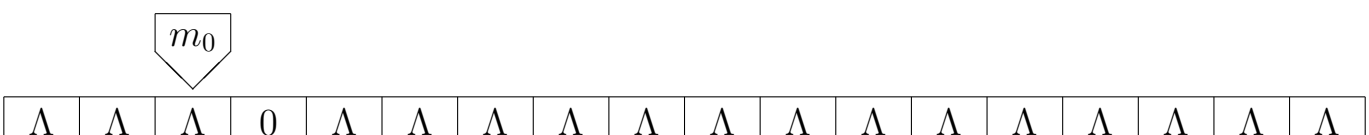
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



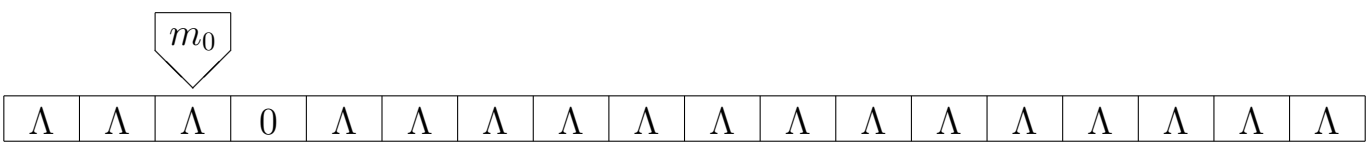
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



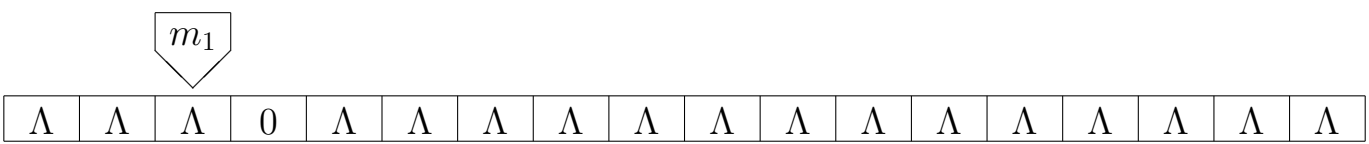
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



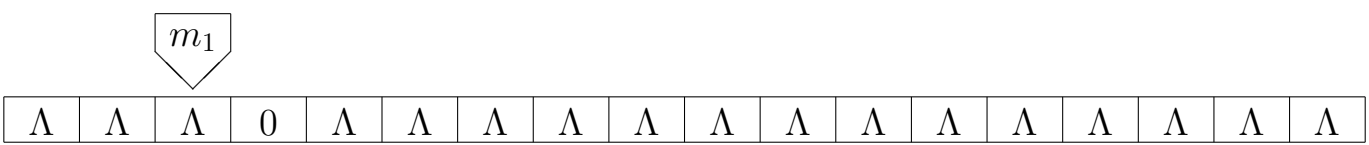
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		

Λ	1	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ	Λ

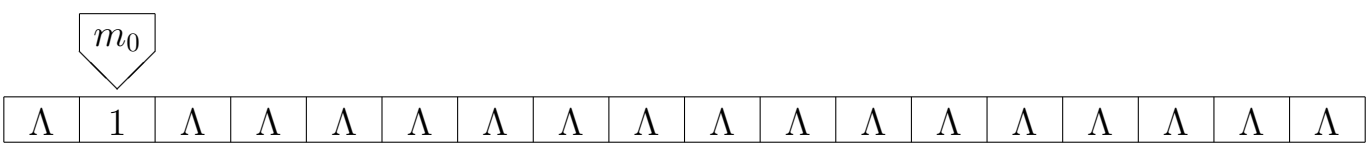
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



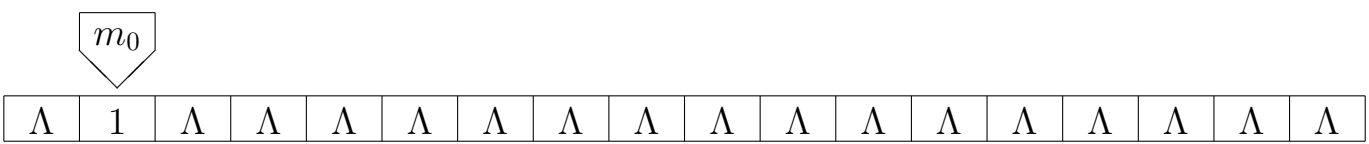
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



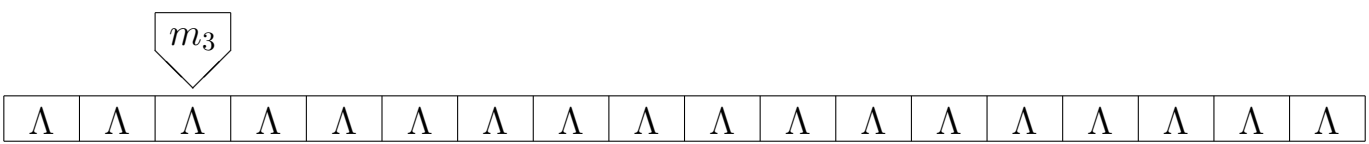
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R\,(\Lambda, m_4)$	$R\,(\Lambda, m_5)$	$L\,(0, m_6)$	$L\,(1, m_6)$	$L\,(\Lambda, m_7)$	$R\,(\Lambda, m_0)$	$R\,(\Lambda, m_4)$		
0	$R\,(\Lambda, m_8)$	$R\,(0, m_2)$	$R\,(0, m_3)$	$R\,(0, m_4)$	$R\,(0, m_5)$	$L\,(0, m_6)$	$L\,(0, m_7)$	$R\,(\Lambda, m_8)$		
1	$R\,(\Lambda, m_3)$	$R\,(1, m_2)$	$R\,(1, m_3)$	$R\,(1, m_4)$	$R\,(1, m_5)$	$L\,(1, m_6)$	$L\,(1, m_7)$	$R\,(1, m_2)$		



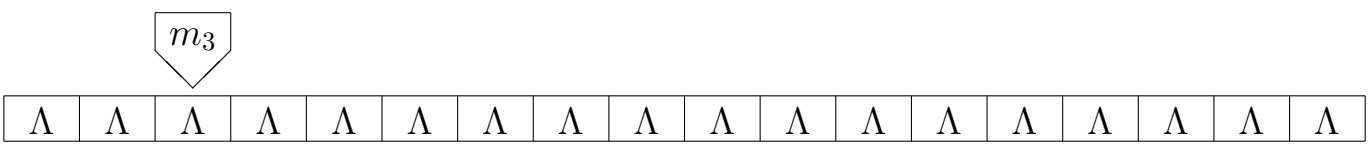
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



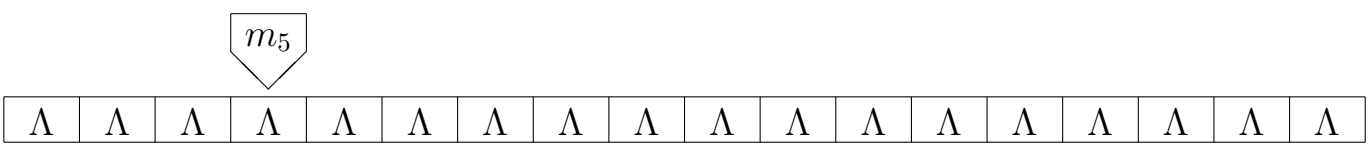
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



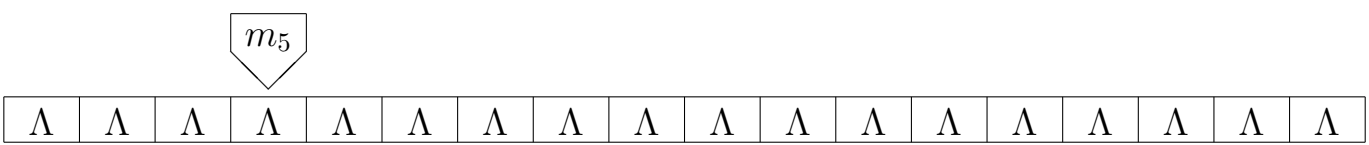
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0, 1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R\,(\Lambda, m_4)$	$R\,(\Lambda, m_5)$	$L\,(0, m_6)$	$L\,(1, m_6)$	$L\,(\Lambda, m_7)$	$R\,(\Lambda, m_0)$	$R\,(\Lambda, m_4)$		
0	$R\,(\Lambda, m_8)$	$R\,(0, m_2)$	$R\,(0, m_3)$	$R\,(0, m_4)$	$R\,(0, m_5)$	$L\,(0, m_6)$	$L\,(0, m_7)$	$R\,(\Lambda, m_8)$		
1	$R\,(\Lambda, m_3)$	$R\,(1, m_2)$	$R\,(1, m_3)$	$R\,(1, m_4)$	$R\,(1, m_5)$	$L\,(1, m_6)$	$L\,(1, m_7)$	$R\,(1, m_2)$		



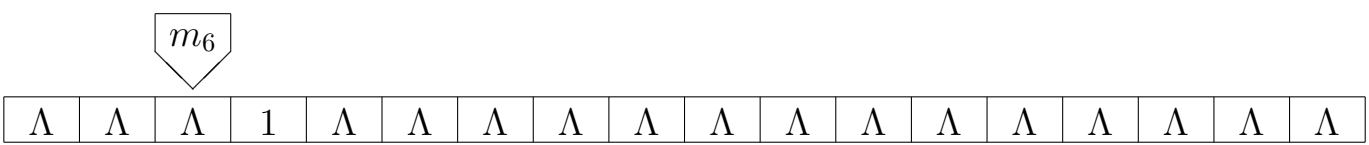
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R\,(\Lambda, m_4)$	$R\,(\Lambda, m_5)$	$L\,(0, m_6)$	$L\,(1, m_6)$	$L\,(\Lambda, m_7)$	$R\,(\Lambda, m_0)$	$R\,(\Lambda, m_4)$		
0	$R\,(\Lambda, m_8)$	$R\,(0, m_2)$	$R\,(0, m_3)$	$R\,(0, m_4)$	$R\,(0, m_5)$	$L\,(0, m_6)$	$L\,(0, m_7)$	$R\,(\Lambda, m_8)$		
1	$R\,(\Lambda, m_3)$	$R\,(1, m_2)$	$R\,(1, m_3)$	$R\,(1, m_4)$	$R\,(1, m_5)$	$L\,(1, m_6)$	$L\,(1, m_7)$	$R\,(1, m_2)$		



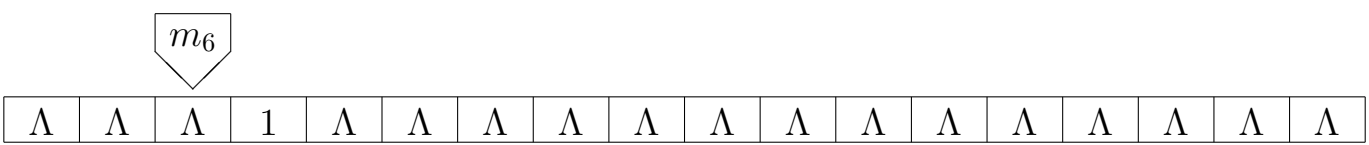
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



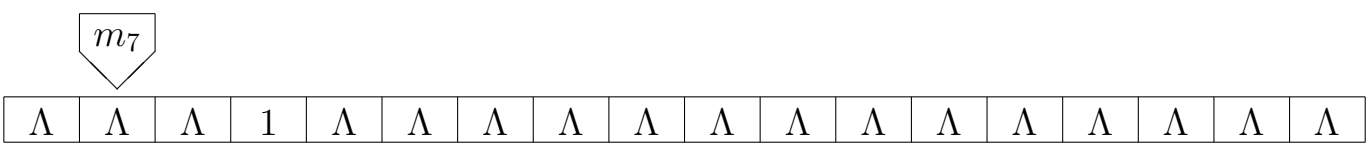
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



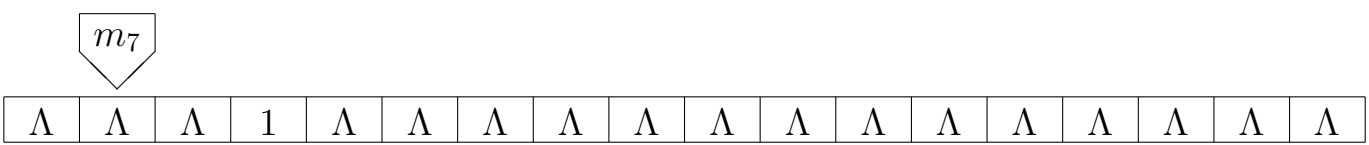
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



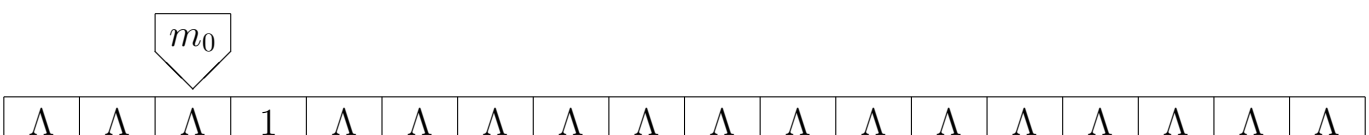
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



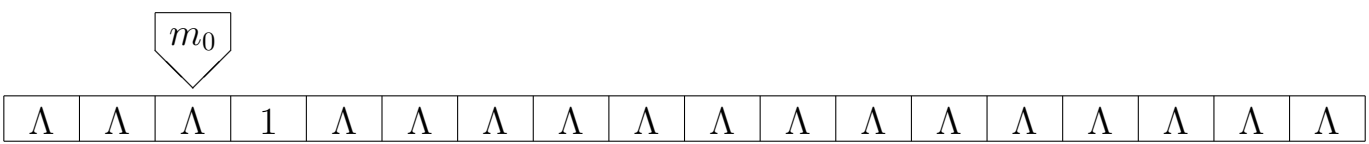
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



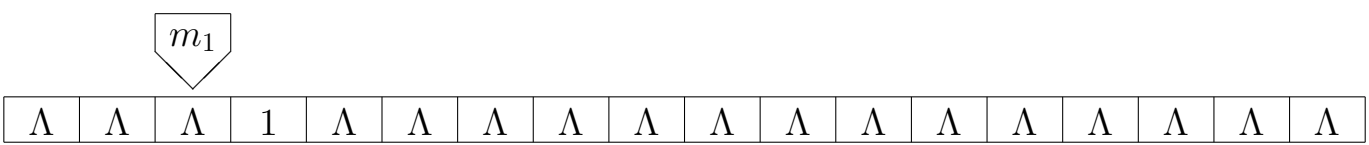
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



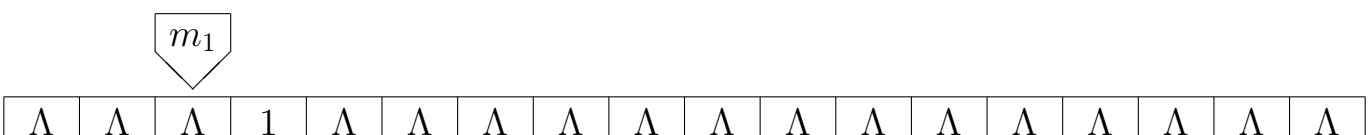
Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		



Задача 5. Написать **a)** нормальный алгорифм Маркова и **b)** машину Тьюринга, во входном слове в алфавите $\{0,1\}$ заменяющую все кратные вхождения нулей на одиночные нули (например, заменяющую слово $0\,1\,\underbrace{0\,0}_{0}\,1\,\underbrace{0\,0\,0}_{0}$ в слово 01010).

Ответ. Реализуем 2 идеи:

а) «подтягивание» правой части к «дырке» с одновременным истреблением кратности нулей;

б) переписать входное слово в другом месте уже в приличном виде.

	m_0	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
Λ	(Λ, m_1)	$R(\Lambda, m_4)$	$R(\Lambda, m_5)$	$L(0, m_6)$	$L(1, m_6)$	$L(\Lambda, m_7)$	$R(\Lambda, m_0)$	$R(\Lambda, m_4)$		
0	$R(\Lambda, m_8)$	$R(0, m_2)$	$R(0, m_3)$	$R(0, m_4)$	$R(0, m_5)$	$L(0, m_6)$	$L(0, m_7)$	$R(\Lambda, m_8)$		
1	$R(\Lambda, m_3)$	$R(1, m_2)$	$R(1, m_3)$	$R(1, m_4)$	$R(1, m_5)$	$L(1, m_6)$	$L(1, m_7)$	$R(1, m_2)$		

Задача решена.

Спасибо за внимание!

e-mail: melnikov@k66.ru, melnikov@r66.ru
сайты: http://melnikov.k66.ru, http://melnikov.web.ur.ru

Вернуться к списку презентаций?