

Государственное образовательное учреждение ВПО
Уральский государственный экономический университет



Ю. Б. Мельников



Стратегия подготовки электронного учебника с помощью издательской системы L^AT_EX: инструкции

e-mail: melnikov@k66.ru,
melnikov@r66.ru

сайты:
<http://melnikov.k66.ru>,
<http://melnikov.web.ur.ru>

Екатеринбург
2012

Введение	15
I. Литература по L^AT_EX	17
II. О системе L^AT_EX	21
II.1. Отличия L ^A T _E X 2 _ε от Word и его аналогов	25
II.2. Установка L ^A T _E X	29
II.2.1. Установка MikTeX2.9	30
II.2.2. Установка TeXCreator	40
II.2.3. Настройка и работа в TeXCreator	42
II.2.4. Система PreT _E X	51
II.2.5. Система TeXSample	65
II.2.6. Система TeXCreator	72
II.3. Создание и компиляция файла *.tex	84

II.3.1. Компиляция файла *.tex	85
II.3.2. Реакция на ошибки компиляции	107
II.3.3. Файлы, создаваемые в процессе компиляции . . .	115
II.4. Структура файла *.tex	120
II.4.1. Преамбула	123
II.4.2. Основной текст	125

III. Набор текста, символов и скобок 128

III.1. Абзац и новая строка	130
III.2. Символы и многострадальная буква ё	132
III.3. Скобки	135
III.4. Начертание и размер шрифта	139
III.5. Пробелы	143

IV. Набор математических формул 145

IV.1. Верхние и нижние индексы	162
--	-----

IV.2. Греческие буквы в формулах	163
IV.3. Готические буквы в формулах	164
IV.4. Некоторые элементарные функции	166
IV.5. Бинарные операции	167
IV.6. Отношения	168
IV.7. Операции	169
IV.8. Операции с пределами	170
IV.9. Одно над другим. Деление. Растягивающиеся скобки .	176
IV.10. Стрелки	194
IV.11. Разные символы	195
IV.12. Некоторые сокращения	196

V. Прямоугольные фрагменты (боксы) 197

V.1. Объединение элементов строки в бокс: <code>\mbox</code>	198
V.2. Выделение элементов строки в рамку: <code>\fbox</code>	206
V.3. Бокс фиксированного размера: <code>\makebox</code>	209

V.4. Бокс из абзацев: <code>\parbox</code>	219
V.5. Смещение бокса вверх или вниз <code>\raisebox</code>	225
V.6. Горизонтальное смещение элементов внутри бокса: <code>\hbox</code>	231
V.7. Вращение бокса	240
VI. Таблицы, системы уравнений и т.п.	246
VI.1. Таблицы в текстовой моде	247
VI.2. Абзац в ячейке таблицы	253
VI.3. Неполные строки в таблице	257
VI.4. Объединение ячеек	265
VI.5. Неполная горизонтальная граница	273
VI.6. Таблицы в математической моде	277
VI.7. Матрицы и системы отношений	278
VII. Команды $\text{\LaTeX}$	282
VII.1. Команды с аргументами	285

VII.2. Создание новых команд	286
VII.2.1. Создание новых команд с помощью <code>\def</code>	291
VII.2.2. Создание новых команд с помощью <code>\newcommand</code> и <code>\renewcommand</code>	301
VII.3. Регистры и счётчики	304
VII.3.1. Введение новых регистров и счётчиков	305
VII.3.2. Печать значений счётчика	306
VII.3.3. Изменение значений счётчиков	313
VII.3.4. Сравнение операций со счётчиками и регистрами	316
VII.3.5. Команды сравнения и цикла	317
VII.3.6. Регистры: введение новых регистров	321
VII.3.7. Регистры: печать и изменение значений	322
VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра	323
VII.3.9. Команда печати простого числа из <code>[1; 999]</code> . . .	328

VII.4. Гипермишени и гиперссылки	329
VIII. Секционирование текста для слайдов	340
IX. Окружения и форматирование текста	345
IX.1. Окружения	346
IX.2. Форматирование текста	347
IX.3. Расстояние между абзацами	351
X. Работа с окружением типа «теорема»	356
X.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»	357
X.2. Задачи для самостоятельного решения и иллюстративные примеры	375

X.3. Требования к оформлению задач для самостоятельного решения	378
X.4. Оформление задач для самостоятельного решения . . .	382
X.5. Окружение «Пример»	391
X.6. Последовательное применение нескольких однотипных окружений типа «теорема»	394

XI. Создание и импорт рисунков 397

XI.1. Импорт рисунков	398
XI.2. Особенности создания рисунков	405
XI.3. Команды для рисования	407
XI.3.1. Псевдорисунки: окружение <code>picture</code>	408
XI.3.2. Помещение объекта в точку с данными координатами: команда <code>\put</code>	410
XI.3.3. Вектор: команда <code>\vector</code>	411
XI.3.4. Отрезок: команда <code>\line</code>	420

XI.3.5. Пунктирный отрезок: команда <code>\dashline</code>	423
XI.3.6. Окружности и круги: команда <code>\circle</code>	425
XI.3.7. Окружности: команда <code>\bigcircle</code>	427
XI.3.8. Кривая Безье: команда <code>\qbezier</code>	428
XI.3.9. Кривые: команда <code>\curve</code> из пакета <code>curves</code>	432
XI.3.10. Овал: команда <code>\oval</code>	438
XI.3.11. Кратное размещение объекта: <code>\multiput</code>	446
XI.3.12. Дуга окружности: команда <code>\arc</code>	451
XI.3.13. Эллипс: команда <code>\ellipsA</code>	453
XI.3.14. Изображение прямоугольной декартовой системы координат: команда <code>\ComSystCoord</code>	478
XI.3.15. Штриховка области	480
XI.4. Импорт рисунков	517
XI.5. Размещение текста слева (справа) от рис.	518
XI.5.1. Пример применения команды <code>\PictText</code>	519

XI.5.2. Пример применения команды <code>\TextPict</code>	520
XI.5.3. Пример применения команды <code>\TextPict</code>	521
XI.6. Создание графиков	522
XI.6.1. Построение гладкой кривой по частям с помощью <code>\qbezier</code>	523
XI.6.2. Создание графика с помощью <code>\curve</code>	527
XI.7. Пример получения некоторых целочисленных параметров для рисования	546

XII. Оформление текста в виде презентаций 547

XII.1. Распределение большого фрагмента текста на слайдах	548
XII.2. Единицы длины	558
XII.3. Команды со значением длины	559
XII.4. Оверлей	566
XII.5. Выделение нескольких неподвижных горизонтальных блоков	567

XII.6. Выделение двух неподвижных горизонтальных блоков	570
XIII. Оформление слайда	577
XIII.1. «Неподвижность» фрагмента при переходе к другому слайду	578
XIII.2. Фоновый рисунок	580
XIV. Разработка тестов в системе ЛАТ_ΕX	587
XIV.1. Тесты для лекционного или практического занятия .	588
XIV.2. Преамбула StartTestBigScr.tex	589
XIV.3. Преамбула StartTestE.tex для пакета exerquiz . . .	618
XIV.4. Пакет exerquiz, окружение shortquiz	619
XIV.5. Пакет exerquiz, окружение quiz	628
XIV.5.1. Тесты с выбором одного варианта	629
XIV.5.2. Тесты с выбором и демонстрацией ответов . . .	642
XIV.5.3. Тесты с текстовым ответом	648

XIV.6. Инструменты преамбулы <code>testsample.tex</code>	658
XIV.6.1. <code>testsample.tex</code> : регистры и счётчики	659
XIV.6.2. <code>testsample.tex</code> : корректировка печати	670
XIV.6.3. <code>testsample.tex</code> : функции на векторах	738
XIV.6.4. <code>testsample.tex</code> : вычисление некоторых целочисленных функций	748
XIV.6.5. <code>testsample.tex</code> : генерирование «псевдослучайных» значений	764
XIV.6.6. <code>testsample.tex</code> : операции с десятичными дробями	772
XIV.6.7. <code>testsample</code> : <u>окружение</u> <code>mathGrp</code>	784
XIV.6.8. <code>testsample.tex</code> : <u>окружение</u> <code>mananswers</code>	789
XIV.6.9. <code>testsample</code> : <u>окружение</u> <code>RespBoxMath</code>	802
XIV.7. Макрокоманды для задания функций в <code>StartDiff.inp</code> .	809
XIV.7.1. <code>StartDiff.inp</code> : <code>\FncA{...}{...}{...}{...}</code> . .	810

XIV.7.2. StartDiff.inp: \FncB{...}{...}{...}{...}	811
XIV.7.3. StartDiff.inp: \FncC{...}{...}{...}{...}	812
XIV.7.4. StartDiff.inp: \FncD{...}{...}{...}{...}	813
XIV.8. Система разработки средств контроля	814
XIV.8.1. Преимущества pdf	815
XIV.8.2. Структурная схема исходного кода для создания электронного теста	816
XIV.9. Структура файла IDZfinish.tex	836
XIV.9.1. IDZfinish: начало файла	837
XIV.9.2. IDZfinish.tex: печать теста	851
XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex	862
XIV.9.4. Пример фрагмента из \STest...	875
XIV.10. Одноразовые мероприятия для генерирования тестов	907
XIV.10.1. Создание файла с данными преподавателя	912

XIV.10.2. Подготовка файла *.tex со списком группы для генерации заданий	914
XIV.11. Структура папок с материалами о группе для со- здания генератора ИДЗ для них	916
XIV.12. Содержимое папки с материалами для генерации конкретного индивидуального домашнего задания . . .	919
XIV.13. Подготовка файлов для генерации заданий	920
XIV.13.1. Подготовка файлов для генерации заданий: начальная подготовка для группы	921
XIV.13.2. Подготовка файлов для генерации заданий: для каждого ИДЗ для группы	923
XIV.13.3. Генерирование заданий	925

Введение

Данный учебник желательно просматривать, используя программу Adobe Reader, поскольку другие программы просмотра не всегда корректно воспроизводят макрокоманды перехода.

Текст, **выделенный синим цветом**, обычно представляет собой гиперссылку в другой документ, а текст **выделенный красным цветом**, обычно представляет собой гиперссылку на мишень в этом же документе.

В программе Adobe Reader для возвращения после перехода по гиперссылке применяется комбинация клавиш (одновременное нажатие) Ctrl←. Кратное нажатие Ctrl← позволяет вернуться на несколько шагов назад (но есть ограничение на число шагов для возвращения).

Введение

При просмотре через Интернет следует настроить браузер таким образом, чтобы файл pdf обрабатывался Adobe Reader.

Например, для Google Chrom такая настройка осуществляется следующим способом:

«Параметры» (символ гаечного ключа), «Дополнительные», «Настройки контента», «Подключаемые модули», отключить Chrome PDF Viewer.

I. Литература по $\text{\LaTeX}$

Мы рекомендуем начать изучение $\text{\LaTeX}$ с работы **С.М. Львовского «Набор и верстка в системе $\text{\LaTeX}$ »** (доступно и раннее издание **С.М. Львовского « $\text{\LaTeX}$: подробное описание»**). Интерес представляет работа, переведенная с английского **«Не очень краткое введение в $\text{\LaTeX 2}_{\epsilon}$ »**. Очень много полезной информации, в частности, по Интернет-ресурсам, содержится в прекрасной книге **М. Гуссенса, Ф. Миттельбаха, А. Самарина «Путеводитель по пакету $\text{\LaTeX}$ и его расширению $\text{\LaTeX 2}_{\epsilon}$ »**. Набор математических формул описан в **работе «Набор математических формул в $\text{\LaTeX 2}_{\epsilon}$ »**. Следует ознакомиться с работой **«Включение рисунков в $\text{\LaTeX 2}_{\epsilon}$ »**.

I. Литература по L^AT_EX

Некоторые особенности русификации L^AT_EX описаны в работе «Русский язык в L^AT_EX».

Различные аспекты создания файлов формата pdf средствами L^AT_EX 2_ε приведены в «Гипертекст в PDF документах, созданных средствами L^AT_EX 2_ε», «Создание цветных иллюстрированных документов на PostScript и в Pdf с помощью L^AT_EX'а», «Цвет в L^AT_EX 2_ε», «Опции пакета hyperref».

Исчерпывающая информация по издательской системе T_EX (развитием которой является L^AT_EX) приведена в книге автора этой системы Дональда Кнута «Все про T_EX».

I. Литература по $\text{\LaTeX}$

Некоторые особенности русификации $\text{\LaTeX}$ описаны в работе «Русский язык в $\text{\LaTeX}$ ».

Различные аспекты создания файлов формата pdf средствами $\text{\LaTeX} 2_{\epsilon}$ приведены в «Гипертекст в PDF документах, созданных средствами $\text{\LaTeX} 2_{\epsilon}$ », «Создание цветных иллюстрированных документов на PostScript и в Pdf с помощью $\text{\LaTeX}$ 'а», «Цвет в $\text{\LaTeX} 2_{\epsilon}$ », «Опции пакета hyperref».

Вы можете воспользоваться нашим «самодельным» справочником, хотя он пока очень беден. Доступен также справочник С.В. Клименко и М.В. Лисиной.

I. Литература по $\text{\LaTeX}$

Использование средств $AMS\text{-}\text{\LaTeX}$ освещено в работе [AMS- \$\text{\LaTeX}\$, версия 1.2: Руководство пользователя](#).

Некоторые особенности применения $\text{\LaTeX}$ для набора химических формул приведены в работе, [видимо, Е.М. Миньковского](#).

II. О системе $\text{\LaTeX}$

Издательская система $\text{\TeX}$ была написана Дональдом Кнутом в 1977 году — в этом году в Калифорнии состоялся первый официальный релиз. По-русски читается как «тех», поскольку Дональд Кнут ссылался на древнегреческое слово «Искусство», т.е. X — это не английская буква «экс», а греческая буква «хи».

II. О системе $\text{\LaTeX}$

Издательская система $\text{\TeX}$ была написана Дональдом Кнутом в 1977 году — в этом году в Калифорнии состоялся первый официальный релиз. По-русски читается как «тех», поскольку Дональд Кнут ссылался на древнегреческое слово «Искусство», т.е. X — это не английская буква «экс», а греческая буква «хи».

Американским математическим обществом в качестве стандарта подготовки статей была принята версия $\text{\TeX}$, названная AMS- $\text{\TeX}$, в которой предусмотрено большое число математических символов и др.

II. О системе $\text{\LaTeX}$

Издательская система $\text{\TeX}$ была написана Дональдом Кнутом в 1977 году — в этом году в Калифорнии состоялся первый официальный релиз. По-русски читается как «тех», поскольку Дональд Кнут ссылался на древнегреческое слово «Искусство», т.е. X — это не английская буква «экс», а греческая буква «хи».

Американским математическим обществом в качестве стандарта подготовки статей была принята версия $\text{\TeX}$, названная AMS-TeX , в которой предусмотрено большое число математических символов и др.

Изначально $\text{\TeX}$ предоставлял немного удобных средств для авторов: сложно было организовать автоматическую нумерацию теорем и т.д. Поэтому профессор Лесли Лампорт разработал вариант $\text{\TeX}$, названный $\text{\LaTeX}$, предоставляющий авторам большое число удобных инструментов для оформления статей, автоматизации ссылок и др.

II. О системе $\text{\LaTeX}$

В проекте $\text{\LaTeX}2_{\epsilon}$ были объединены возможности AMS-TeX и $\text{\LaTeX}$.

Поэтому мы сосредоточим свои усилия на изучении самой популярной версии издательской системы TeX — пакете $\text{\LaTeX}2_{\epsilon}$.

II.1. Отличия $\text{\LaTeX}$ 2 ϵ от Word и его аналогов

От популярного в нашей стране Word'а или Writer из пакета Open Office рассматриваемая издательская система $\text{\LaTeX}$ отличается радикально.

II.1. Отличия $\text{\LaTeX}$ 2 $_{\epsilon}$ от Word и его аналогов

Во-первых, $\text{\LaTeX}$ 2 $_{\epsilon}$ относится к к системе *логического программирования* не к системе *визуального программирования*, как Word или Writer.

II.1. Отличия $\text{\LaTeX 2}_{\epsilon}$ от Word и его аналогов

Во-вторых, $\text{\LaTeX 2}_{\epsilon}$ — это не редактор, а компилятор текста. Исходный текст фактически представляет собой текст программы, он может быть подготовлен в любом текстовом редакторе, не использующем «невидимых» форматирующих символов (например, в Блокноте или Notepad).

Таким образом, автор готовит специальный текст, а система $\text{\LaTeX 2}_{\epsilon}$ обеспечивает его преобразование в текст, предназначенный для читателя.

II.1. Отличия $\text{\LaTeX}$ 2 ϵ от Word и его аналогов

Во-вторых, $\text{\LaTeX}$ 2 ϵ — это не редактор, а компилятор текста. Исходный текст фактически представляет собой текст программы, он может быть подготовлен в любом текстовом редакторе, не использующем «невидимых» форматирующих символов (например, в Блокноте или Notepad).

Исходный текст содержится, в основном, в файле с расширением `tex`, а на выходе, как правило, получается файл в одном из следующих форматов:

`dvi` (Device View Independent — независимый от средства просмотра),

`ps` (на языке PostScript, изначально предназначенном для управления принтерами),

`pdf`.

II.2. Установка $\text{\LaTeX}$

Следует установить новейшую версию $\text{\LaTeX}$, в настоящее время для Windows рекомендуем MikTeX2.9 , для Linux рекомендуем TeXLive .

II.2.1. Установка MikTeX2.9

Для установки MikTeX2.9 следует запустить
`setup-2.9.4321[1].exe`,
находящийся в папке `...\MikTeX2.9`

II.2.1. Установка MikTeX2.9

Для Windows следует после установки MikTeX2.9 скопировать в Program Files\MikTeX2.9\tex содержимое папки tex\TeXSample.

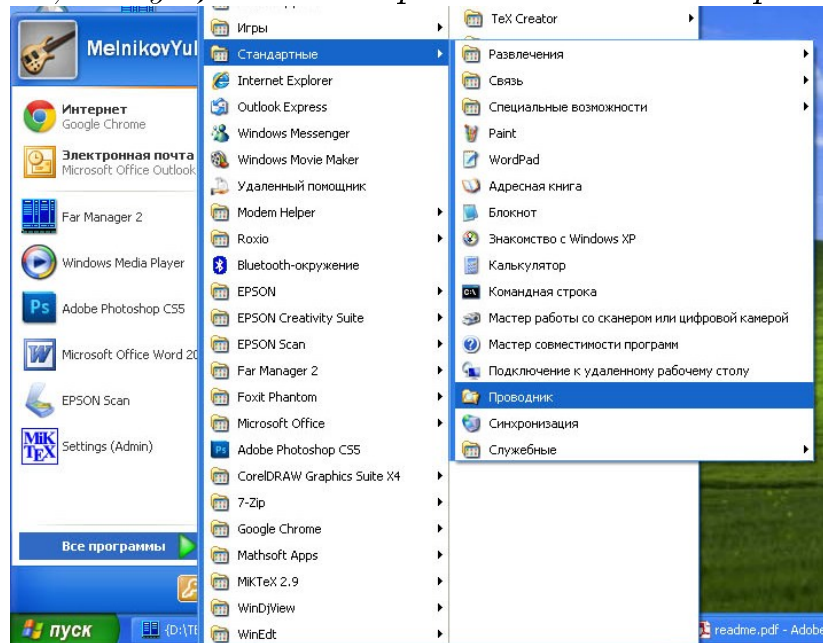
II.2.1. Установка MikTeX2.9

Следует учесть, что под Windows-7 папка Program Files\MikTeX2.9\tex должна быть открыта в проводнике, запущенном с правами администратора.

Для Windows следует после установки MikTeX2.9 скопировать в Program Files\MikTeX2.9\tex содержимое папки tex\TeXSample.

II.2.1. Установка MikTeX2.9

Следует учесть, что под Windows-7 папка Program Files\MikTeX2.9\tex должна быть открыта в проводнике, запущенном с правами администратора.

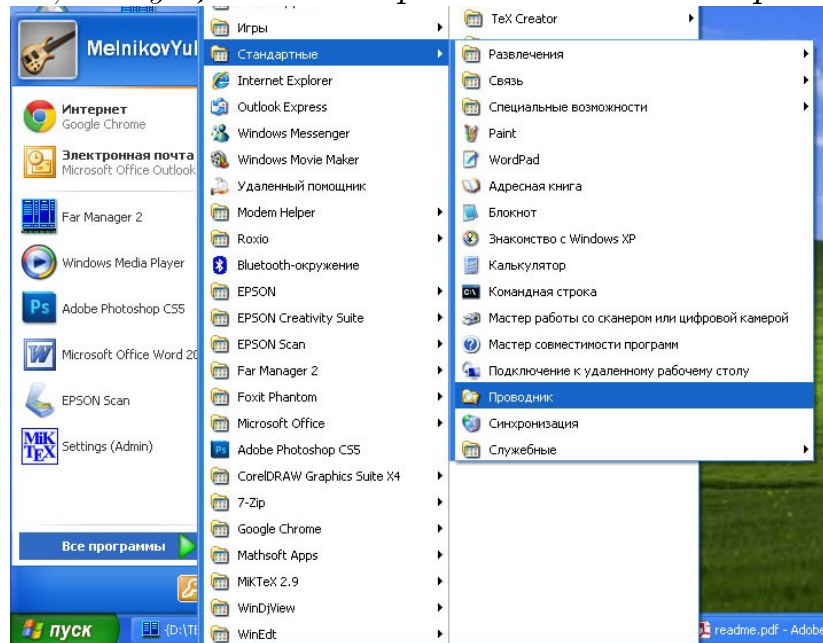


Для этого следует
правой **кнопкой**
мышки кликнуть на
Проводник:

Пуск→Все программы→
→Стандартные→
→Проводник

II.2.1. Установка MikTeX2.9

Следует учесть, что под Windows-7 папка Program Files\MikTeX2.9\tex должна быть открыта в проводнике, запущенном с правами администратора.



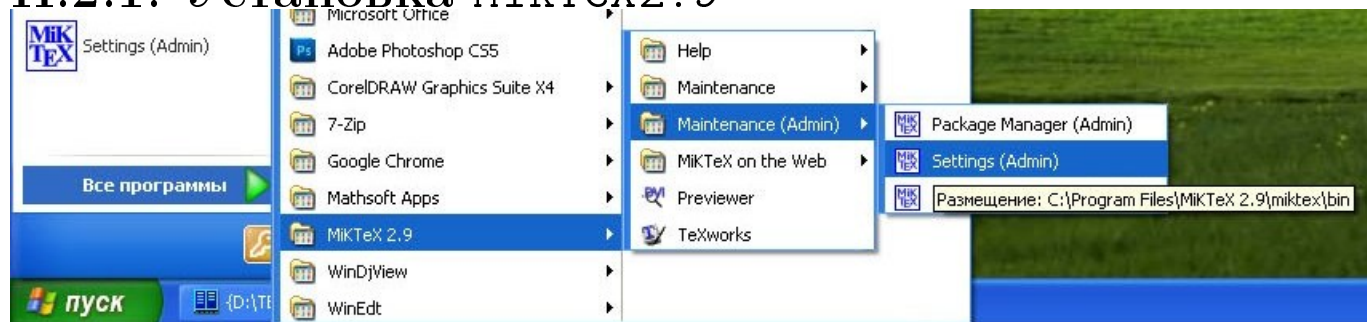
Для этого следует **правой кнопкой** мышки кликнуть на Проводник и выбрать в появившемся окне «Запуск от имени администратора».

II.2.1. Установка MikTeX2.9

Для Windows следует после установки MikTeX2.9 скопировать в Program Files\MikTeX2.9\tex содержимое папки tex\TeXSample.

Для Linux после установки TeXLive следует скопировать содержимое папки tex\TeXSample в usr/share/texlive/texmf/tex, например, с помощью GNOME Commander, запущенным с правами root.

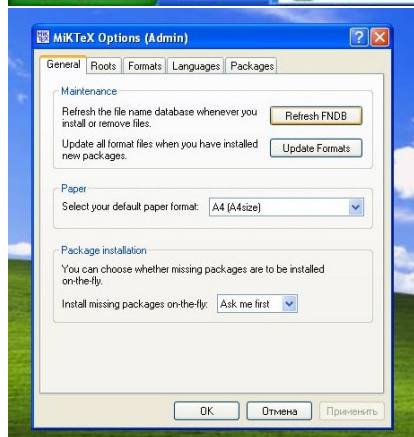
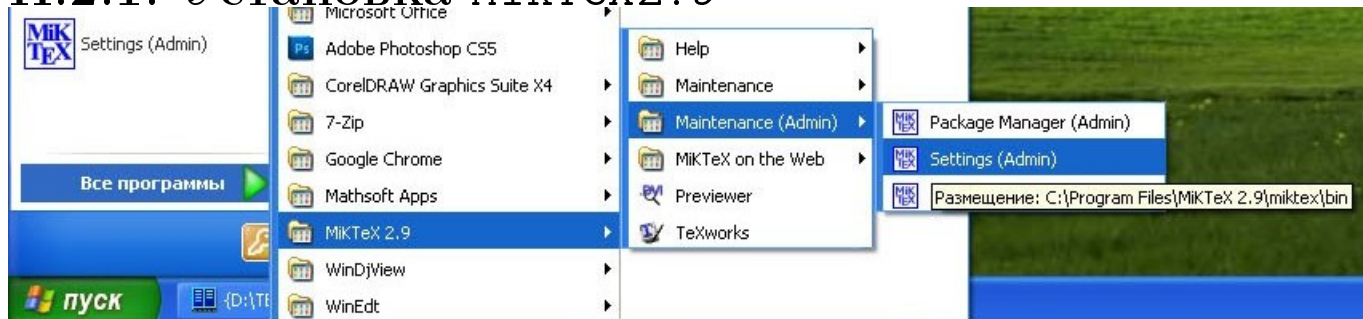
II.2.1. Установка MikTeX2.9



Следует настроить компилятор MikTeX, для чего следует запустить Settings (Admin):

Пуск→Все программы →MikTeX 2.9 →Maintance (Admin)→
→Settings (Admin)

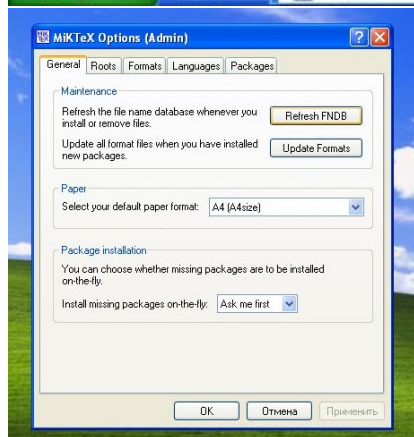
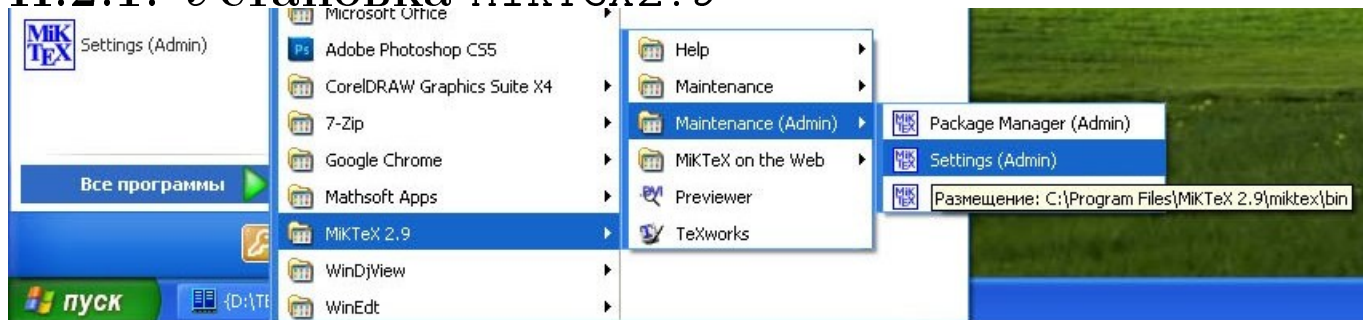
II.2.1. Установка MikTeX2.9



В окне, открывшемся после запуска Settings, следует кликнуть левой кнопки мышки сначала на Refresh FNDB...

Следует подождать, это потребует времени...

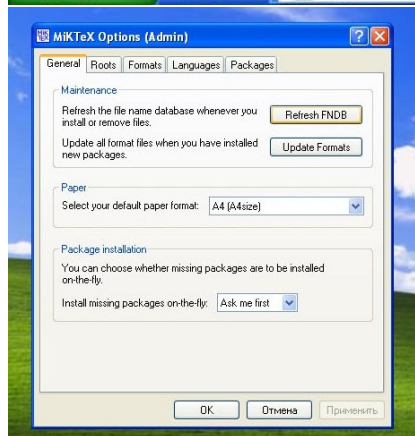
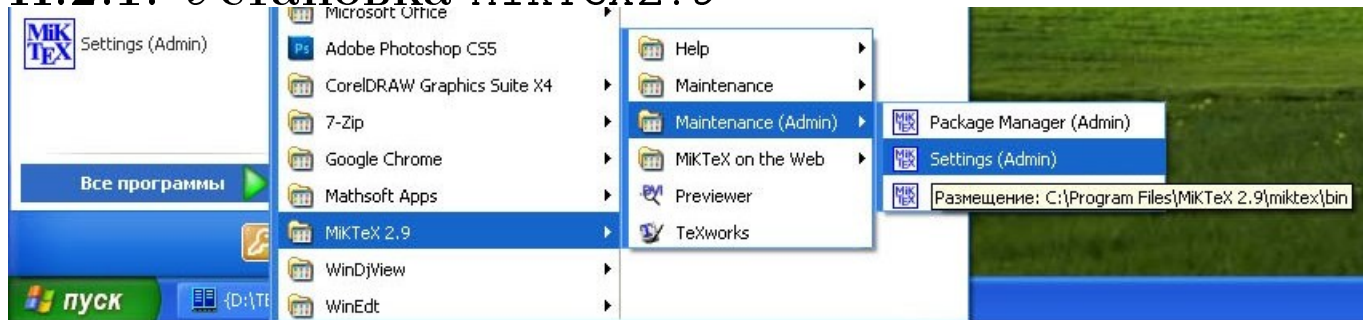
II.2.1. Установка MikTeX2.9



В окне, открывшемся после запуска Settings, следует кликнуть левой кнопки мышки сначала на Refresh FNDB, а потом Update Formats.

Это тоже потребует времени...

II.2.1. Установка MikTeX2.9



В окне, открывшемся после запуска Settings, следует кликнуть левой кнопки мышки сначала на

Refresh FNDB, а потом Update Formats.

Теперь L^AT_EX настроен. Осталось установить редактор (для редактирования кода L^AT_EX).

II.2.2. Установка TeXCreator

Широко распространены разные редакторы: WinEdit, TeXMaker, Emacs и др.

Для использования системы «сэмплов» мы рекомендуем редактор TeXCreator, разработанный А. Богдановым.

II.2.2. Установка TeXCreator

Широко распространены разные редакторы: WinEdit, TeXMaker, Emacs и др.

Для использования системы «сэмплов» мы рекомендуем редактор TeXCreator, разработанный А. Богдановым.

В папке TeXCreator находится
папка с программой установки TeXCreator для Windows
папка с программой установки TeXCreator для Linux.

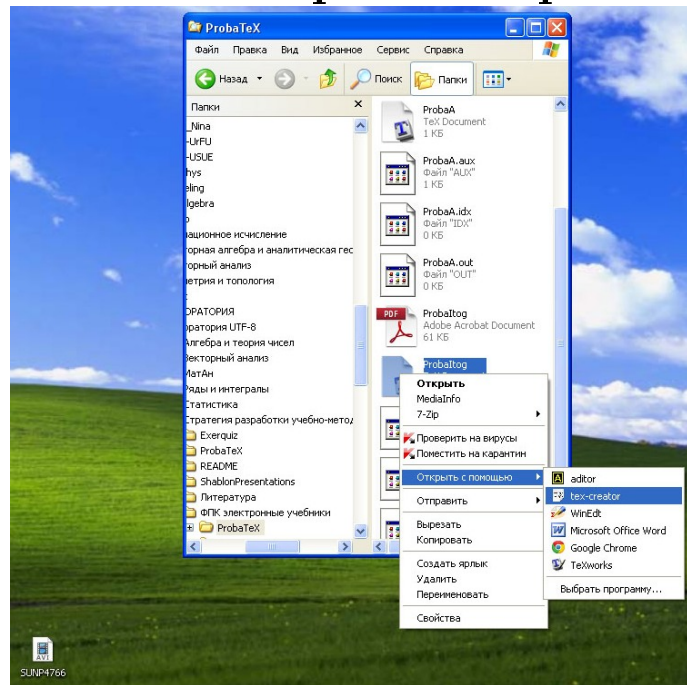
II.2.3. Настройка и работа в TeXCreator

Инструкция по работе в редакторе TeXCreator и его настройке можно получить после запуска редактора, нажав клавишу F1.

II.2.3. Настройка и работа в TeXCreator

Настройте запуск TeXCreator
«по умолчанию» для файлов
с расширением `tex`, `inp`, `sty`,
`pic`.

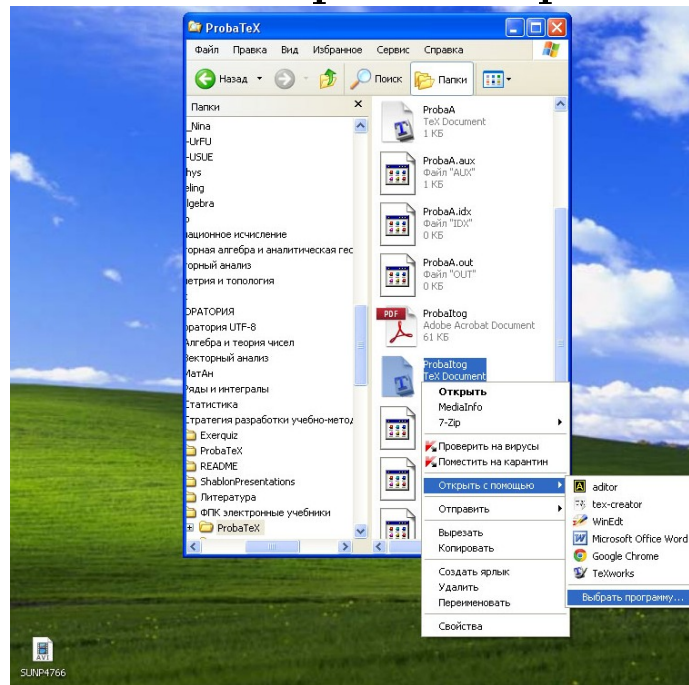
II.2.3. Настройка и работа в TeXCreator



Настройте запуск `texcreator` «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Для этого надо надо навести указатель мыши на файл одного из этих типов, «кликнуть» *правой* кнопкой мышки, выбрать в выпавшем списке «Открыть с помощью», потом `TeXCreator`, если он уже есть в списке.

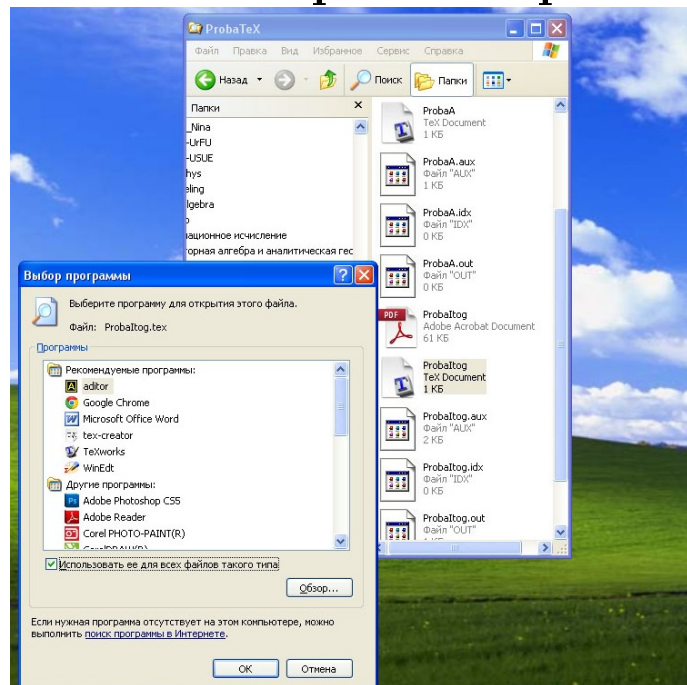
II.2.3. Настройка и работа в TeXCreator



Настройте запуск `texcreator` «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Если `TeXCreator` в этом списке отсутствует, кликнем левой кнопкой мыши на «Выбрать программу»...

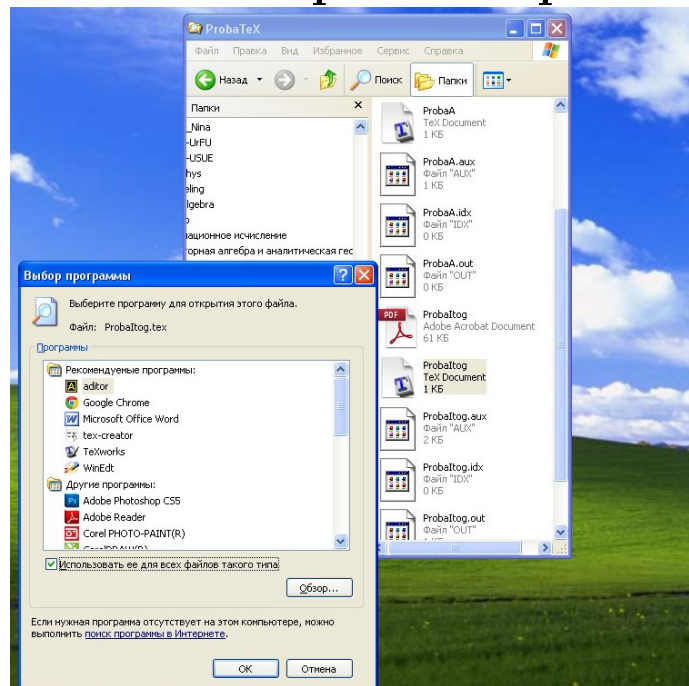
II.2.3. Настройка и работа в TeXCreator



Настройте запуск `texcreator` «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Если `TeXCreator` в этом списке отсутствует, кликнем левой кнопкой мыши на «Выбрать программу», поставим галочку в квадратике «Использовать для всех файлов такого типа».

II.2.3. Настройка и работа в TeXCreator

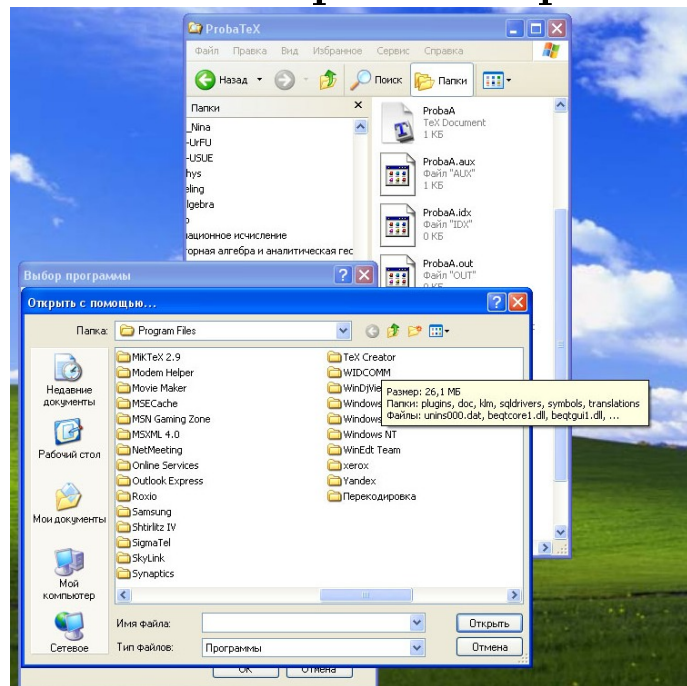


Настройте запуск texcreator «по умолчанию» для файлов с расширением tex, inp, sty, pic.

Если TeXCreator в этом списке отсутствует, кликнем левой кнопкой мыши на «Выбрать программу», поставим галочку в квадратике «Использовать для всех файлов такого типа».

Теперь кликнем левой кнопкой мыши на пункт «Обзор»,

II.2.3. Настройка и работа в TeXCreator

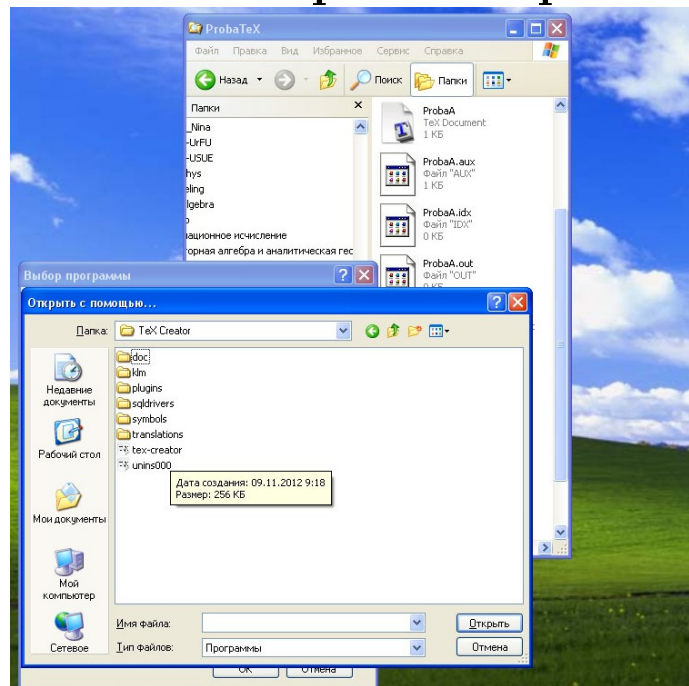


Настройте запуск texcreator «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Если TeXCreator в этом списке отсутствует, кликнем левой кнопкой мыши на «Выбрать программу», поставим галочку в квадратике «Использовать для всех файлов такого типа».

Теперь кликнем левой кнопкой мыши на пункт «Обзор», выберем папку TeX Creator,

II.2.3. Настройка и работа в TeXCreator

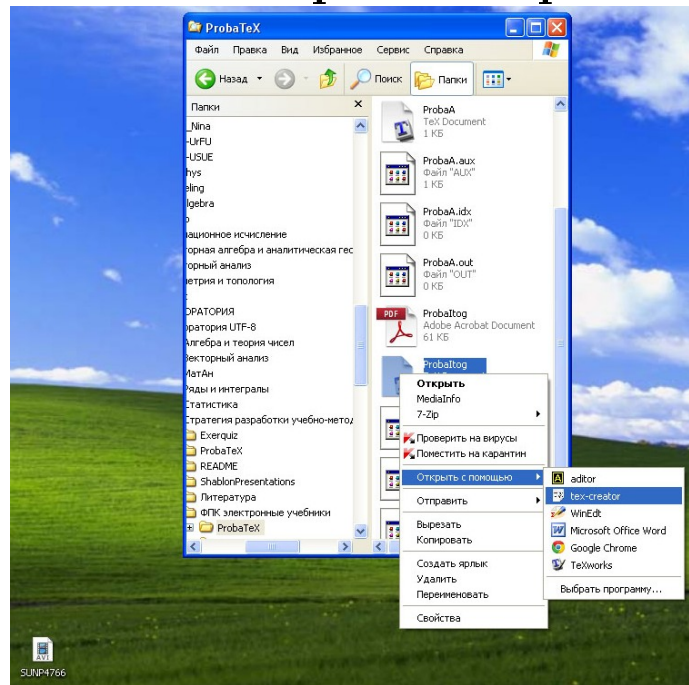


Настройте запуск `texcreator` «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Если `TeXCreator` в этом списке отсутствует, кликнем левой кнопкой мыши на «Выбрать программу», поставим галочку в квадратике «Использовать для всех файлов такого типа».

Теперь кликнем левой кнопкой мыши на пункт «Обзор», выберем папку `TeX Creator`, и в ней программу `texcreator`...

II.2.3. Настройка и работа в TeXCreator



Настройте запуск `texcreator` «по умолчанию» для файлов с расширением `tex`, `inp`, `sty`, `pic`.

Эту процедуру надо провести для всех перечисленных типов файлов.

II.2.4. Система PreT_EX

L_AT_EX является системой логического, а не визуального программирования.

Это значит, что функция просмотра отделена от функции редактирования.

II.2.4. Система PreT_EX

Подготовка документа в издательской системе L^AT_EX представляет собой процесс написания программного кода.



II.2.4. Система PreT_EX

Подготовка документа в издательской системе L^AT_EX представляет собой процесс написания программного кода.

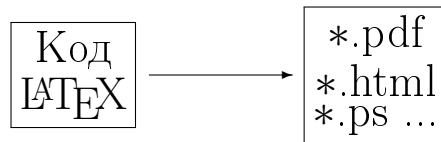
В результате компиляции этого кода получается документ в одном из распространённых форматов.



II.2.4. Система PreTeX

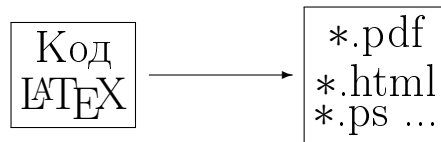
Подготовка документа в издательской системе $\text{\LaTeX}$ представляет собой процесс написания программного кода.

В результате компиляции этого кода получается документ в одном из распространённых форматов.



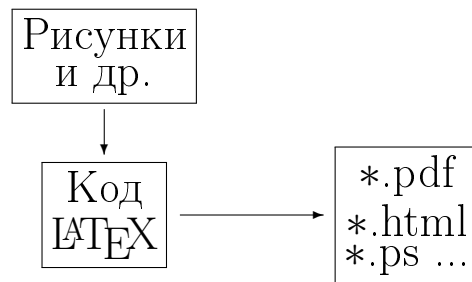
II.2.4. Система PreTeX

В исходный код можно вставлять ссылки, например, на файлы с рисунками, представленными в различных форматах.



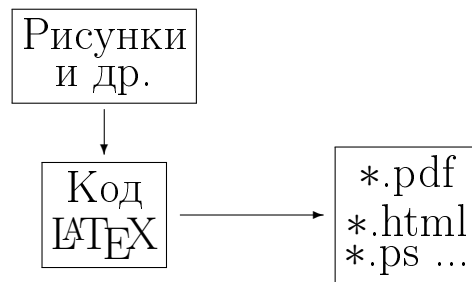
II.2.4. Система PreT_EX

В исходный код можно вставлять ссылки, например, на файлы с рисунками, представленными в различных форматах.



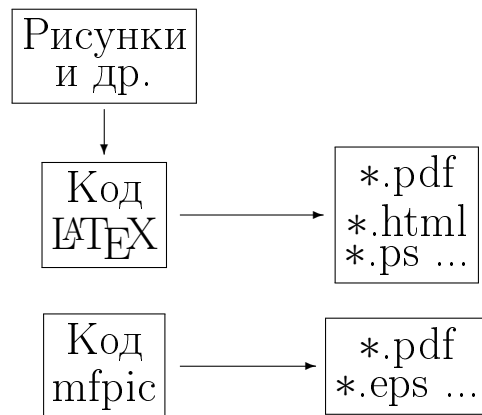
II.2.4. Система PreT_EX

Графики, чертежи и некоторые рисунки удобно готовить с помощью дополнительных средств издательской системы L^AT_EX.



II.2.4. Система PreTeX

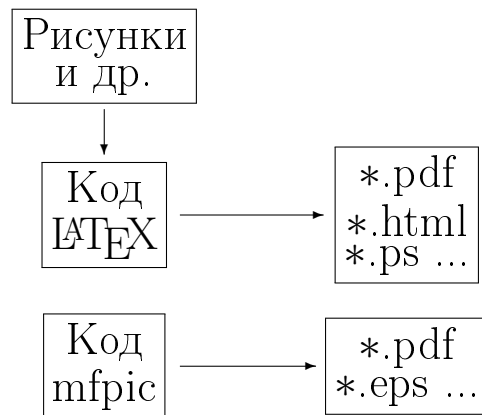
Графики, чертежи и некоторые рисунки удобно готовить с помощью дополнительных средств издательской системы L^AT_EX.



II.2.4. Система PreTeX

Графики, чертежи и некоторые рисунки удобно готовить с помощью дополнительных средств издательской системы L^AT_EX.

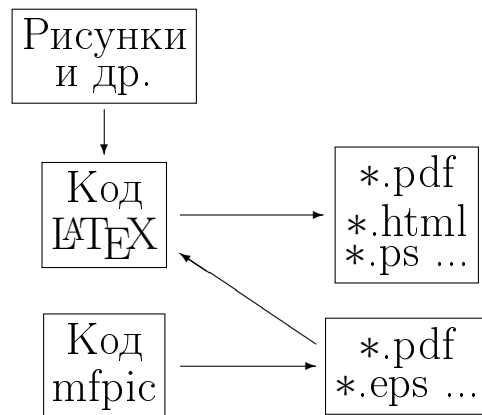
Ссылки на полученные файлы можно внедрять в код L^AT_EX.



II.2.4. Система PreTeX

Графики, чертежи и некоторые рисунки удобно готовить с помощью дополнительных средств издательской системы L^AT_EX.

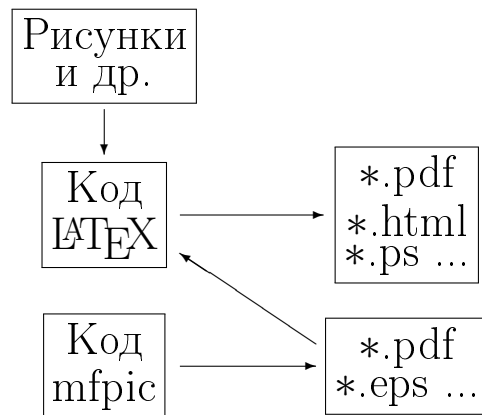
Ссылки на полученные файлы можно внедрять в код L^AT_EX.



II.2.4. Система PreTeX

Нередко возникает потребность генерирования больших объемов рутинного кода L^AT_EX:

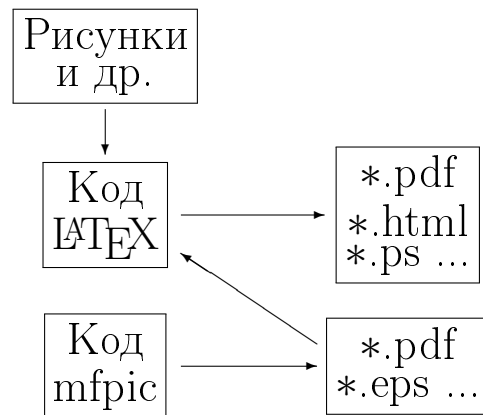
генерация большого числа вариантов заданий, массивы команд для построения графиков и т.п.



II.2.4. Система PreTeX

Нередко возникает потребность генерирования больших объемов рутинного кода $\text{\LaTeX}$:

генерация большого числа вариантов заданий, массивы команд для построения графиков и т.п.

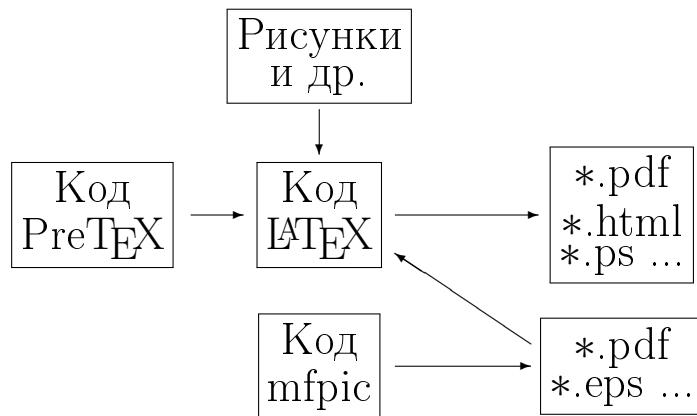


Для генерирования больших объемов рутинного кода $\text{\LaTeX}$ нами разработана система PreTeX.

II.2.4. Система PreTeX

Нередко возникает потребность генерирования больших объемов рутинного кода $\text{\LaTeX}$:

генерация большого числа вариантов заданий, массивы команд для построения графиков и т.п.

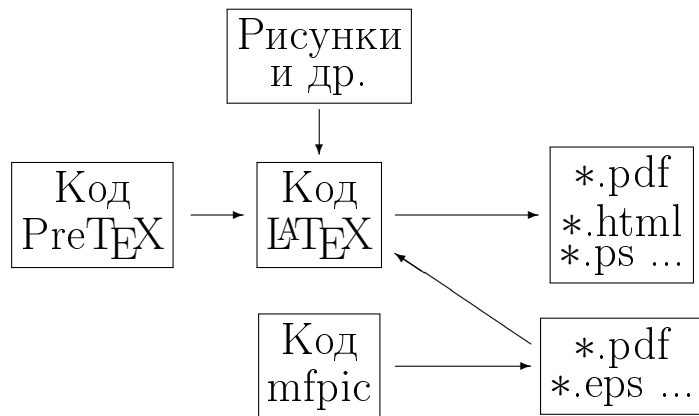


Для генерирования больших объемов рутинного кода $\text{\LaTeX}$ нами разработана система PreTeX.

II.2.4. Система PreTeX

Нередко возникает потребность генерирования больших объемов рутинного кода L^AT_EX:

генерация большого числа вариантов заданий, массивы команд для построения графиков и т.п.



Для генерирования больших объемов рутинного кода L^AT_EX нами разработана система PreTeX.

Система PreTeX является составной частью созданной нами системы TeXCreator:

редактор кода L^AT_EX, система для работы с сэмплами.

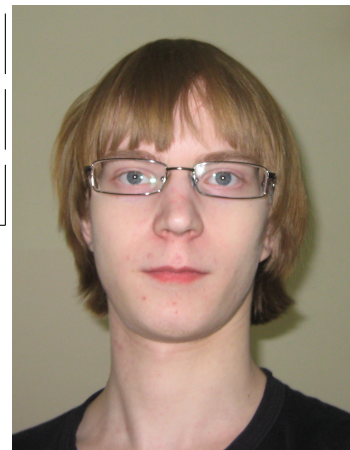
II.2.5. Система TeXSample

Разработанная нами система PreTeX также, по-видимому, не имеет прямых аналогов.

II.2.5. Система TeXSample

Сервис на www.usue.ru

Преподаватель,
разработчик
ЭОР

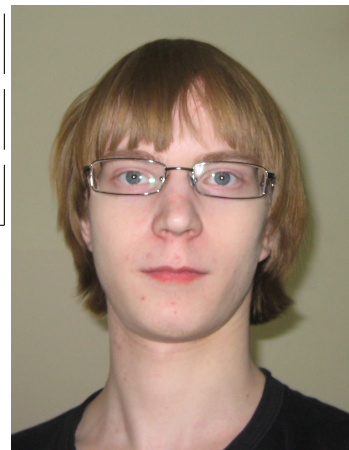


II.2.5. Система TeXSample

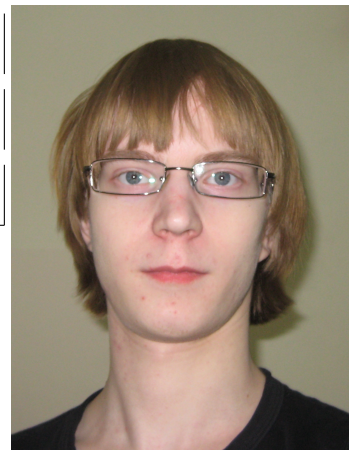
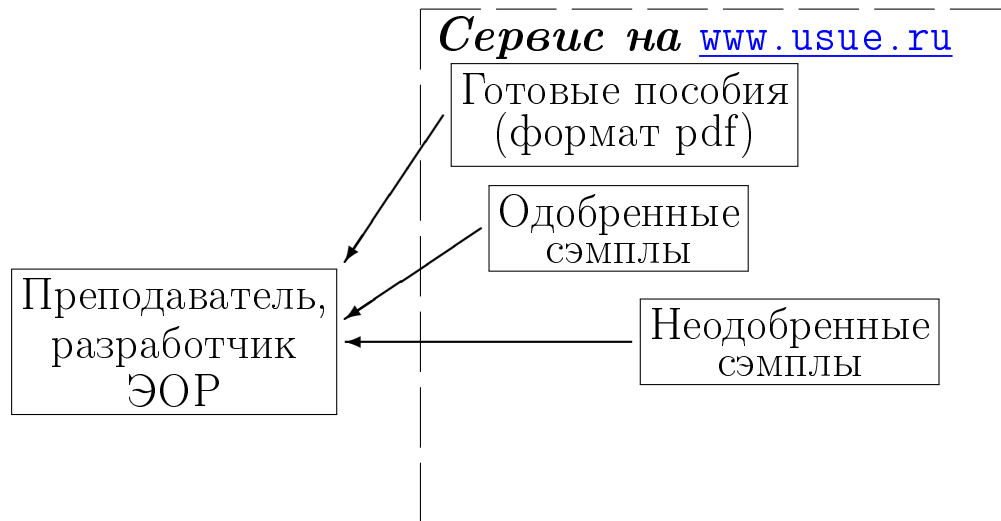
Сервис на www.usue.ru

Готовые пособия
(формат pdf)

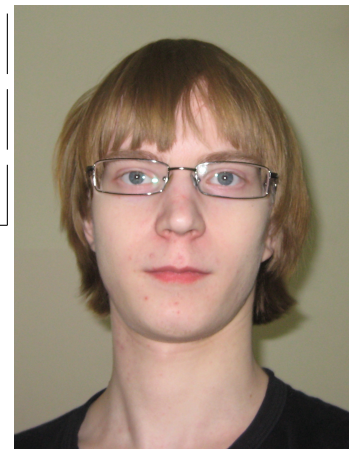
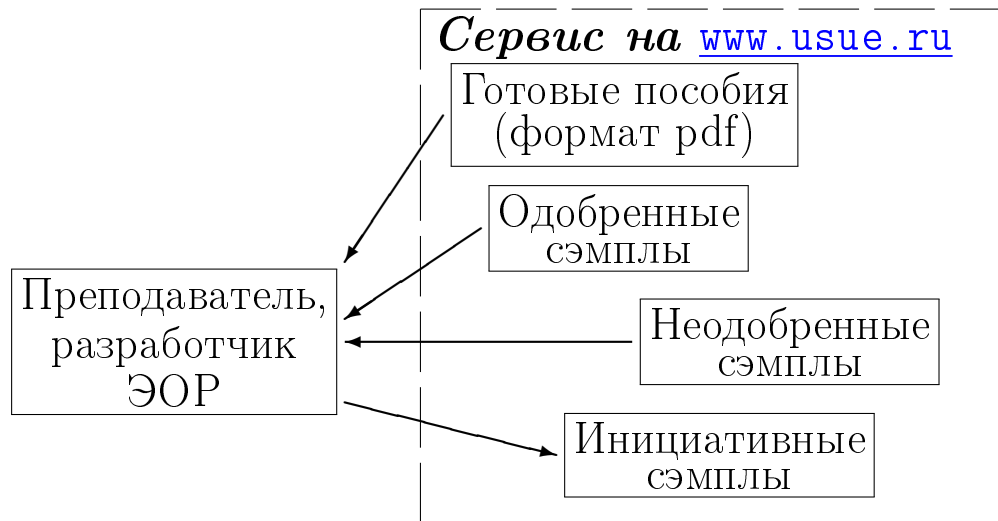
Преподаватель,
разработчик
ЭОР



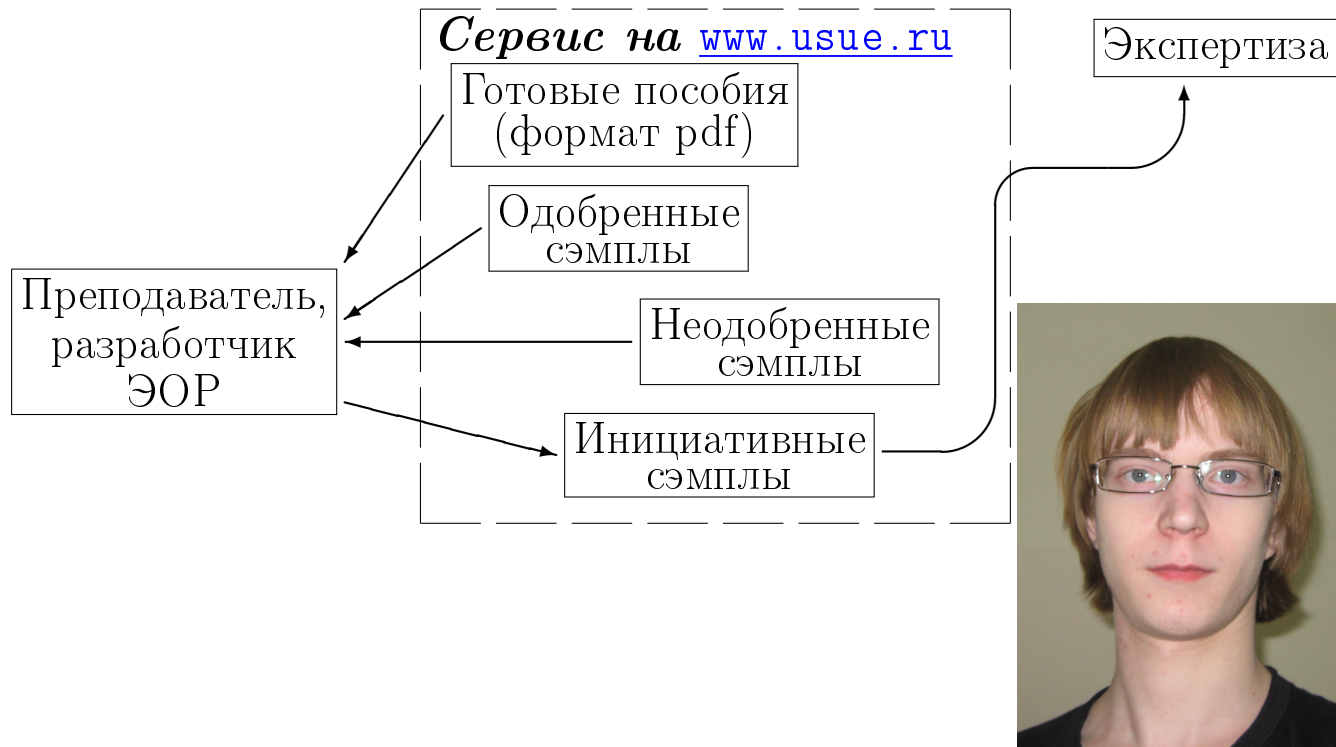
II.2.5. Система TeXSample



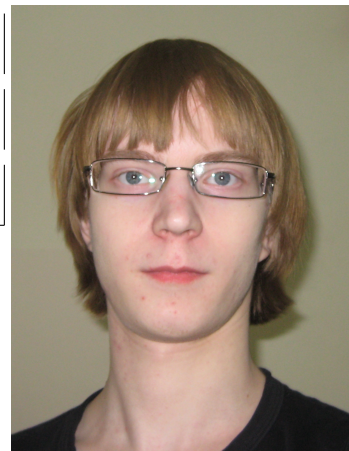
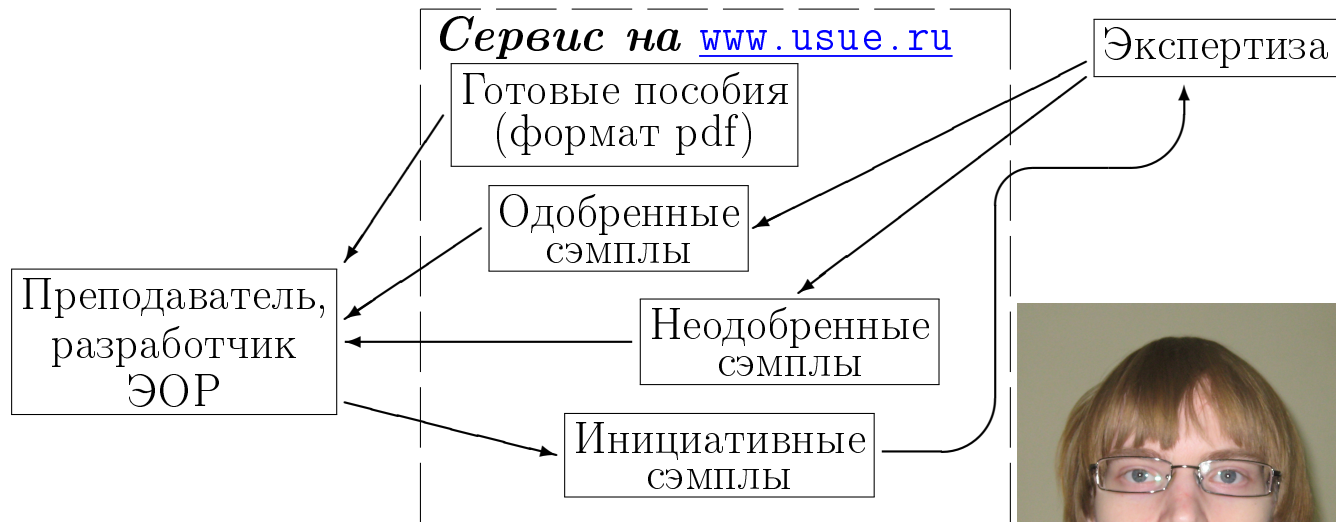
II.2.5. Система TeXSample



II.2.5. Система TeXSample



II.2.5. Система TeXSample



II.2.6. Система TeXCreator

Рассмотрим функционально-структурную схему системы TeXCreator.

II.2.6. Система TeXCreator



TeXCreator

II.2.6. Система TeXCreator

Система
TeXCreator
разработана
коллективом
в составе:

Программист

Андрей Олегович Богданов (ЭМА-10)

Аналитик и заказчик:

Мельников Юрий Борисович



TeXCreator

II.2.6. Система TeXCreator

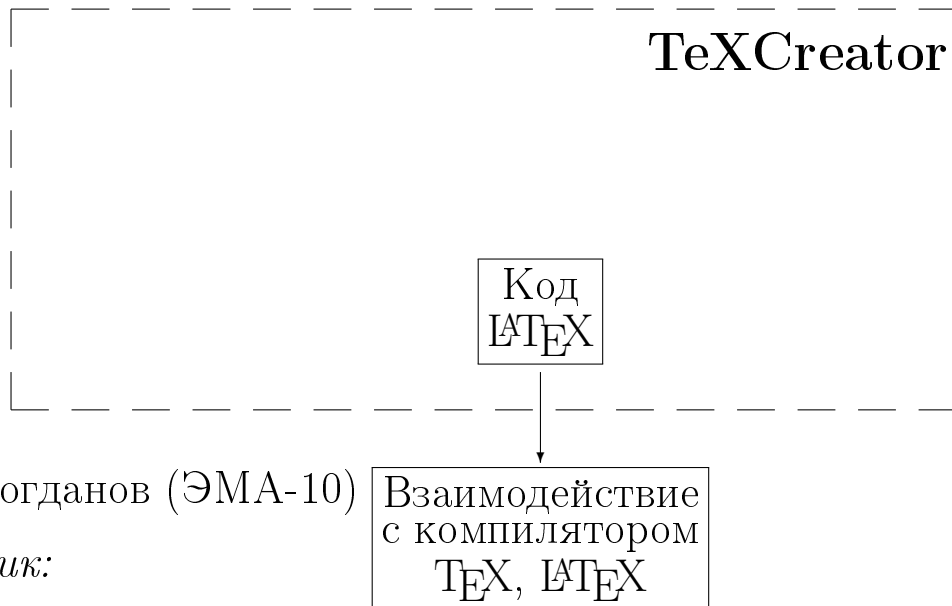
Система
TeXCreator
разработана
коллективом
в составе:

Программист

Андрей Олегович Богданов (ЭМА-10)

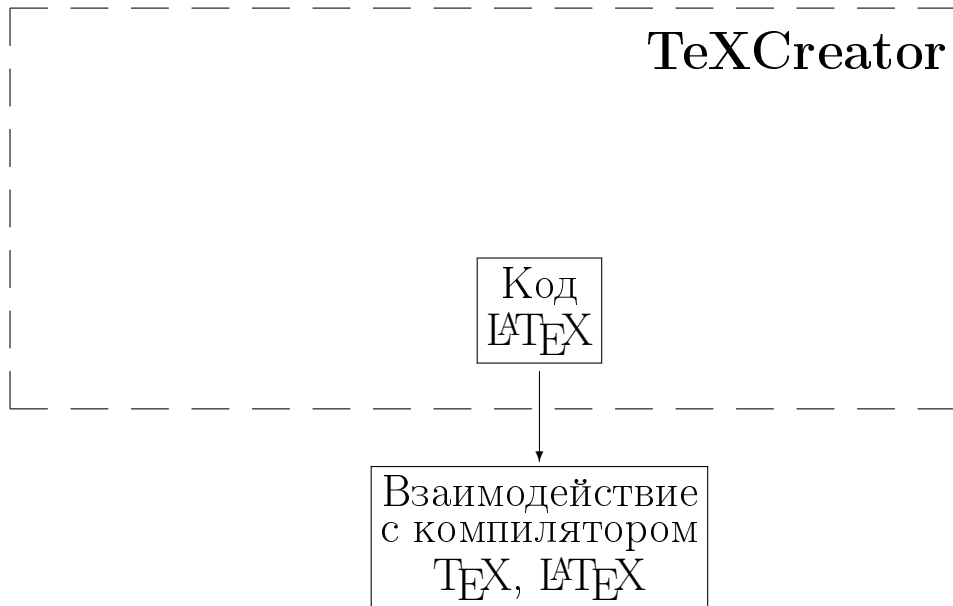
Аналитик и заказчик:

Мельников Юрий Борисович



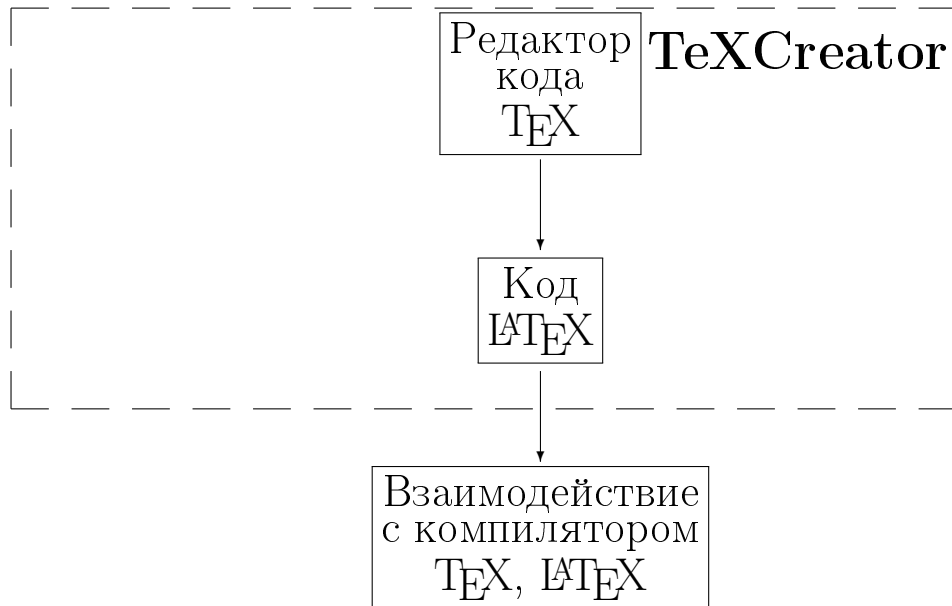
II.2.6. Система TeXCreator

Используются
стандартные компиляторы $\text{T}_{\text{E}}\text{X}$.



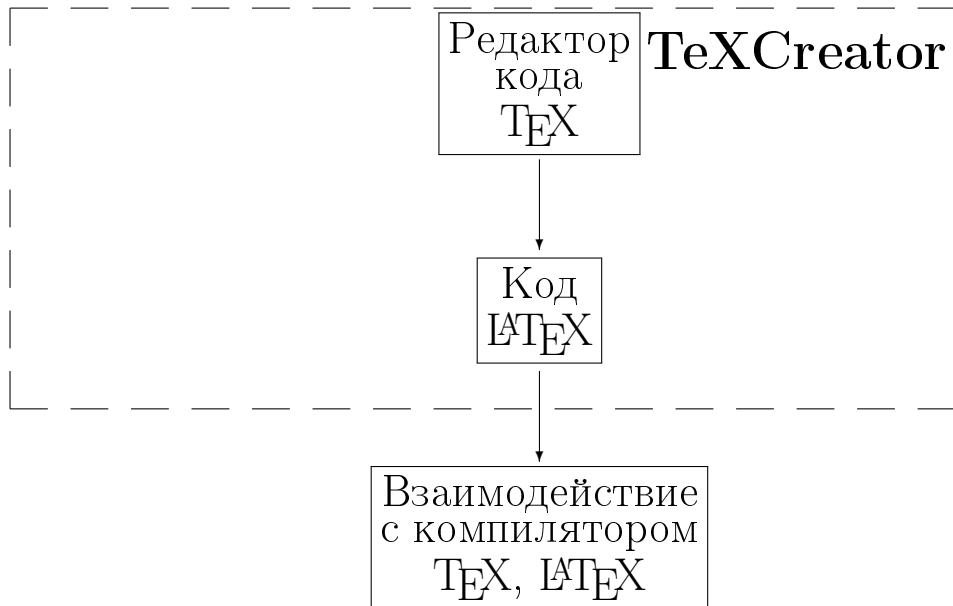
II.2.6. Система TeXCreator

Используются
стандартные компиляторы $\text{\TeX}$.



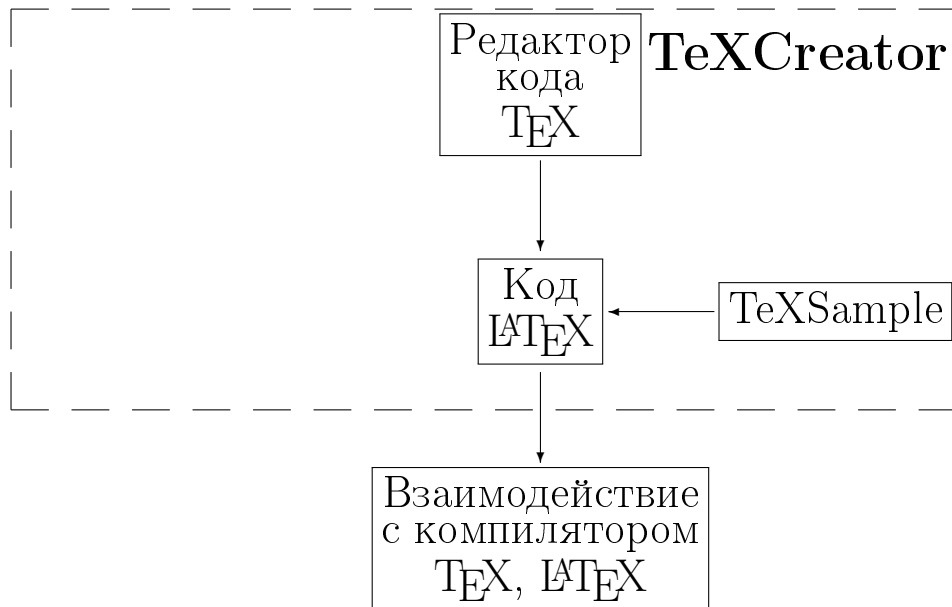
II.2.6. Система TeXCreator

Существует большое число редакторов кода $\text{\TeX}$. Данный редактор удобен и, как минимум, не уступает по удобству другим популярным системам: WinEdit, TeXMaker и др.



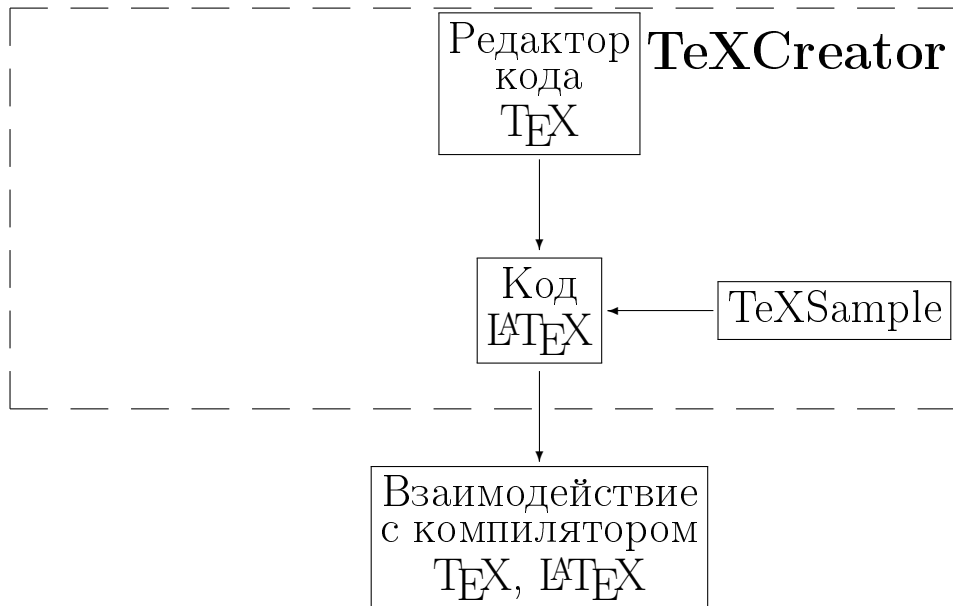
II.2.6. Система TeXCreator

Существует большое число редакторов кода $\text{\TeX}$. Данный редактор удобен и, как минимум, не уступает по удобству другим популярным системам: WinEdit, TeXMaker и др.



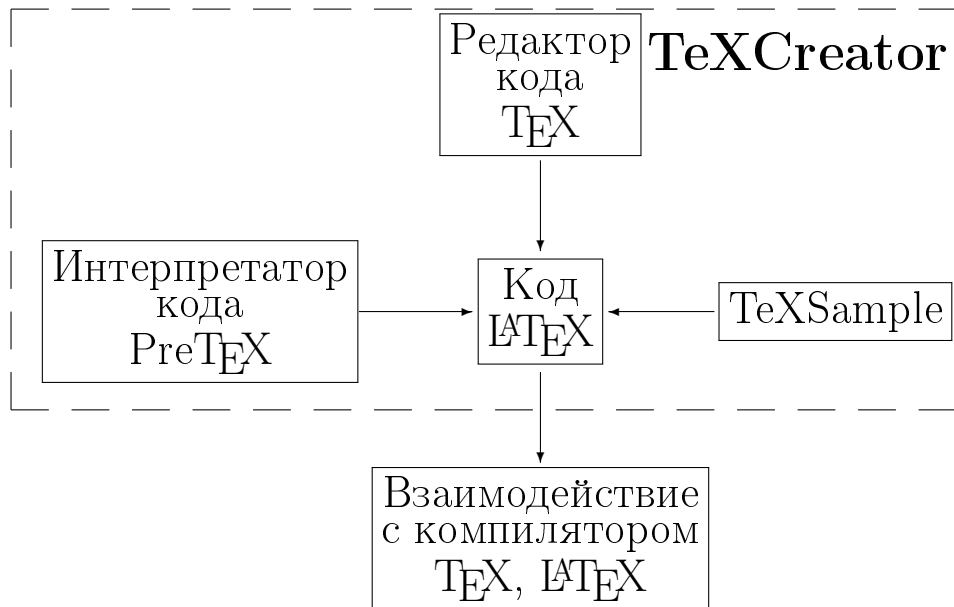
II.2.6. Система TeXCreator

Система TeXSample не имеет прямых аналогов, она разработана тем же коллективом.



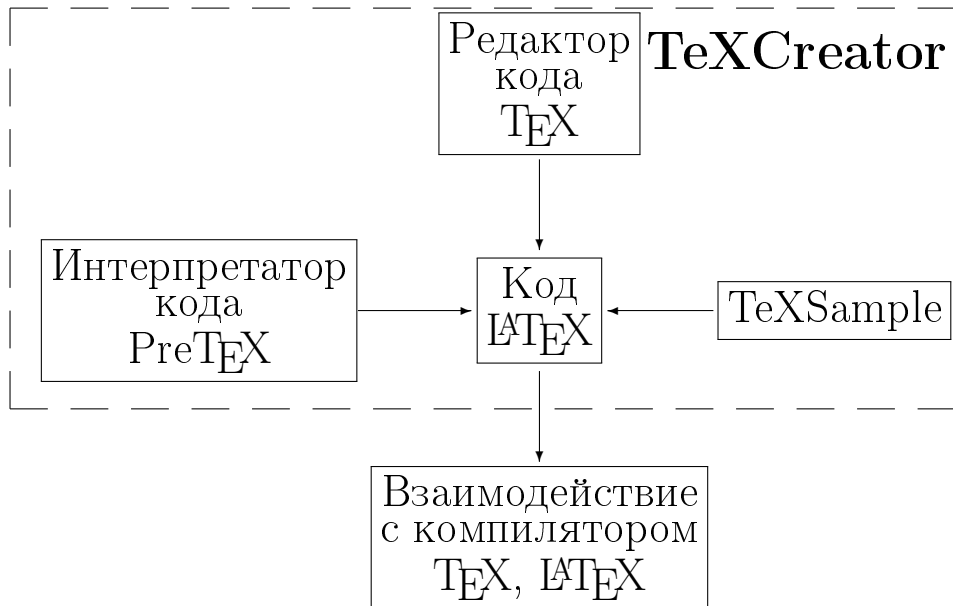
II.2.6. Система TeXCreator

Система TeXSample не имеет прямых аналогов, она разработана тем же коллективом.



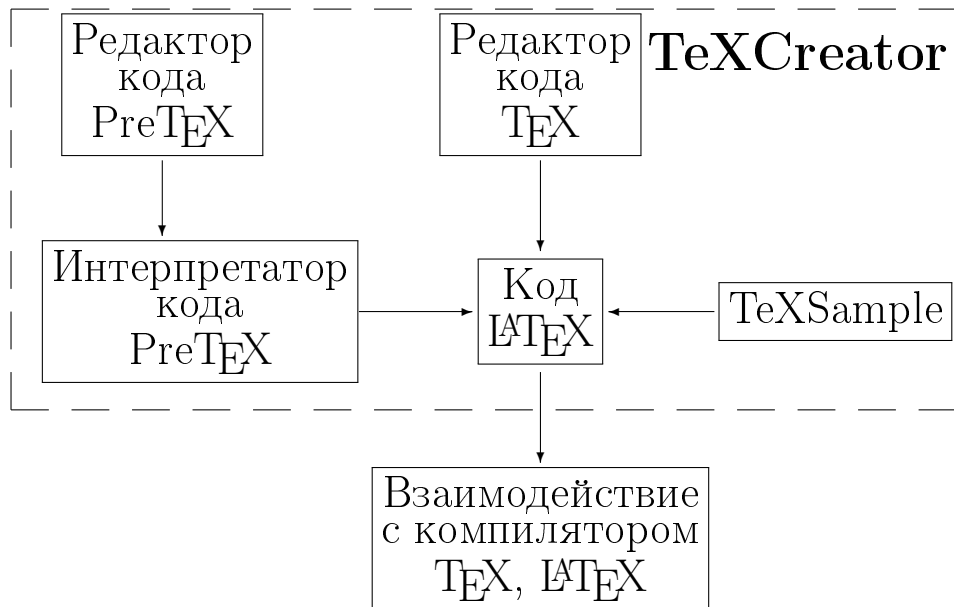
II.2.6. Система TeXCreator

Система PreTeX также, по-видимому, не имеет прямых аналогов, она разработана тем же коллективом.



II.2.6. Система TeXCreator

Система PreTeX также, по-видимому, не имеет прямых аналогов, она разработана тем же коллективом.



II.3. Создание и компиляция файла *.tex

Создать файл, например, `Proba.tex` можно с помощью файлового менеджера (например, `Far`) или с помощью редактора `texcreator`: откройте `texcreator` и воспользуйтесь инструкцией, вызываемой с помощью `F1`.

II.3.1. Компиляция файла *.tex

Для того, чтобы получить на экране компьютера или напечатанным на принтере текст, предназначенный для читателя, необходимо выполнить компиляцию соответствующего файла *.tex с помощью программы `pdflatex` (здесь мы рассматриваем только интересующий нас вариант компиляции).

II.3. Компиляция файла *.tex

Исходный код,
файл
*.tex

II.3. Компиляция файла *.tex

Исходный код,
файл
*.tex

Компиляция кода

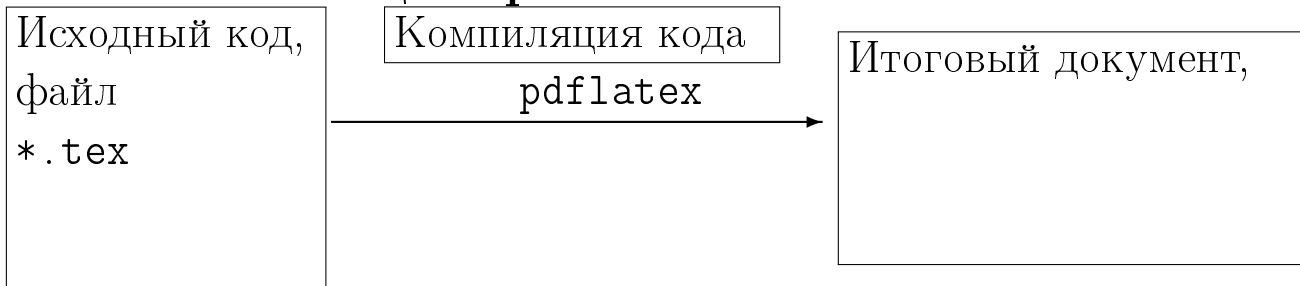
II.3. Компиляция файла *.tex

Исходный код,
файл
*.tex

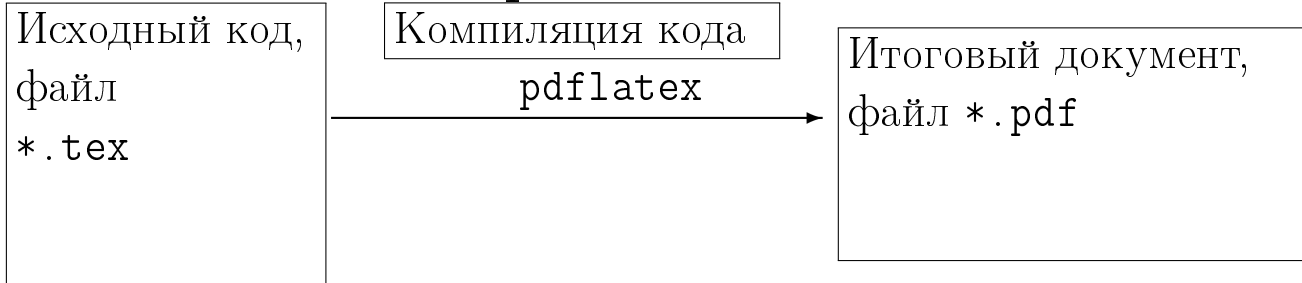
Компиляция кода

pdflatex

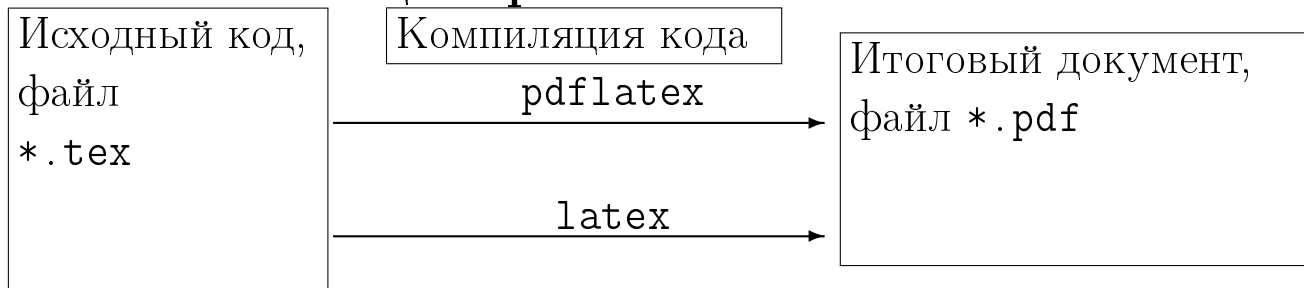
Итоговый документ,



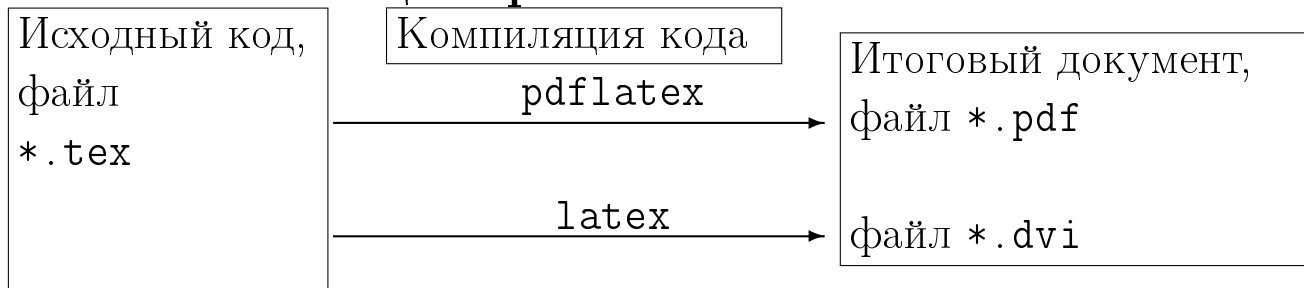
II.3. Компиляция файла *.tex



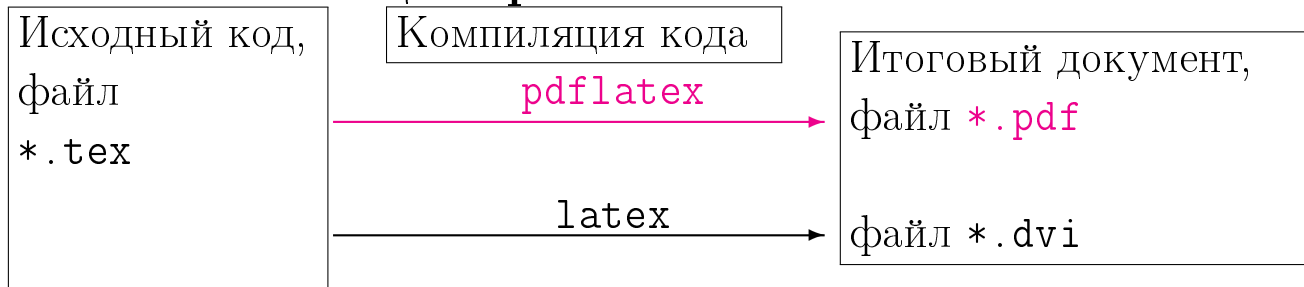
II.3. Компиляция файла *.tex



II.3. Компиляция файла *.tex



II.3. Компиляция файла *.tex



Нас в данном случае интересует получение документа в формате pdf.

II.3. Компиляция файла *.tex

Исходный код,
файл
`Proba.tex`

Допустим, был подготовлен код в файле `Proba.tex`.

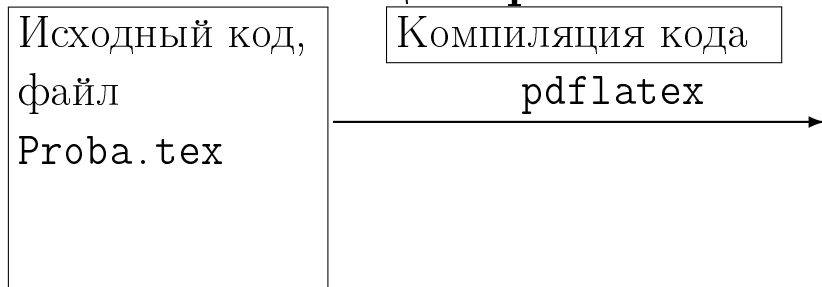
II.3. Компиляция файла *.tex

Исходный код,
файл
Proba.tex

Допустим, был подготовлен код в файле Proba.tex.

Код запустили на компиляцию командой `pdflatex Proba.tex`.

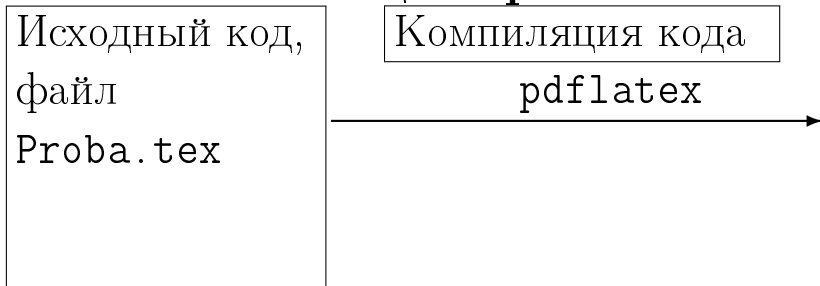
II.3. Компиляция файла *.tex



Допустим, был подготовлен код в файле `Proba.tex`.

Код запустили на компиляцию командой `pdflatex Proba.tex`.

II.3. Компиляция файла *.tex

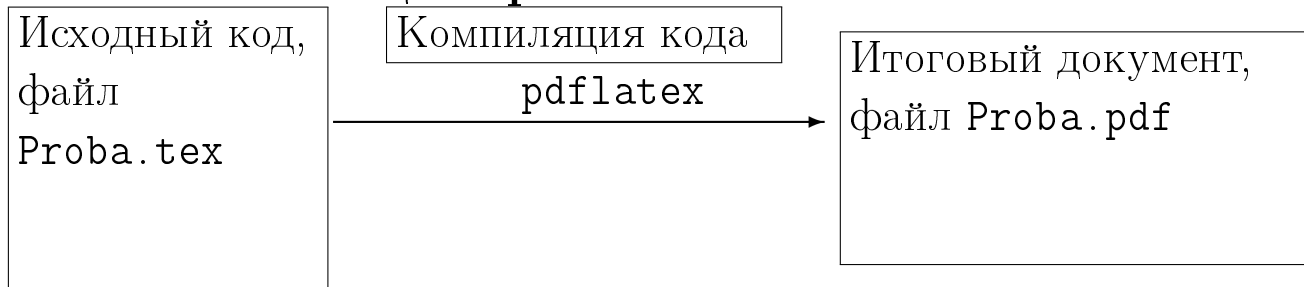


Допустим, был подготовлен код в файле Proba.tex.

Код запустили на компиляцию командой `pdflatex Proba.tex`.

После компиляции получили файл Proba.pdf.

II.3. Компиляция файла *.tex



Допустим, был подготовлен код в файле `Proba.tex`.

Код запустили на компиляцию командой `pdflatex Proba.tex`.

После компиляции получили файл `Proba.pdf`.

II.3. Компиляция файла *.tex

Исходный код,
файл
`Proba.tex`

Итоговый документ,
файл `Proba.pdf`

Допустим, был подготовлен код в файле `Proba.tex`.

Код запустили на компиляцию командой `pdflatex Proba.tex`.

После компиляции получили файл `Proba.pdf`.

Теперь имеем исходный код `Proba.tex` и результат компиляции `Proba.pdf`.

Рассмотрим пример?

II.3. Компиляция файла *.tex

Исходный код,
файл
Proba.tex

Итоговый документ,
файл Proba.pdf

Предположим, в исходный код внесли изменения.

II.3. Компиляция файла *.tex

Исходный код,
файл
Proba.tex
Изменения

Итоговый документ,
файл Proba.pdf

Предположим, в исходный код внесли изменения.

II.3. Компиляция файла *.tex

Исходный код,
файл
Proba.tex
Изменения

Итоговый документ,
файл Proba.pdf

Предположим, в исходный код внесли изменения.

Итоговый файл остаётся прежним!

II.3. Компиляция файла *.tex

Исходный код,
файл
`Proba.tex`
Изменения

Итоговый документ,
файл `Proba.pdf`

Предположим, в исходный код внесли изменения.

Чтобы отразить изменения в итоговом файле, следует вновь запустить файл `Proba.tex` на компиляцию.

II.3. Компиляция файла *.tex

Исходный код,
файл

Proba.tex

Изменения

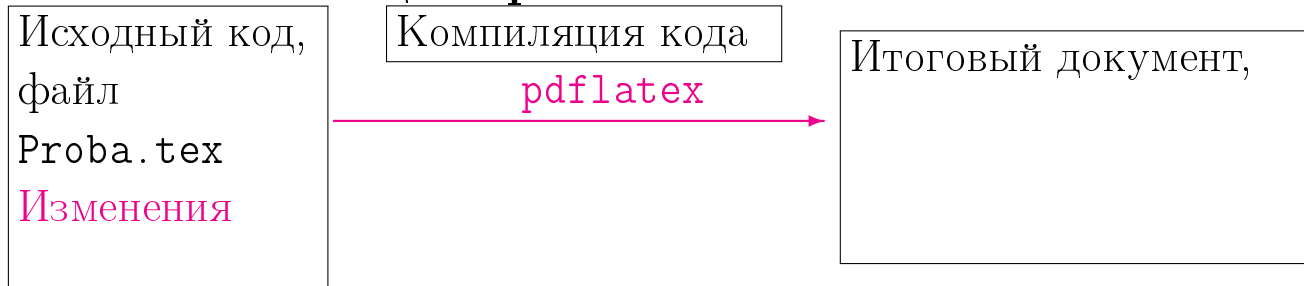
Компиляция кода

Итоговый документ,
файл Proba.pdf

Предположим, в исходный код внесли изменения.

Чтобы отразить изменения в итоговом файле, следует вновь запустить файл Proba.tex на компиляцию.

II.3. Компиляция файла *.tex

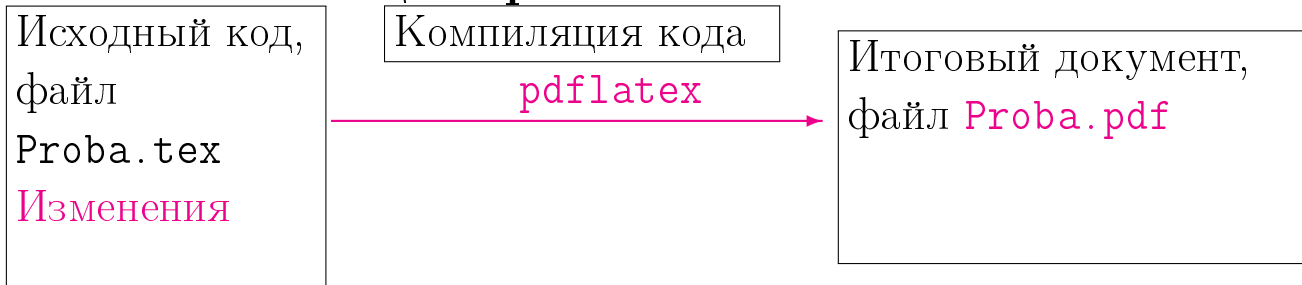


Предположим, в исходный код внесли изменения.

Чтобы отразить изменения в итоговом файле, следует вновь запустить файл `Proba.tex` на компиляцию.

В процессе компиляции прежний файл `Proba.pdf` уничтожается, и заменяется на новый.

II.3. Компиляция файла *.tex



Предположим, в исходный код внесли изменения.

Чтобы отразить изменения в итоговом файле, следует вновь запустить файл Proba.tex на компиляцию.

В процессе компиляции прежний файл Proba.pdf уничтожается, и заменяется на новый.

II.3. Компиляция файла *.tex

Исходный код,
файл
`Proba.tex`
Изменения

Итоговый документ,
файл `Proba.pdf`

Предположим, в исходный код внесли изменения.

Чтобы отразить изменения в итоговом файле, следует вновь запустить файл `Proba.tex` на компиляцию.

В процессе компиляции прежний файл `Proba.pdf` уничтожается, и заменяется на новый.

Итак, изменения в итоговый файл `Proba.pdf` вносятся не сразу после изменения исходного файла `Proba.tex`, а только *после компиляции*!

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если ошибка относительно небольшая: слово «вылезает» за пределы абзаца (например, потому, что его не удаётся перенести) и т.п., то

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если ошибка относительно небольшая: слово «вылезает» за пределы абзаца (например, потому, что его не удаётся перенести) и т.п., то после обнаружения ошибки соответствующее замечание записывается в протокол компиляции, который выводится на экран и в файл `*.log`,

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если ошибка относительно небольшая: слово «вылезает» за пределы абзаца (например, потому, что его не удаётся перенести) и т.п., то после обнаружения ошибки соответствующее замечание записывается в протокол компиляции, который выводится на экран и в файл `*.log`, и компиляция продолжается.

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если ошибка относительно небольшая: слово «вылезает» за пределы абзаца (например, потому, что его не удаётся перенести) и т.п., то после обнаружения ошибки соответствующее замечание записывается в протокол компиляции, который выводится на экран и в файл `*.log`, и компиляция продолжается.

Вывод отладочной информации в `*.log` можно осуществить командой `\typeout{...}`, где троеточием обозначен текст, который будет записан в файл `*.log`.

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если же ошибка серьёзная, то дальнейшие действия зависят от среды компиляции:

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если же ошибка серьёзная, то дальнейшие действия зависят от среды компиляции:

либо информация об ошибке записывается в файл `*.log` и работа компилятора прекращается;

II.3.2. Реакция на ошибки компиляции

Если в файле `*.tex` была допущена ошибка, реакция компилятора зависит от качества ошибки.

Если же ошибка серьёзная, то дальнейшие действия зависят от среды компиляции:

либо информация об ошибке записывается в файл `*.log` и работа компилятора прекращается;

либо компиляция приостанавливается, и компьютер переходит в режим диалога с пользователем.

II.3.2. Реакция на ошибки компиляции

Если же ошибка серьёзная, то дальнейшие действия зависят от среды компиляции:

либо информация об ошибке записывается в файл *.log и работа компилятора прекращается;

либо компиляция приостанавливается, и компьютер переходит в режим диалога с пользователем.

В последнем случае с клавиатуры можно ввести

- либо x — закрывается программа компиляции;
- либо **Ctrl**+ z — закрывается программа компиляции;
- либо r — в этом случае компиляция продолжается, программа исправляет ошибки «по своему усмотрению». Если компилятор не в состоянии исправить все ошибки, программа-компилятор закрывается.

Рассмотрим пример?

II.3.3. Файлы, создаваемые в процессе компиляции

Написанный вами текст следует рассматривать как программный код. Предполагается, что в **рассмотренном нами примере** файл с этим кодом называется `Proba.tex`.

II.3.3. Файлы, создаваемые в процессе компиляции

Написанный вами текст следует рассматривать как программный код. Предполагается, что в **рассмотренном нами примере** файл с этим кодом называется `Proba.tex`.

В процессе компиляции ведётся протокол, записываемый в файл с тем же именем, что и исходный файл `*.tex`, но с расширением `*.log`.

В **рассмотренном нами примере** этот файл называется `Proba.log`.

II.3.3. Файлы, создаваемые в процессе компиляции

Написанный вами текст следует рассматривать как программный код. Предполагается, что в **рассмотренном нами примере** файл с этим кодом называется `Proba.tex`.

В процессе компиляции ведётся протокол, записываемый в файл с тем же именем, что и исходный файл `*.tex`, но с расширением `*.log`.

В **рассмотренном нами примере** этот файл называется `Proba.log`.

Файл `*.log` (например, `Proba.log`) можно просмотреть любым редактором ASCII-кодов.

II.3.3. Файлы, создаваемые в процессе компиляции

Написанный вами текст следует рассматривать как программный код. Предполагается, что в **рассмотренном нами примере** файл с этим кодом называется `Proba.tex`.

В процессе компиляции создаются также файлы с расширениями `aux`, `toc` и `ind`, в которые записывается информация о метках, представленных Вами в тексте (коде), об элементах оглавления и об элементах предметного указателя.

II.3.3. Файлы, создаваемые в процессе компиляции

Написанный вами текст следует рассматривать как программный код. Предполагается, что в **рассмотренном нами примере** файл с этим кодом называется `Proba.tex`.

В процессе компиляции создаются также файлы с расширениями `aux`, `toc` и `ind`, в которые записывается информация о метках, представленных Вами в тексте (коде), об элементах оглавления и об элементах предметного указателя.

Процесс компиляции начинается с чтения файлов `*.aux` и `*.toc`, поэтому всегда выставляется оглавление и метки, оставшиеся от предыдущей компиляции.

Рассмотрим пример?

II.4. Структура файла *.tex

В исходном файле *.tex можно выделить два основных блока:
преамбулу и **основной текст**.

II.4. Структура файла *.tex

В исходном файле *.tex можно выделить два основных блока: **преамбулу** и **основной текст**.

```
\documentclass{article}
\usepackage[dvipdfm]{graphicx}
\usepackage{curves}
\usepackage{amsfonts}
\usepackage{amsmath}
```

...

...

```
\begin{document}
```

Издательская система $\text{\LaTeX}$ 2 ϵ обладает мощными средствами для набора формул, таблиц. $\text{\LaTeX}$ 2 ϵ приспособлен для подготовки структурированного текста. Оглавление и предметный указатель формируются и обновляются автоматически при каждой компиляции файла *.tex.

```
\end{document}
```

II.4. Структура файла *.tex

В исходном файле *.tex можно выделить два основных блока:
преамбулу и **основной текст**.

преамбула	{	<pre>\documentclass{article} \usepackage[dvipdfm]{graphicx} \usepackage{curves} \usepackage{amsfonts} \usepackage{amsmath}</pre>
основной текст	{	<pre>\begin{document} Издательская система $\TeX$ 2ϵ обладает мощными средствами для набора формул, таблиц. $\TeX$ 2ϵ приспособлен для подготовки структурированного текста. Оглавление и предметный указатель формируются и обновляются автоматически при каждой компиляции файла *.tex. \end{document}</pre>

II.4.1. Преамбула

В преамбуле содержатся команды на подключение стилевых файлов, некоторые команды, форматы представления и др.

II.4.1. Преамбула

В преамбуле содержатся команды на подключение стилевых файлов, некоторые команды, форматы представления и др.

Для презентаций учебного назначения нами разработана типовая преамбула, содержащаяся в файле `texsample.tex`. Для ее подключения достаточно создаваемый вами файл начать командой

```
\input texsample.tex
```

Рассмотрим пример?

II.4.2. Основной текст

В дальнейшем мы бóльшую часть времени посвятим именно подготовке основного текста.

II.4.2. Основной текст

В дальнейшем мы бóльшую часть времени посвятим именно подготовке основного текста.

Мы рассмотрим структурирование текста: выделение разделов и подразделов, создание оглавления, предметного указателя, гиперссылок, использование так называемых окружений и др.

II.4.2. Основной текст

В дальнейшем мы бóльшую часть времени посвятим именно подготовке основного текста.

Мы рассмотрим структурирование текста: выделение разделов и подразделов, создание оглавления, предметного указателя, гиперссылок, использование так называемых окружений и др.

Мы подготовили шаблон для создания презентаций к лекциям:
`00Lectures.tex`,

а также шаблон для создания презентаций к практическим занятиям:

`PractShablon.tex`

III. Набор текста, символов и скобок

Компилятор $\text{\TeX}$ по-разному обрабатывает, например, текст естественного языка и математические формулы.

Например, выражение

$2x+1$ обработано как текст;

$2x + 1$ обработано как формула.

III. Набор текста, символов и скобок

Компилятор $\text{\TeX}$ по-разному обрабатывает, например, текст естественного языка и математические формулы.

Компилятор $\text{\TeX}$ во время работы в каждый момент находится в определённых модах (режимах компиляции), например, горизонтальной или вертикальной, текстовой или математической.

Перейти в математическую моду можно, например, с помощью символа $\$$.

Повторное применение символа $\$$ приводит к возвращению в текстовую моду.

III.1. Абзац и новая строка

Компилятор $\text{\TeX}$ по-разному обрабатывает, например, текст естественного языка и математические формулы.

Компилятор $\text{\TeX}$ во время работы в каждый момент находится в определённых модах (режимах компиляции), например, горизонтальной или вертикальной, текстовой или математической.

Перейти к новому абзацу можно только в текстовой моде. Для этого достаточно вставить в текст пустую строку или команду `\par`. Подробнее форматирование текста рассмотрено в **разделе IX.2.**

III.1. Абзац и новая строка

Компилятор $\text{\TeX}$ по-разному обрабатывает, например, текст естественного языка и математические формулы.

Компилятор $\text{\TeX}$ во время работы в каждый момент находится в определённых модах (режимах компиляции), например, горизонтальной или вертикальной, текстовой или математической.

Перейти к новой строке в текстовой моде можно с помощью команды `\newline`.

III.2. Символы и многострадальная буква ё

При компиляции печатаются обычным образом:

- все буквы английского и почти все буквы русского алфавита;
- цифры от 0 до 9;
- знаки пунктуации:
: ; ! ? () [] ‘ ’ - * / . ,

Даже в текстовой моде можно использовать обычным образом символы $+$ и $=$, но лучше применять их в математической моде.

Мода	Код	Результат компиляции
Текстовая	2+3=5	2+3=5
Математическая	\$2+3=5\$	$2 + 3 = 5$

III.2. Символы и многострадальная буква ё

Символы \$ # % & печатают так: \\$ \# \% \&.

Символ % $\text{\TeX}$ воспринимает как знак комментария, т.е. все символы в строке, находящиеся после символа %, $\text{\TeX}$ не воспринимает.

Символ \$ $\text{\TeX}$ воспринимает как сигнал для перехода в математическую моду из текстовой или в текстовую моду из математической.

III.2. Символы и многострадальная буква ё

Символы \$ # % & печатают так: \\$ \# \% \&.

Символ % $\TeX$ воспринимает как знак комментария, т.е. все символы в строке, находящиеся после символа %, $\TeX$ не воспринимает.

Символ \$ $\TeX$ воспринимает как сигнал для перехода в математическую моду из текстовой или в текстовую моду из математической.

Специфично кратное использование символа -:

- для «-» (дефис), -- для «—», --- для «—» (тире).

Мы используем кодировку UTF-8, в которой буква ё компилируется корректно. Но если ваш $\TeX$ -овский код придётся перекодировать и использовать, например, в стандартной кодировке Windows cp1251, то результат компиляции буквы ё может быть неверным. В этом случае следует писать ещё `\symbol{188}` или ещё `\{"e}` для получения после компиляции слова «ещё». **Рассмотрим пример?**

III.3. Скобки

Обычным образом печатаются скобки: (,), [,], /.

III.3. Скобки

Обычным образом печатаются скобки: (,), [,], /.

Код для печати фигурных скобок { } таков: \{ \}.

Фигурные скобки, набранные обычным образом, имеют особый смысл: {фигурные скобки выделяют так называемую группу.}

Изменения параметров текста (например, размер и начертание шрифта, его цвет и др.), произведённые внутри группы, обычно по выходе из него возвращаются к исходным значениям.

Например, действие команд указания размера шрифта

\Huge и др. прекращается по выходу из группы:

{\Huge Текст} {\huge Текст}

{\LARGE Текст} {\Large Текст} {\large Текст}

{\normalsize Текст} {\small Текст}.

III.3. Скобки

Обычным образом печатаются скобки: (,), [,], /.

Код для печати фигурных скобок { } таков: \{ \}.

Фигурные скобки, набранные обычным образом, имеют особый смысл: {фигурные скобки выделяют так называемую группу.}

В математической моде применяются также команды `\langle` для скобки $\langle$ и `\rangle` для скобки $\rangle$.

`\langle abcd\rangle` даёт следующий результат компиляции:

III.3. Скобки

Обычным образом печатаются скобки: (,), [,], /.

Код для печати фигурных скобок { } таков: \{ \}.

Фигурные скобки, набранные обычным образом, имеют особый смысл: {фигурные скобки выделяют так называемую группу.}

В математической моде применяются также команды `\langle` для скобки $\langle$ и `\rangle` для скобки $\rangle$.

`\langle abcd\rangle` даёт следующий результат компиляции: $\langle abcd \rangle$.

Важным инструментом являются **растягивающиеся скобки**.

III.4. Начертание и размер шрифта

Доступны два типа простых команд изменения шрифта.

III.4. Начертание и размер шрифта

Доступны два типа простых команд изменения шрифта.

Старый способ

Новый способ

`{\rm Текст}`

`\textrm{Текст}`

`{\it Текст}`

`\textit{Текст}`

`{\sf Текст}`

`\textsf{Текст}`

`{\sl Текст}`

`\textsl{Текст}`

`{\tt Текст}`

`\texttt{Текст}`

III.4. Начертание и размер шрифта

Доступны два типа простых команд изменения шрифта.

Старый способ	Новый способ
<code>{\rm Текст}</code>	<code>\textrm{Текст}</code>
<code>{\it Текст}</code>	<code>\textit{Текст}</code>
<code>{\sf Текст}</code>	<code>\textsf{Текст}</code>
<code>{\sl Текст}</code>	<code>\textsl{Текст}</code>
<code>{\tt Текст}</code>	<code>\texttt{Текст}</code>

Разница проявляется при сочетании этих команд: старый тип команды отменяет действие предыдущей, а новый тип добавляет своё действие к действию предыдущей.

Компиляция кода `{\tt abc{\it def}}` даёт *abcdef*, а кода `\texttt{abc\textit{def}}` даёт *abcdef*.

III.4. Начертание и размер шрифта

Размер шрифта указывается с помощью команд

`{\Huge` Самый большой}, `{\huge` поменьше},
`{\LARGE` ещё меньше}, `{\Large` ещё меньше}, `{\large` ещё меньше},
`{\normalsize` ещё меньше}, `{\small` ещё меньше}.

Фигурные скобки здесь применены для ограничения области действия команд.

III.5. Пробелы

Код	Результат компиляции	
$a\qquad b$ $a\quad b$ $a\ b$ $a\;b$ $a\colon b$ $a\,b$ ab $a!b$	$a\quad b$ $a\quad b$ $a\ b$ $a\ b$ $a\ b$ $a\ b$ ab ab	В математической моде пробелы между символами расставляются автоматически, но данные команды позволяют корректировать эти пробелы.
$a\ b$ $a\quad b$ $a\quad b$ $a\quad b$	$a\ b$ ab ab ab	Кратные пробелы (двойной, тройной и др.) в файле <code>*.tex</code> при компиляции игнорируются.

III.5. Пробелы

Код	Результат компиляции	
<code>a\qqquad b</code>	<i>a b</i>	В текстовой моде можно использовать «неразрывный» пробел: знак ~. Например, компиляции фрагмента кода <code>число~2</code> даёт: «число 2», причём символ 2 не может быть перенесен на другую строку отдельно от слова «число».
<code>a\quad b</code>	<i>a b</i>	
<code>a\ b</code>	<i>a b</i>	
<code>a\;b</code>	<i>a b</i>	
<code>a\:b</code>	<i>a b</i>	
<code>a\,b</code>	<i>a b</i>	
<code>ab</code>	<i>ab</i>	
<code>a\!b</code>	<i>ab</i>	
<code>a b</code>	<i>a b</i>	Кратные пробелы (двойной, тройной и др.) в файле <code>*.tex</code> при компиляции игнорируются.
<code>a b</code>	<i>ab</i>	
<code>a b</code>	<i>ab</i>	
<code>a b</code>	<i>ab</i>	

IV. Набор математических формул

Для печати математических формул и математических символов необходимо перейти из *текстовой моды* в *математическую моду*.

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

— с помощью одиночного знака \$;

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

— с помощью одиночного знака \$;

Повторное применение этого знака возвращает в текстовую моду.

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

— с помощью одиночного знака \$;

Например, компиляция фрагмента

выражение $2x-1$ отличается от $2x-1$.

даёт следующий результат:

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

— с помощью одиночного знака \$;

Например, компиляция фрагмента

выражение $2x-1$ отличается от $2x-1$.

даёт следующий результат:

выражение $2x-1$ отличается от $2x-1$.

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;

Это означает, что формула будет напечатана в отдельной строке, выровненная по центру.

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;

Это означает, что формула будет напечатана в отдельной строке, выровненная по центру.

Повторное применение \$\$ возвращает в текстовую моду.

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;

Например, компиляция фрагмента

«выражение $2x-1$ отличается от $2x-1$.»

даёт следующий результат:

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;

Например, компиляция фрагмента

«выражение $2x-1$ отличается от $2x-1.$ »

даёт следующий результат:

«выражение

$$2x - 1$$

отличается от $2x-1.$ »

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];

Например, компиляция фрагмента

«выражение $\left[2x-1\right]$ отличается от $2x-1$.»

даёт следующий результат:

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];

Например, компиляция фрагмента

«выражение `\[2x-1\]` отличается от `2x-1.`»

даёт следующий результат:

«выражение

$$2x - 1$$

отличается от `2x-1.`»

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];
- использование окружения `{equation}` для создания нумерованных «выключных формул».

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];
- использование окружения `{equation}` для создания нумерованных «выключных формул».

«выражение $\begin{equation}2x-1\end{equation}$ отличается от $2x-1.$ »

в результате компиляции даёт следующий результат:

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];
- использование окружения `{equation}` для создания нумерованных «выключных формул».

«выражение `\begin{equation}2x-1\end{equation}`
отличается от `2x-1.`»

в результате компиляции даёт следующий результат:

«выражение

$$2x - 1 \tag{1}$$

отличается от `2x-1.`»

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];
- использование окружения `{equation}` для создания нумерованных «выключных формул».

«выражение `\begin{equation}2x-1\end{equation}`
отличается от `2x-1.`»

в результате компиляции даёт следующий результат:

«выражение

$$2x - 1 \tag{1}$$

отличается от `2x-1.`»

IV. Набор математических формул

Переход в математическую моду и возвращение в текстовую моду осуществляется одним из следующих способов:

- с помощью одиночного знака \$;
- с помощью двойного \$\$ получаем «выключную формулу»;
- «выключную формулу» можно получить с помощью пары \[и \];
- использование окружения `{equation}` для создания нумерованных «выключных формул».

«выражение `\begin{equation}2x-1\end{equation}`
отличается от $2x-1.$ »

в результате компиляции даёт следующий результат:

«выражение

$$2x - 1 \tag{1}$$

отличается от $2x-1.$ » Обратите внимание на (1) справа.

Рассмотрим пример?

IV.1. Верхние и нижние индексы

Верхний и нижний индекс ставится *только в математической моде*.

Верхний индекс ставится с помощью символа $\wedge$. Например, код $2^{\{x\}}$ в результате компиляции даёт 2^x .

Нижний индекс ставится с помощью символа $_$. Например, код $2_{\{x\}}$ в результате компиляции даёт 2_x .

IV.2. Греческие буквы в формулах

α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>
δ	<code>\delta</code>	ϵ	<code>\epsilon</code>	ε	<code>\varepsilon</code>
ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>
ϑ	<code>\vartheta</code>	ι	<code>\iota</code>	κ	<code>\kappa</code>
λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>
ξ	<code>\xi</code>	π	<code>\pi</code>	ϖ	<code>\varpi</code>
ρ	<code>\rho</code>	ϱ	<code>\varrho</code>	σ	<code>\sigma</code>
ς	<code>\varsigma</code>	τ	<code>\tau</code>	υ	<code>\upsilon</code>
ϕ	<code>\phi</code>	φ	<code>\varphi</code>	χ	<code>\chi</code>
ψ	<code>\psi</code>	ω	<code>\omega</code>		

Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>
Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>	Π	<code>\Pi</code>
Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>
Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>		

Рассмотрим пример?

IV.3. Готические буквы в формулах

Буквы готического шрифта набираются в математической моде.

Переключение на готический шрифт происходит командой `\mathfrak`.

Например, в результате компиляции кода

`$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$` получаем
$$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$$

А в результате компиляции кода

`$\mathfrak{abcdefghijklmnopqrstuvwxyz}$` получаем
$$\mathfrak{abcdefghijklmnopqrstuvwxyz}$$

IV.3. Готические буквы в формулах

Буквы готического шрифта набираются в математической моде.

Переключение на готический шрифт происходит командой `\mathfrak`.

Соответствие букв латинского и готического алфавитов:

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>	<i>M</i>	<i>N</i>	<i>O</i>	<i>P</i>	<i>Q</i>	<i>R</i>	<i>S</i>	<i>T</i>	<i>U</i>	<i>V</i>	<i>W</i>	<i>X</i>	<i>Y</i>	<i>Z</i>
<i>Ɑ</i>	<i>Ɱ</i>	<i>Ɐ</i>	<i>Ɒ</i>	<i>ⱱ</i>	<i>Ⱳ</i>	<i>ⱳ</i>	<i>ⱴ</i>	<i>Ⱶ</i>	<i>ⱶ</i>	<i>ⱷ</i>	<i>ⱸ</i>	<i>ⱹ</i>	<i>ⱺ</i>	<i>ⱻ</i>	<i>ⱼ</i>	<i>ⱽ</i>	<i>Ȿ</i>	<i>Ɀ</i>	<i>Ⳑ</i>	<i>ⳑ</i>	<i>Ⳓ</i>	<i>ⳓ</i>	<i>Ⳕ</i>	<i>ⳕ</i>	<i>Ⳗ</i>
<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>	<i>h</i>	<i>i</i>	<i>j</i>	<i>k</i>	<i>l</i>	<i>m</i>	<i>n</i>	<i>o</i>	<i>p</i>	<i>q</i>	<i>r</i>	<i>s</i>	<i>t</i>	<i>u</i>	<i>v</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
<i>ⱦ</i>	<i>Ⱨ</i>	<i>ⱨ</i>	<i>Ⱪ</i>	<i>ⱪ</i>	<i>ⱬ</i>	<i>Ɑ</i>	<i>Ɱ</i>	<i>Ɐ</i>	<i>Ɒ</i>	<i>ⱱ</i>	<i>Ⱳ</i>	<i>ⱳ</i>	<i>ⱴ</i>	<i>Ⱶ</i>	<i>ⱶ</i>	<i>ⱷ</i>	<i>ⱸ</i>	<i>ⱹ</i>	<i>ⱺ</i>	<i>ⱻ</i>	<i>ⱼ</i>	<i>ⱽ</i>	<i>Ȿ</i>	<i>Ɀ</i>	<i>ⳗ</i>

IV.4. Некоторые элементарные функции

Естественно, корректно компилируются только при использовании в математической моде.

$\log_x y$	<code>\log_{x}{y}</code>	$\lg x$	<code>\lg{x}</code>	$\ln x$	<code>\ln{x}</code>
$\sin$	<code>\sin</code>	$\cos$	<code>\cos</code>	$\arcsin$	<code>\arcsin</code>
tg	<code>\operatorname{tg}</code>	ctg	<code>\operatorname{ctg}</code>	$\arccos$	<code>\arccos</code>
$\sqrt{}$	<code>\sqrt{\dots}</code>	arctg	<code>\operatorname{arctg}</code>	arcctg	<code>\operatorname{arcctg}</code>
a^b	<code>a^{b}</code>	$\sinh$	<code>\sinh</code>	$\cosh$	<code>\cosh</code>

и др.

IV.5. Бинарные операции

$+$	<code>+</code>	$-$	<code>-</code>	$*$	<code>*</code>
$\pm$	<code>\pm</code>	$\mp$	<code>\mp</code>	$\times$	<code>times</code>
$\div$	<code>\div</code>	$\setminus$	<code>\setminus</code>	$\cdot$	<code>\cdot</code>
$\circ$	<code>\circ</code>	$\bullet$	<code>\bullet</code>	$\cap$	<code>\cap</code>
$\cup$	<code>\cup</code>	$\oplus$	<code>\oplus</code>	$\sqcap$	<code>\sqcap</code>
$\sqcup$	<code>\sqcup</code>	$\vee$	<code>\vee</code>	$\wedge$	<code>\wedge</code>
$\oplus$	<code>\oplus</code>	$\ominus$	<code>\ominus</code>	$\otimes$	<code>\otimes</code>
$\odot$	<code>\odot</code>	$\oslash$	<code>\oslash</code>	$\triangleleft$	<code>\triangleleft</code>
$\triangleright$	<code>\triangleright</code>	$\amalg$	<code>\amalg</code>	$\diamond$	<code>\diamond</code>
$\wr$	<code>\wr</code>	$\star$	<code>\star</code>	$\dagger$	<code>\dagger</code>
$\ddagger$	<code>\ddagger</code>	$\bigcirc$	<code>\bigcirc</code>	$\triangle$	<code>\triangle</code>
$\bigtriangledown$	<code>\bigtriangledown</code>				

IV.6. Отношения

$\lt$	<code><</code>	$\gt$	<code>></code>	$=$	<code>=</code>
$\vdots$	<code>:</code>	$\sim$	<code>\sim</code>	$\simeq$	<code>\simeq</code>
$\approx$	<code>\approx</code>	$\cong$	<code>\cong</code>	$\equiv$	<code>\equiv</code>
$\ll$	<code>\ll</code>	$\gg$	<code>\gg</code>	$\doteq$	<code>\doteq</code>
$\leq$	<code>\le, \leq</code>	$\geq$	<code>\ge, \geq</code>	$\in$	<code>\in</code>
$\notin$	<code>\notin</code>	$\ni$	<code>\ni</code>	$\subset$	<code>\subset</code>
$\subseteq$	<code>\subseteq</code>	$\supset$	<code>\supset</code>	$\supseteq$	<code>\supseteq</code>
$\succ$	<code>\succ</code>	$\prec$	<code>\prec</code>	$\succeq$	<code>\succeq</code>
$\preceq$	<code>\preceq</code>	$\asymp$	<code>\asymp</code>	$\sqsubseteq$	<code>\sqsubseteq</code>
$\sqsupseteq$	<code>\sqsupseteq</code>	$\models$	<code>\models</code>	$\vdash$	<code>\vdash</code>
$\dashv$	<code>\dashv</code>	$\smile$	<code>\smile</code>	$\frown$	<code>\frown</code>
$\mid$	<code>\mid</code>	$\bowtie$	<code>\bowtie</code>	$\Join$	<code>\Join</code>
$\propto$	<code>\propto</code>	$\parallel$	<code>\parallel</code>	$\perp$	<code>\perp</code>

IV.7. Операции

log	\log	lg	\lg	ln	\ln
arg	\arg	ker	\ker	dim	\dim
hom	\hom	deg	\deg	exp	\exp
sin	\sin	arcsin	\arcsin	cos	\cos
arccos	\arccos	tan	\tan	arctan	\arctan
cot	\cot	sec	\sec	csc	\csc
sinh	\sinh	cosh	\cosh	tanh	\tanh
coth	\coth				

IV.8. Операции с пределами

Σ	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

Обратите внимание на разницу между знаками

$$\int_a^{\quad} \quad \int_a^b \quad \int_a^b \quad \int_a^b$$

↑ ↑ ↑ ↑
результат компиляции выражения `\int_{a}^{b}`

IV.8. Операции с пределами

Σ	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

Обратите внимание на разницу между знаками

$$\int_a^b \quad \int_a^b \quad \int_a^b \quad \int_a^b$$

↑
↑
↑
↑
результат компиляции выражения `\int_{a}^{b}`
результат компиляции выражения `\int\limits_{a}^{b}`

IV.8. Операции с пределами

$\sum$	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

Обратите внимание на разницу между знаками

$$\int_a^{} \quad \int_a^b \quad \int_a^b \quad \int_a^b$$

↑ результат компиляции выражения `\int_{a}^{b}`
 ↑ результат компиляции выражения `\int\limits_{a}^{b}`
 ↑ результат компиляции выражения `\displaystyle\int_{a}^{b}`

IV.8. Операции с пределами

Σ	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

Обратите внимание на разницу между знаками

$$\int_a^{\quad} \quad \int_a^b \quad \int_a^b \quad \int_a^b$$

результат компиляции выражения `\int_{a}^{b}`

результат компиляции выражения `\int\limits_{a}^{b}`

результат компиляции выражения `\displaystyle\int_{a}^{b}`

результат компиляции выражения `\displaystyle\int\limits_{a}^{b}`

IV.8. Операции с пределами

Σ	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

$\int_a^b$	<code>\int_{a}^{b}</code>	$\int_a^b$	<code>\int_{a}^{b}</code>
$\int_a^b$	<code>\displaystyle\int_{a}^{b}</code>	$\int_a^b$	<code>\displaystyle\int_{a}^{b}</code>

В «выключных» формулах (переход в которые и выход из которых осуществляется с помощью знака `$$`) режим `\displaystyle` выставлен «по умолчанию».

IV.8. Операции с пределами

$\sum$	<code>\sum</code>	$\prod$	<code>\prod</code>	$\bigcup$	<code>\bigcup</code>
$\bigcap$	<code>\bigcap</code>	$\coprod$	<code>\coprod</code>	$\bigoplus$	<code>\bigoplus</code>
$\bigotimes$	<code>\bigotimes</code>	$\bigodot$	<code>\bigodot</code>	$\bigvee$	<code>\bigvee</code>
$\bigwedge$	<code>\bigwedge</code>	$\biguplus$	<code>\biguplus</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\lim$	<code>\lim</code>	$\limsup$	<code>\limsup</code>	$\liminf$	<code>\liminf</code>
$\max$	<code>\max</code>	$\min$	<code>\min</code>	$\sup$	<code>\sup</code>
$\inf$	<code>\inf</code>	$\det$	<code>\det</code>	$\Pr$	<code>\Pr</code>
$\gcd$	<code>\gcd</code>	$\int$	<code>\int</code>		

$\int_a^b$	<code>\int_{a}^{b}</code>	$\int_a^b$	<code>\int\limits_{a}^{b}</code>
		$\int_a^b$	<code>\displaystyle\int_{a}^{b}</code>

$\int f(x)g(x) dx$ `\int f(x)g(x)\,dx` — здесь выражение «`\,`» обеспечивает вставку «тонкого пробельчика» (шпации), отделяющего dx от $f(x)g(x)$.

IV.9. Одно над другим. Деление. Растягивающиеся скобки

$$x_{\alpha}^{t+1}$$

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}

`x_{\alpha}^{t+1}`

Индексы можно поставить *только в математической моде* (внутри выделения `$...$` или `$$...$$`)

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_α^{t+1}

$x_{\backslash\alpha}^{t+1}$

$A \overset{f}{\rightarrow} B,$

$P \overset{x}{y} Q$

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	$x_{\backslash\alpha}^{\wedge\{t+1\}}$
$A\overset{f}{\rightarrow}B, P\overset{x}{y}Q$	$A\stackrel{f}{\rightarrow}B, P\stackrel{x}{y}Q$

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	$x_{\backslash\alpha}^{\wedge\{t+1\}}$
$A\overset{f}{\rightarrow}B, P\overset{x}{y}Q$	$A\stackrel{f}{\rightarrow}B, P\stackrel{x}{y}Q$
$x+\frac{a}{b}$	

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b}$	<code>x+\frac{a}{b}</code>

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}</code>

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\to} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\to} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	<code>\underbrace{x+y}_{A}+z+\overbrace{p+q}^{B}</code>

Рассмотрим пример?

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	<code>\underbrace{x+y}_{A}+z+\overbrace{p+q}^{B}</code>
$\{\frac{x}{y}\}$	<code>\{\displaystyle\frac{x}{y}\}</code>
Φу, как некрасиво...	

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	<code>\underbrace{x+y}_{A}+z+\overbrace{p+q}^{B}</code>
$\left\{\frac{x}{y}\right\}$	<code>\left\{\displaystyle\frac{x}{y}\right\}</code>
Другое дело!	

IV.9. Одно над другим. Деление. Растягивающиеся скобки

$$x_{\alpha}^{t+1}$$

$$x_{\backslash\alpha}^{\wedge\{t+1\}}$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$A \stackrel{\text{f}}{\rightarrow} B, P \stackrel{\text{x}}{\text{y}} Q$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}$$

$$\underbrace{x+y}_A + z + \overbrace{p+q}^B$$

$$\underbrace{\{x+y\}}_{\{A\}}+z+\overbrace{\{p+q\}}^{\{B\}}$$

$$\left\{ \frac{x}{y} \right\}$$

$$\backslash\left\{\displaystyle\frac{x}{y}\right\}\backslash$$

Другое дело! Каждая открывающаяся «растягивающаяся» скобка $\backslash\left\{$, $\backslash\left[$ и т.д. должна закрываться тоже «растягивающейся» скобкой $\right\}\backslash$, $\right]\backslash$ и др., хотя закрываться может «чужой» скобкой, например, $[...]$.

IV.9. Одно над другим. Деление. Растягивающиеся скобки

$$x_{\alpha}^{t+1}$$

$$x_{\alpha}^{t+1}$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\left\{ \frac{x}{y} \right\}$$

$$\left\{ \frac{x}{y} \right\}$$

Если одна из «растягивающихся» скобок не нужна, ее заменяют на «`\left.`» или «`\right.`».

IV.9. Одно над другим. Деление. Растягивающиеся скобки

$$x_{\alpha}^{t+1}$$

$$x_{\alpha}^{t+1}$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\left. \frac{x}{y} \right\}$$

$$\left. \frac{x}{y} \right\}$$

Если одна из «растягивающихся» скобок не нужна, ее заменяют на «\left.» или «\right.».

IV.9. Одно над другим. Деление. Растягивающиеся скобки

$$x_{\alpha}^{t+1}$$

$$x_{\alpha}^{t+1}$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$x + \frac{a}{b} \quad x + \frac{a}{b}$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\underbrace{x + y}_A + z + \overbrace{p + q}^B$$

$$\left[\frac{x}{y} \right]$$

$$\left[\frac{x}{y} \right].$$

Если одна из «растягивающихся» скобок не нужна, ее заменяют на «`\left.`» или «`\right.`».

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	<code>\underbrace{x+y}_{A}+z+\overbrace{p+q}^{B}</code>
$\left[\frac{x}{y}\right]$	<code>\left[\displaystyle\frac{x}{y}\right].</code>

В текстовой моде можно применить команду `\shortstack{первая строка\кратко{вторая строка}}`, при компиляции дающую следующий результат:

	первая строка
второй результат:	вторая строка,

IV.9. Одно над другим. Деление. Растягивающиеся скобки

x_{α}^{t+1}	<code>x_{\alpha}^{t+1}</code>
$A \overset{f}{\rightarrow} B, P \overset{x}{y} Q$	<code>A \stackrel{f}{\rightarrow} B, P \stackrel{x}{y} Q</code>
$x + \frac{a}{b} \quad x + \frac{a}{b}$	<code>x+\frac{a}{b}\quad x+\displaystyle\frac{a}{b}</code>
$\underbrace{x+y}_A + z + \overbrace{p+q}^B$	<code>\underbrace{x+y}_{A}+z+\overbrace{p+q}^{B}</code>
$\left[\frac{x}{y}\right]$	<code>\left[\displaystyle\frac{x}{y}\right].</code>

В текстовой моде можно применить команду `\shortstack{первая строка\secondrow{вторая строка}}`, при компиляции дающую следующий результат:

	первая строка	вторая строка	
первая строка	вторая строка	третья строка	и т.д.

IV.10. Стрелки

$\Rightarrow$	<code>\Longrightarrow</code>	$\nwarrow$	<code>\nwarrow</code>	$\nearrow$	<code>\nearrow</code>
$\Leftarrow$	<code>\Longleftarrow</code>	$\uparrow$	<code>\uparrow</code>	$\downarrow$	<code>\downarrow</code>
$\longleftarrow$	<code>\longleftarrow</code>	$\updownarrow$	<code>\updownarrow</code>	$\leftarrow$	<code>\gets</code>
$\longrightarrow$	<code>\longrightarrow</code>	$\searrow$	<code>\searrow</code>	$\swarrow$	<code>\swarrow</code>
$\longleftrightarrow$	<code>\longleftrightarrow</code>	$\leftrightarrow$	<code>\leftrightarrow</code>	$\rightarrow$	<code>\to</code>
$\Longleftrightarrow$	<code>\Longleftrightarrow</code>	$\mapsto$	<code>\mapsto</code>	$\longmapsto$	<code>\longmapsto</code>
$\leftharpoonup$	<code>\leftharpoonup</code>	$\Uparrow$	<code>\Uparrow</code>	$\Downarrow$	<code>\Downarrow</code>
$\rightharpoonup$	<code>\rightharpoonup</code>	$\Updownarrow$	<code>\Updownarrow</code>	$\Rightarrow$	<code>\Rightarrow</code>
$\leftharpoonup$	<code>\leftharpoonup</code>	$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Leftarrow$	<code>\Leftarrow</code>
$\rightharpoonup$	<code>\rightharpoonup</code>	$\hookleftarrow$	<code>\hookleftarrow</code>		
$\rightharpoonup$	<code>\rightharpoonup</code>	$\hookrightarrow$	<code>\hookrightarrow</code>	$\leadsto$	<code>\leadsto</code>

IV.11. Разные символы

∂	<code>\partial</code>	$\triangle$	<code>\triangle</code>	$\angle$	<code>\angle</code>
∞	<code>\infty</code>	$\forall$	<code>\forall</code>	$\exists$	<code>\exists</code>
$\emptyset$	<code>\emptyset</code>	$\neg$	<code>\neg</code>	$\aleph$	<code>\aleph</code>
$'$	<code>\prime</code>	$\hbar$	<code>\hbar</code>	∇	<code>\nabla</code>
$\imath$	<code>\imath</code>	$\jmath$	<code>\jmath</code>	ℓ	<code>\ell</code>
$\sqrt{}$	<code>\surd</code>	$\flat$	<code>\flat</code>	$\sharp$	<code>\sharp</code>
$\natural$	<code>\natural</code>	$\top$	<code>\top</code>	$\bot$	<code>\bot</code>
$\wp$	<code>\wp</code>	$\Re$	<code>\Re</code>	$\Im$	<code>\Im</code>
<code>\</code>	<code>\backslash</code>	$\parallel$	<code>\parallel</code>	$\spadesuit$	<code>\spadesuit</code>
$\clubsuit$	<code>\clubsuit</code>	$\diamondsuit$	<code>\diamondsuit</code>	$\heartsuit$	<code>\heartsuit</code>
$\mho$	<code>\mho</code>	$\Box$	<code>\Box</code>	$\Diamond$	<code>\Diamond</code>
$\dagger$	<code>\dag</code>	$\S$	<code>\S</code>	$\copyright$	<code>\copyright</code>
$\ddagger$	<code>\ddag</code>	$\P$	<code>\P</code>	$\pounds$	<code>\pounds</code>

IV.12. Некоторые сокращения

*	*	\ast	≠	\ne	\neq
≠	\ne	\neq			
≤	\le	\leq	≥	\ge	\geq
[	[	\lbrack	]	]	\rbrack
{	\{	\lbrace	{	\{	\rbrace
→	\to	\rightarrow	←	\gets	\leftarrow
∃	\ni	\owns		\Vert	\
∧	\wedge	\land	∨	\vee	\lor
¬	\neg	\lnot			

V. Прямоугольные фрагменты (боксы)

L^AT_EX при формировании текста оперирует, в основном, с прямоугольными объектами. Например, букву он воспринимает как прямоугольник.

Можно из объектов самостоятельно формировать прямоугольники, называемые боксами, и оперировать с ними. Для этого применяются команды

`\mbox`, `\fbox`, `\makebox`, `\parbox`, `\raisebox`, `\hbox`,

описанные ниже.

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для превращения нежелательного переноса на другую строку.

Синтаксис команды: `\mbox{фрагмент_строки}`

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку. В результате компиляции кода:

Вот пример строки, содержащей очень длинную формулу:

$2x^5+3x^4-37x^3+6x^2-51x>18.$ получим:

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку. В результате компиляции кода:

Вот пример строки, содержащей очень длинную формулу:

`$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$` получим:

Вот пример строки, содержащей очень длинную формулу: $2x^5 + 3x^4 - 37x^3 + 6x^2 - 51x > 18.$

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку. В результате компиляции кода:

Вот пример строки, содержащей очень длинную формулу:

`$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$` получим:

Вот пример строки, содержащей очень длинную формулу: $2x^5 + 3x^4 - 37x^3 + 6x^2 - 51x > 18.$

Перенос произведён по правилам английского языка, и результат отличается от привычного.

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку. В результате компиляции кода:

Вот пример строки, содержащей очень длинную формулу:

`$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$` получим:

Вот пример строки, содержащей очень длинную формулу: $2x^5 + 3x^4 - 37x^3 + 6x^2 - 51x > 18.$

Отредактируем код:

Вот пример строки, содержащей очень длинную формулу:

`\mbox{$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$}`

и после компиляции получим:

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку. В результате компиляции кода:

Вот пример строки, содержащей очень длинную формулу:
`$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$` получим:

Вот пример строки, содержащей очень длинную формулу: $2x^5 + 3x^4 - 37x^3 + 6x^2 - 51x > 18.$

Отредактируем код:

Вот пример строки, содержащей очень длинную формулу:
`\mbox{$2x^{\{5\}}+3x^{\{4\}}-37x^{\{3\}}+6x^{\{2\}}-51x>18.$}`

и после компиляции получим:

Вот пример строки, содержащей очень длинную формулу:
 $2x^5 + 3x^4 - 37x^3 + 6x^2 - 51x > 18.$

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для предотвращения нежелательного переноса на другую строку.

Предотвращать переносы иногда приходится при использовании слов с дефисами: «когда-нибудь», «формула Ньютона-Лейбница».

V.1. Объединение элементов строки в бокс: `\mbox`

Объединение элементов строки в один бокс бывает необходимо, например, для превращения нежелательного переноса на другую строку.

Иногда применение `\mbox` приводит к переполнению строки, и её часть «вылезает» за пределы строки. тогда приходится перестраивать предложение или использовать `\newline`.

V.2. Выделение элементов строки в рамку: `\fbox`

Синтаксис команды: `\fbox{содержимое_бокса}`.

Например, компиляция фрагмента кода

Можно `\fbox{заклучить в рамку}` часть строки.

приведёт к такому результату:

Можно заклучить в рамку часть строки.

V.2. Выделение элементов строки в рамку: `\fbox`

Синтаксис команды: `\fbox{содержимое_бокса}`.

Например, компиляция фрагмента кода

Можно `\fbox{заклучить в рамку}` часть строки.

приведёт к такому результату:

Можно заклучить в рамку часть строки.

При этом на проведение границы необходимо дополнительное пространство. Это видно, если заменить `\fbox` на `\mbox`.

V.2. Выделение элементов строки в рамку: `\fbox`

Синтаксис команды: `\fbox{содержимое_бокса}`.

Например, компиляция фрагмента кода

Можно `\mbox{заклучить в рамку}` часть строки.

приведёт к такому результату:

Можно заключить в рамку часть строки.

При этом на проведение границы необходимо дополнительное пространство. Это видно, если заменить `\fbox` на `\mbox`.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L, H)[pq]{содержимое_бокса}`, где L и H — числа, означающие длину и высоту прямоугольника, отводимого в тексте, чтобы разместить «содержимое_бокса»; $p \in \{l, c, r\}$, $q \in \{b, c, t\}$ — буквы, определяющие, как «содержимое_бокса» будет расположено в прямоугольнике длиной L и шириной H ,

$$\left\{ \begin{array}{l} l — \text{«left»}, \text{«влево»}, \\ c — \text{«center»}, \text{«по центру»}, \\ r — \text{«right»}, \text{«вправо»}, \end{array} \right. \quad \left\{ \begin{array}{l} t — \text{«top»}, \text{«сверху»}, \\ c — \text{«center»}, \text{«по центру»}, \\ b — \text{«bottom»}, \text{«снизу»}. \end{array} \right.$$

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L,H)[pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(25,4)[lt]{часть выделена}` в бокс.}

в строке

в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L, H) [pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(25,4)[ct]{часть выделена}` в бокс.
часть выделена

в строке

в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L, H) [pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(25,4)[rt]{часть выделена}` в бокс.
часть выделена

в строке

в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L,H)[pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(25,4)[rc]{часть выделена}` в бокс.}

в строке часть выделена в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L,H)[pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(25,4)[rb]{часть выделена}` в бокс.}

в строке часть выделена в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L,H)[pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(15,4)[rt]{часть выделена}` в бокс.
часть выделена

в строке

в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L, H) [pq] {содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(10,4)[rt]{часть выделена}` в бокс.}

часть выделена

в строке

в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox( $L$ ,  $H$ ) [ $pq$ ]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=2mm` в строке

`\makebox(20,1)[rt]{часть выделена}` в бокс.}

в строке часть выделена в бокс.

V.3. Бокс фиксированного размера: `\makebox`

При использовании этой команды обычно предполагается, что зафиксирована некоторая единица измерения, например, следующим образом: `\unitlength=1mm`

Синтаксис команды: `\makebox(L,H)[pq]{содержимое_бокса}`.

Например, компиляция кода

`{\unitlength=1mm` в строке

`\makebox(20,4)[rt]{часть выделена}` в бокс.}

часть выделена в бокс.
в строке

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

Например, компиляция кода

```
«Часть \parbox{10em}{текста сформирована в абзац.\par  
Абзацев может быть несколько.} Здесь это не обосновано...»
```

приводит к следующему результату:

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

Например, компиляция кода

```
«Часть \parbox{10em}{текста сформирована в абзац.\rag  
Абзацев может быть несколько.} Здесь это не обосновано...»  
приводит к следующему результату:
```

	текста сформирована	
Часть	в абзац.	Здесь это не обосновано...
Абзацев	может быть	
несколько.		

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

Например, компиляция кода

```
«Часть \parbox{12em}{текста сформирована в абзац.\rag  
Абзацев может быть несколько.} Здесь это не обосновано...»  
приводит к следующему результату:
```

текста сформирована в аб-		
зац.		
Часть	Здесь это не обосновано...	
Абзацев	может	быть
несколько.		

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

Например, компиляция кода

```
«Часть \parbox{50mm}{текста сформирована в абзац.\par  
Абзацев может быть несколько.} Здесь это не обосновано...»  
приводит к следующему результату:
```

текста	сформиро-	
Часть	вана в абзац.	Здесь это не обосновано...
Абзацев	может	
быть	несколько.	

V.4. Бокс из абзацев: `\parbox`

Синтаксис команды:

```
\parbox{ширина}{содержимое_бокса}
```

Например, компиляция кода

```
«Часть \parbox{50mm}{текста сформирована в абзац.\par  
Абзацев может быть несколько.} Здесь это не обосновано...»  
приводит к следующему результату:
```

текста	сформиро-	
Часть	вана в абзац.	Здесь это не обосновано...
Абзацев	может	
быть несколько.		

В тексте «содержимое_бокса» в абзацах подавлен абзацный отступ.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки, обозначенных как «содержимое_бокса» и поднять его (если аргумент «высота» имеет положительное значение) или опустить (если аргумент «высота» отрицательный) на именованую величину «высота», причём указание **единицы измерения** обязательно.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки.

Например, компиляция кода
«В строке `\raisebox{1ex}{вот эта часть}`
поднята или опущена.»
приводит к такому результату:

В строке ^{вот эта часть} поднята или опущена.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки.

Например, компиляция кода
«В строке `\raisebox{2ex}{вот эта часть}`
поднята или опущена.»
приводит к такому результату:
вот эта часть
В строке поднята или опущена.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки.

Например, компиляция кода
«В строке `\raisebox{-1ex}{вот эта часть}`
поднята или опущена.»
приводит к такому результату:

В строке `вот эта часть` поднята или опущена.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки.

Например, компиляция кода
«В строке `\raisebox{3mm}{вот эта часть}`
поднята или опущена.»
приводит к такому результату:

В строке ^{вот эта часть} поднята или опущена.

V.5. Смещение бокса вверх или вниз `\raisebox`

Команда `\raisebox{высота}{содержимое_бокса}` позволяет сформировать бокс из элементов строки.

Например, компиляция кода
«В строке `\raisebox{-1.5cm}{вот эта часть}`
поднята или опущена.»
приводит к такому результату:

В строке поднята или опущена.

ВОТ ЭТА ЧАСТЬ

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст может занимать весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 20em {этот текст \hss}}` может занимать весь бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 20em {этот текст \hss}}` занимает весь бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст, хотя и длинный, занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 20em {этот текст, хотя и длинный \hss занимает весь}}` бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст, хотя и очень длинный занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 20em {этот текст, хотя и очень длинный \hss}}` занимает весь бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 100mm {этот текст \hss}}` занимает весь бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 100mm {этот
текст занимает \hss весь}}` бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 100mm {этот текст \hss занимает \hss весь}}` бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

Команд `\hss` может быть несколько.

V.6. Горизонтальное смещение элементов внутри бокса: `\hbox`

Иногда возникает необходимость не просто зафиксировать размеры бокса, как в команде `\hbox`, но и требуется, чтобы элементы строки занимали всё отведённое место.

Например, этот текст занимает весь бокс.

Последнее предложение есть результат компиляции кода:

Например, `\underline{\hbox to 100mm {этот
\hss текст \hss занимает \hss весь}}` бокс.

Здесь `\hss` — это «пружинка», на которой происходит «растяжение» текста.

Команд `\hss` может быть несколько.

V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:
`\begin{rotate}{\alpha} ... \end{rotate}`.

Например, в результате компиляции кода

```
\rotatebox{60}{Наклон} \rotatebox{45}{Наклон}  
\rotatebox{30}{Наклон} \rotatebox{-30}{Наклон}  
\rotatebox{-45}{Наклон} \rotatebox{-60}{Наклон}
```

получим

V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:
`\begin{rotate}{\alpha} ... \end{rotate}`.

Например, в результате компиляции кода

```
\rotatebox{60}{Наклон} \rotatebox{45}{Наклон}  
\rotatebox{30}{Наклон} \rotatebox{-30}{Наклон}  
\rotatebox{-45}{Наклон} \rotatebox{-60}{Наклон}
```

получим

Наклон Наклон Наклон Наклон Наклон Наклон

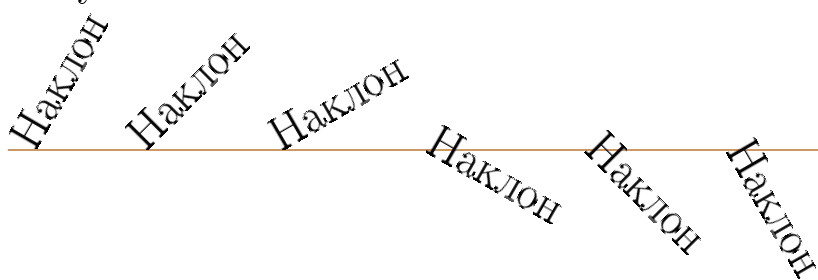
V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:
`\begin{rotate}{\alpha} ... \end{rotate}`.

Например, в результате компиляции кода

```
\rotatebox{60}{Наклон} \rotatebox{45}{Наклон}  
\rotatebox{30}{Наклон} \rotatebox{-30}{Наклон}  
\rotatebox{-45}{Наклон} \rotatebox{-60}{Наклон}
```

получим



Наклон

Наклон

Наклон

Наклон

Наклон

Наклон

V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:

```
\begin{rotate}{\alpha} ... \end{rotate}.
```

В результате компиляции кода

```
\begin{rotate}{60}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}

\begin{rotate}{-45}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}
```

получим

V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:

```
\begin{rotate}{\alpha} ... \end{rotate}.
```

В результате компиляции кода

```
\begin{rotate}{60}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}

\begin{rotate}{-45}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}
```

Наклон
получим

Наклон
текста

V.7. Вращение бокса

Поворот содержимого бокса на угол α° осуществляется командой `\rotatebox{\alpha}{...}` или с помощью **окружения** `rotate`:

```
\begin{rotate}{\alpha} ... \end{rotate}.
```

В результате компиляции кода

```
\begin{rotate}{60}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}

\begin{rotate}{-45}
  \parbox{5em}{Наклон\newline текста}
\end{rotate}
```

Наклон
текста
получим

Окружение `rotate` не умеет правильно вычислять размер бокса!

VI. Таблицы, системы уравнений и т.п.

Мы рассмотрим только простейшие случаи. Более полное изложение можно найти, например, в [книге Львовского](#).

VI.1. Таблицы в текстовой моде

В текстовой моде для создания таблиц можно использовать **окру-**
жение `tabular`.

VI.1. Таблицы в текстовой моде

Формат команды:

```
\begin{tabular}{параметры}  
...\\  
\end{tabular}
```

Параметры представляют собой строку, элементами которой могут быть: а) буквы l, c, r, б) слово p{...}, в) символ |

Каждому из этих элементов соответствует столбец таблицы, в частности, количество элементов вида а)-в) равно числу столбцов таблицы.

VI.1. Таблицы в текстовой моде

Формат команды:

```
\begin{tabular}{параметры}
```

```
...\\
```

```
\end{tabular}
```

Параметр	Действие
l	выравнивать элементы по левому краю столбца;
c	выравнивать элементы по центру столбца;
r	выравнивать элементы по правому краю столбца;
p{ <i>длина</i> }	представить элемент столбца в виде абзаца шириной « <i>длина</i> », например, p{3.2cm}, p{27mm}, p{3em};
	отделить от следующего столбца вертикальной чертой.

Каждая строка таблицы завершается командой \\.

VI.1. Таблицы в текстовой моде

Формат команды:

```
\begin{tabular}{параметры}
```

```
...\\
```

```
\end{tabular}
```

Параметр	Действие
l	выравнивать элементы по левому краю столбца;
c	выравнивать элементы по центру столбца;
r	выравнивать элементы по правому краю столбца;
p{ <i>длина</i> }	представить элемент столбца в виде абзаца шириной « <i>длина</i> », например, p{3.2cm}, p{27mm}, p{3em};
	отделить от следующего столбца вертикальной чертой.

Каждая строка таблицы завершается командой `\\`.

Команда `\hline` применяется для проведения горизонтальной линии по всей ширине таблицы.

VI.1. Таблицы в текстовой моде

Формат команды:

```
\begin{tabular}{параметры}
```

```
...\\
```

```
\end{tabular}
```

Параметр	Действие
l	выравнивать элементы по левому краю столбца;
c	выравнивать элементы по центру столбца;
r	выравнивать элементы по правому краю столбца;
p{длина}	представить элемент столбца в виде абзаца шириной «длина», например, p{3.2cm}, p{27mm}, p{3em};
	отделить от следующего столбца вертикальной чертой.

Каждая строка таблицы завершается командой `\\`.

Команда `\cline{a-b}` применяется для проведения горизонтальной линии через столбцы с номерами $a, a + 1, \dots, b$.

VI.1. Таблицы в текстовой моде

Формат команды:

```
\begin{tabular}{параметры}
```

```
...\\
```

```
\end{tabular}
```

Параметр	Действие
l	выравнивать элементы по левому краю столбца;
c	выравнивать элементы по центру столбца;
r	выравнивать элементы по правому краю столбца;
p{ <i>длина</i> }	представить элемент столбца в виде абзаца шириной « <i>длина</i> », например, p{3.2cm}, p{27mm}, p{3em};
	отделить от следующего столбца вертикальной чертой.

В строке элементы столбцов отделяются друг от друга символом &.

Рассмотрим пример?

VI.2. Абзац в ячейке таблицы

```
\begin{tabular}{|p{30mm}|c|c|}
```

```
\hline
```

Способ задания & объект & комментарий\\

```
\hline
```

```
\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

В данной таблице элементы первого столбца форматируются в абзац шириной 30 миллиметров.

VI.2. Абзац в ячейке таблицы

```

\begin{tabular}{|p{30mm}|c|c|}
\hline
Способ задания & объект & комментарий\\
\hline
\hline
Система уравнений & линия & в пространстве\\
\hline
Система неравенств & область & на плоскости\\
\hline
\end{tabular}

```

Способ за- дания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

В данной таблице элементы первого столбца форматируются в абзац шириной 30 миллиметров.

VI.2. Абзац в ячейке таблицы

```
\begin{tabular}{|p{30mm}|c|c|}
```

```
\hline
```

Способ задания & объект & комментарий\\

```
\hline
```

```
\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Способ за- дания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Ширину абзаца можно задавать в относительных единицах: **em** (это ширина буквы m), **ex** (это высота буквы x), и долях ширины текста на странице, т.е. долях от `\textwidth`.

VI.2. Абзац в ячейке таблицы

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ задания & объект & комментарий\\

```
\hline
```

```
\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Способ за- дания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Ширину абзаца можно задавать в относительных единицах: **em** (это ширина буквы m), **ex** (это высота буквы x), и долях ширины текста на странице, т.е. долях от `\textwidth`.

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Знак \\ окончания строки
можно поставить даже после
первого элемента строки.

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Знак `\\` окончания строки можно поставить даже после первого элемента строки.

Но при этом прекращается действие команды изображения вертикальных границ (если они были предусмотрены, например, с помощью `|` в **строке параметров**).

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Способ задания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Знак `\\` окончания строки можно поставить даже после первого элемента строки.

Но при этом прекращается действие команды изображения вертикальных границ (если они были предусмотрены, например, с помощью `|` в **строке параметров**).

VI.3. Неполные строки в таблице

```

\begin{tabular}{|p{0.16\textwidth}|c|c|}
\hline
Способ & объект & комментарий\\
задания\\
\hline\hline
Система уравнений & линия & в пространстве\\
\hline
Система неравенств & область & на плоскости\\
\hline
\end{tabular}

```

Способ задания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Чтобы избежать исчезновения границ столбцов достаточно поставить в «укороченной» строке знаки & границы столбцов (с отсутствующим элементом столбца).

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Чтобы избежать исчезновения границ столбцов достаточно поставить в «укороченной» строке знаки & границы столбцов (с отсутствующим элементом столбца).

Попробуем.

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания & &\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Чтобы избежать исчезновения границ столбцов достаточно поставить в «укороченной» строке знаки & границы столбцов (с отсутствующим элементом столбца).

Попробуем.

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания & &\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Способ задания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Чтобы избежать исчезновения границ столбцов достаточно поставить в «укороченной» строке знаки & границы столбцов (с отсутствующим элементом столбца).

Попробуем.

VI.3. Неполные строки в таблице

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания & &\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Способ задания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Кстати, не было необходимости представлять в коде $\text{\LaTeX}$ первую строку таблицы в виде двух строк.

VI.4. Объединение ячеек

```

\begin{tabular}{|p{0.16\textwidth}|c|c|}
\hline
Способ & объект & комментарий\\
задания & & \\
\hline\hline
Система уравнений & линия & в пространстве\\
\hline
Система неравенств & область & на плоскости\\
\hline
\end{tabular}

```

Способ задания	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Допустим, необходимо, чтобы в первой строке вместо слов «объект» и «комментарий» было напечатано одно слово «прообраз», занимающее второй и третий столбцы.

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ & объект & комментарий\\

задания & &\\

```
\hline\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Допустим, необходимо, чтобы в первой строке вместо слов «объект» и «комментарий» было напечатано одно слово «прообраз», занимающее второй и третий столбцы.

Для этого можно примерить команду `\multicolumn`.

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

Способ задания & \multicolumn{2}{|c|}{прообраз}\\

```
\hline
```

```
\hline
```

Система уравнений & линия & в пространстве\\

```
\hline
```

Система неравенств & область & на плоскости\\

```
\hline
```

```
\end{tabular}
```

Допустим, необходимо, чтобы в первой строке вместо слов «объект» и «комментарий» было напечатано одно слово «прообраз», занимающее второй и третий столбцы.

Для этого можно примерить команду `\multicolumn`.

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}  
  \hline  
  Способ задания & \multicolumn{2}{|c|}{прообраз}\\  
  \hline  
  \hline  
  Система уравнений & линия & в пространстве\\  
  \hline  
  Система неравенств & область & на плоскости\\  
  \hline  
\end{tabular}
```

Способ за- дания	прообраз	
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Допустим, необходимо, чтобы в первой строке вместо слов «объект» и «комментарий» было напечатано одно слово «прообраз», занимающее второй и третий столбцы. Для этого можно примерить команду `\multicolumn`.

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}  
  \hline  
  Способ задания & \multicolumn{2}{|c|}{прообраз}\\  
  \hline  
  \hline  
  Система уравнений & линия & в пространстве\\  
  \hline  
  Система неравенств & область & на плоскости\\  
  \hline  
\end{tabular}
```

Способ за- дания	прообраз	
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Допустим, необходимо, чтобы под словом «прообраз» во втором и третьем столбцах были напечатаны слова «объект» и «комментарий».

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

```
Способ задания & \multicolumn{2}{|c|}{прообраз}\\
```

```
\hline
```

```
\hline
```

```
Система уравнений & линия & в пространстве\\
```

```
\hline
```

```
Система неравенств & область & на плоскости\\
```

```
\hline
```

```
\end{tabular}
```

Допустим, необходимо, чтобы под словом «прообраз» во втором и третьем столбцах были напечатаны слова «объект» и «комментарий».

Попробуем.

VI.4. Объединение ячеек

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}  
  \hline  
  Способ & \multicolumn{2}{|c|}{прообраз}\\  
  задания & объект & комментарий\\  
  \hline\hline  
  Система уравнений & линия & в пространстве\\  
  \hline  
  Система неравенств & область & на плоскости\\  
  \hline  
\end{tabular}
```

Допустим, необходимо, чтобы под словом «прообраз» во втором и третьем столбцах были напечатаны слова «объект» и «комментарий».

Попробуем.

VI.4. Объединение ячеек

```

\begin{tabular}{|p{0.16\textwidth}|c|c|}
\hline
Способ & \multicolumn{2}{|c|}{прообраз}\\
задания & объект & комментарий\\
\hline\hline
Система уравнений & линия & в пространстве\\
\hline
Система неравенств & область & на плоскости\\
\hline
\end{tabular}

```

Способ задания	прообраз	
	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Допустим, необходимо, чтобы под словом «прообраз» во втором и третьем столбцах были напечатаны слова «объект» и «комментарий».

Попробуем.

VI.5. Неполная горизонтальная граница

```

\begin{tabular}{|p{0.16\textwidth}|c|c|}

```

```

\hline
Способ & \multicolumn{2}{|c|}{прообраз}\\
задания & объект & комментарий\\

```

```

\hline\hline
Система уравнений & линия & в пространстве\\
\hline

```

```

Система неравенств & область & на плоскости\\
\hline
\end{tabular}

```

Способ задания	прообраз	
	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Надо бы слово «прообраз»
отделить от слов «объект»
и «комментарий» отрезком
линии.

VI.5. Неполная горизонтальная граница

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

```
Способ & \multicolumn{2}{|c|}{прообраз}\\
```

```
задания & объект & комментарий\\
```

```
\hline\hline
```

```
Система уравнений & линия & в пространстве\\
```

```
\hline
```

```
Система неравенств & область & на плоскости\\
```

```
\hline
```

```
\end{tabular}
```

Надо бы слово «прообраз» отделить от слов «объект» и «комментарий» отрезком линии.

Применим команду

```
\cline{m-n},
```

где m и n — начальный и конечный номера столбцов, через которые проходит отрезок прямой.

VI.5. Неполная горизонтальная граница

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

```
Способ & \multicolumn{2}{|c|}{прообраз}\\
```

```
\cline{2-3}
```

```
задания & объект & комментарий\\ \hline\hline
```

```
Система уравнений & линия & в пространстве\\
```

```
\hline
```

```
Система неравенств & область & на плоскости\\
```

```
\hline
```

```
\end{tabular}
```

Надо бы слово «прообраз» отделить от слов «объект» и «комментарий» отрезком линии.

Применим команду

```
\cline{ $m-n$ },
```

где m и n — начальный и конечный номера столбцов, через которые проходит отрезок прямой.

VI.5. Неполная горизонтальная граница

```
\begin{tabular}{|p{0.16\textwidth}|c|c|}
```

```
\hline
```

```
Способ & \multicolumn{2}{|c|}{прообраз}\\
```

```
\cline{2-3}
```

```
задания & объект & комментарий\\ \hline\hline
```

```
Система уравнений & линия & в пространстве\\
```

```
\hline
```

```
Система неравенств & область & на плоскости\\
```

```
\hline
```

```
\end{tabular}
```

Способ задания	прообраз	
	объект	комментарий
Система уравнений	линия	в пространстве
Система неравенств	область	на плоскости

Надо бы слово «прообраз» отделить от слов «объект» и «комментарий» отрезком линии.

Применим команду

```
\cline{m-n},
```

где m и n — начальный и конечный номера столбцов, через которые проходит отрезок прямой.

VI.6. Таблицы в математической моде

Формат команды:

```
\begin{array}{параметры}
```

```
...\\
```

```
\end{array}
```

Параметр	Действие
l	выравнивать элементы по левому краю столбца
c	выравнивать элементы по центру столбца
r	выравнивать элементы по правому краю столбца
	отделить от следующего столбца вертикальной чертой

Синтаксис команд `array` и `tabular` почти полностью аналогичен.

VI.7. Матрицы и системы отношений

$$\begin{pmatrix} a & b \\ c & d \\ e & f \end{pmatrix}$$

Данная матрица получается, например, компиляцией следующего фрагмента.

VI.7. Матрицы и системы отношений

$$\begin{pmatrix} a & b \\ c & d \\ e & f \end{pmatrix}$$

Данная матрица получается, например, компиляцией следующего фрагмента.

VI.7. Матрицы и системы отношений

$$\begin{cases} 2x - y + 3z = 3, \\ x - y + 5z = 1. \end{cases}$$

Система уравнений получается, например, компиляцией следующего фрагмента.

VI.7. Матрицы и системы отношений

```
$\left\{\begin{array}{l}2x-y+3z=3,\\x-y+5z=1.\end{array}\right\}.
```

$$\begin{cases} 2x - y + 3z = 3, \\ x - y + 5z = 1. \end{cases}$$

Система уравнений получается, например, компиляцией следующего фрагмента.

VII. Команды $\text{\LaTeX}$

Большинство команд $\text{\TeX}$ начинается символом $\backslash$, например,

- команды для указания размера шрифта: $\backslash\text{Huge}$, $\backslash\text{LARGE}$;
- команды для математических символов: $\backslash\text{ge}$ (для символа $\geq$), $\backslash\text{cdot}$ (для символа $\cdot$), $\backslash\text{partial}$ (для символа ∂);
- команды для греческих букв: $\backslash\text{alpha}$ (для буквы α), $\backslash\text{epsilon}$ (для буквы ϵ), $\backslash\text{varepsilon}$ (для буквы ε),
- команды для специальных символов: $\backslash\text{S}$ (знак §); $\backslash\text{No}$ (знак №);
- команды для некоторых видов скобок: $\backslash\{, \backslash\}$, $\backslash\text{langle}$ (для $\langle$), $\backslash\text{rangle}$ (для $\rangle$);
- команды для стрелок: $\backslash\text{to}$ (для $\rightarrow$), $\backslash\text{Rightarrow}$ (для $\Rightarrow$), $\backslash\text{Uparrow}$ (для $\Uparrow$), $\backslash\text{Leftrightarrow}$ (для $\Leftrightarrow$).

VII. Команды L^AT_EX

Пробелы, следующие непосредственно за командой, компилятор T_EX воспринимает как знак окончания имени команды, т.е. эти пробелы на печати игнорируются. Например,

`$\partial abcd$, \S 7, \No 6`

после компиляции даёт:

$\partial abcd$, §7, №6

а не

$\partial abcd$, § 7, № у.

VII. Команды $\text{\LaTeX}$

Пробелы, следующие непосредственно за командой, компилятор $\text{\TeX}$ воспринимает как знак окончания имени команды, т.е. эти пробелы на печати игнорируются. Например,

$\text{\textbackslash partial } abcd\text{\textbackslash}$, $\text{\textbackslash S } 7$, $\text{\textbackslash No } 6$

после компиляции даёт:

$\partial abcd$, §7, №6

а не

$\partial abcd$, § 7, № у.

Если после команды необходим пробел, следует **вставить его принудительно**:

$\text{\textbackslash partial\ } abcd\text{\textbackslash}$, $\text{\textbackslash S\ } 7$, $\text{\textbackslash No\ } у$,

или, в текстовой моде, можно вставить «неразрывный пробел»:

$\text{\textbackslash S~}7$, $\text{\textbackslash No~}у$.

VII.1. Команды с аргументами

Результат выполнения многих команд зависит от аргументов, например команда `\hat{...}` для установления «домика» над буквой в математической моде:

`\hat{x}` для $\hat{x}$, `\hat{p}` для $\hat{p}$, `\hat{F}` для $\hat{F}$.

Аргументы могут быть обязательными и необязательными.

Как правило, обязательные аргументы указываются в фигурных скобках, а необязательные — в квадратных.

Но для некоторых команд обязательные аргументы указываются в круглых скобках, например, в команде `\qbezier`.

VII.2. Создание новых команд

Для создания новых команд $\text{\LaTeX}$ используются команды `\def`, `\newcommand` и `\renewcommand`.

VII.2. Создание новых команд

Для создания новых команд $\text{\LaTeX}$ используются команды

$\text{\code{\def}}$, $\text{\code{\newcommand}}$ и $\text{\code{\renewcommand}}$.

$\text{\code{\def}}$ — команда $\text{\TeX}$.

$\text{\code{\newcommand}}$ и $\text{\code{\renewcommand}}$ — команды $\text{\LaTeX}$.

VII.2. Создание новых команд

Для создания новых команд $\text{\LaTeX}$ используются команды `\def`, `\newcommand` и `\renewcommand`.

Разница между командами `\def` и `\newcommand` состоит в следующем. Допустим, ранее уже была введена команду `\abs`, но мы хотим переопределить эту команду с помощью `\def` или `\newcommand`.

VII.2. Создание новых команд

Для создания новых команд $\text{\LaTeX}$ используются команды `\def`, `\newcommand` и `\renewcommand`.

Разница между командами `\def` и `\newcommand` состоит в следующем. Допустим, ранее уже была введена команду `\abc`, но мы хотим переопределить эту команду с помощью `\def` или `\newcommand`.

Команда `\def` переопределит команду `\abc` без каких-либо затруднений.

VII.2. Создание новых команд

Для создания новых команд $\text{\LaTeX}$ используются команды `\def`, `\newcommand` и `\renewcommand`.

Разница между командами `\def` и `\newcommand` состоит в следующем. Допустим, ранее уже была введена команду `\abc`, но мы хотим переопределить эту команду с помощью `\def` или `\newcommand`.

Команда `\def` переопределит команду `\abc` без каких-либо затруднений.

Но выполнение команды `\newcommand` в этом случае вызовет сообщение об ошибке. Это удобно, если мы случайно использовали обозначение `\abc`, забыв, что она уже была определена ранее.

Если мы сознательно хотим переопределить `\abc`, следует использовать `\renewcommand`.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды},`

где n — количество аргументов (параметров), не превосходящее 9.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды},`

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

$$\backslash\mathrm{def}\backslash\mathrm{имя_команды}\{\mathrm{текст_команды}\} \quad \text{или}$$
$$\backslash\mathrm{def}\backslash\mathrm{имя_команды}\#1\#2\ldots\#n\{\mathrm{текст_команды}\},$$

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

$$\backslash\mathrm{Xxx}\{p\}\{q\}$$

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код (здесь цвет использован только для выделения гиперссылок):

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

`#1` — обозначение первого аргумента определяемой команды `\Xxx`,

`#2` — обозначение второго аргумента и т.п.

Всего может быть до 9 аргументов.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

$\#1$ — обозначение первого аргумента определяемой команды `\Xxx`,

$\#2$ — обозначение второго аргумента и т.п.

При выполнении команды `\Xxx{2}{5}` имеем $\#1=$ $\#2=$

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

`#1` — обозначение первого аргумента определяемой команды `\Xxx`,

`#2` — обозначение второго аргумента и т.п.

При выполнении команды `\Xxx{2}{5}` имеем `#1=2`, `#2=5`.

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

$\#1$ — обозначение первого аргумента определяемой команды `\Xxx`,

$\#2$ — обозначение второго аргумента и т.п.

При выполнении команды `\Xxx{4}{0}` имеем $\#1=$ $\#2=$

VII.2.1. Создание новых команд с помощью `\def`

Формат команды:

`\def\имя_команды{текст_команды}` или
`\def\имя_команды#1#2...#n{текст_команды}`,

где n — количество аргументов (параметров), не превосходящее 9.

Допустим, мы хотим определить команду `\Xxx` с двумя аргументами, такую, что

`\Xxx{p}{q}`

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код

`\def\Xxx#1#2{либо $#1<#2$, либо $#1\ge #2$}`.

`#1` — обозначение первого аргумента определяемой команды `\Xxx`,

`#2` — обозначение второго аргумента и т.п.

При выполнении команды `\Xxx{4}{0}` имеем `#1=4`, `#2=0`.

VII.2.2. Создание новых команд с помощью `\newcommand` и `\renewcommand`

Формат команды:

```
\newcommand{\имя_команды}{текст_команды}
```

или

```
\newcommand{\имя_команды}[число_параметров]{текст_команды}
```

VII.2.2. Создание новых команд с помощью `\newcommand` и `\renewcommand`

Формат команды:

$$\backslash\text{newcommand}\{\backslash\text{имя_команды}\}\{\text{текст_команды}\}$$

или

$$\backslash\text{newcommand}\{\backslash\text{имя_команды}\}[\text{число_параметров}]\{\text{текст_команды}\}$$

Допустим, мы хотим с помощью `\newcommand` определить команду `\Xxx` с двумя аргументами, такую, что

$$\backslash\text{Xxx}\{p\}\{q\}$$

выводит на печать текст

$$\text{либо } p < q, \text{ либо } p \geq q.$$

Для этого можно набрать в `*.tex`, например, следующий код (здесь цвет использован только для выделения гиперссылок):

VII.2.2. Создание новых команд с помощью `\newcommand` и `\renewcommand`

Формат команды:

$$\backslash\text{newcommand}\{\backslash\text{имя_команды}\}\{\text{текст_команды}\}$$

или

$$\backslash\text{newcommand}\{\backslash\text{имя_команды}\}[\text{число_параметров}]\{\text{текст_команды}\}$$

Допустим, мы хотим с помощью `\newcommand` определить команду `\Xxx` с двумя аргументами, такую, что

$$\backslash\text{Xxx}\{p\}\{q\}$$

выводит на печать текст

либо $p < q$, либо $p \geq q$.

Для этого можно набрать в `*.tex`, например, следующий код (здесь цвет использован только для выделения гиперссылок):

$$\backslash\text{newcommand}\{\backslash\text{Xxx}\}[2]\{\text{либо } \text{\$}\#1<\#2\text{\$, либо } \text{\$}\#1\geq \#2\text{\$}\}.$$

VII.3. Регистры и счётчики

В $\text{\LaTeX}$ определено большое число счётчиков и 256 регистров.

Счётчик может принимать только целые (в том числе отрицательные) значения. В регистре может храниться только целое число.

VII.3.1. Введение новых регистров и счётчиков

Новый регистр системы T_EX, например, \CountX можно ввести командой

```
\newcount{\CountX}
```

Новый счётчик L^AT_EX, например, CountY можно ввести командой

```
\newcounter{CountY}
```

Если счётчик с таким именем уже есть, при компиляции вы получите сообщение об ошибке.

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\arabic{section}`
получим

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\arabic{section}`
получим

«Текущее значение счётчика `section` равно 7».

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Напечатать значение счётчика строчными римскими цифрами можно командой `\roman {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\roman{section}`
получим

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Напечатать значение счётчика строчными римскими цифрами можно командой `\roman {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\roman{section}`

получим

«Текущее значение счётчика `section` равно `vii`».

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Напечатать значение счётчика строчными римскими цифрами можно командой `\roman {имя_счётчика}`

Напечатать значение счётчика заглавными римскими цифрами можно командой `\Roman {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\Roman{section}`
получим

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Напечатать значение счётчика строчными римскими цифрами можно командой `\roman {имя_счётчика}`

Напечатать значение счётчика заглавными римскими цифрами можно командой `\Roman {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика `section`
равно `\Roman{section}`
получим

«Текущее значение счётчика `section` равно VII».

VII.3.2. Печать значений счётчика

Напечатать значение счётчика арабскими цифрами можно командой `\arabic {имя_счётчика}`

Напечатать значение счётчика строчными римскими цифрами можно командой `\roman {имя_счётчика}`

Напечатать значение счётчика заглавными римскими цифрами можно командой `\Roman {имя_счётчика}`

Напечатать значение счётчика заглавными или, соответственно, строчными греческими цифрами можно командами `\Alph {имя_счётчика}` или `\alph {имя_счётчика}`

Например, в результате компиляции кода
Текущее значение счётчика section
равно `\Alph{section}= \alph{section}`
получим

«Текущее значение счётчика section равно G=g».

VII.3.3. Изменение значений счётчиков

Присвоение значения счётчику осуществляется командой

```
\setcounter {имя_счётчика}{значение_счётчика}
```

Например,

```
\setcounter {subsection}{-3}
```

установит значение счётчика `subsection` (т.е. номера подраздела) равным (-3) .

Если попытаться присвоить счётчику нецелое значение, при компиляции получим ошибку.

VII.3.3. Изменение значений счётчиков

Присвоение значения счётчику осуществляется командой

```
\setcounter {имя_счётчика}{значение_счётчика}
```

Добавление к счётчику некоторого целого значения (в том числе отрицательного) осуществляется командой

```
\addtocounter{CountX}{добавляемое_значение}
```

Например,

```
\setcounter{subsection}{-3}
```

уменьшит значение счётчика `subsection` (т.е. номера подраздела) на (-3) .

VII.3.3. Изменение значений счётчиков

Присвоение значения счётчику осуществляется командой

```
\setcounter {имя_счётчика}{значение_счётчика}
```

Добавление к счётчику некоторого целого значения (в том числе отрицательного) осуществляется командой

```
\addtocounter{CountX}{добавляемое_значение}
```

Однако использование команды `\addtocounter` не учитывается командами `\label` и `\mylabel`, т.е. даже сразу после применения команды `\addtocounter` команда `\label{YYY}` запомнит не новое значение счётчика YYY, а значение, получившееся после последнего применения команды

```
\refstepcounter{имя_счётчика}
```

после выполнения которой значение счётчика «имя_счётчика» будет увеличено на 1.

VII.3.4. Сравнение операций со счётчиками и регистрами

Действие	Регистр, например, <code>\reg</code>	Счётчик, например, <code>cnt</code>
Присвоение значения n	<code>\reg = n</code>	<code>\setcounter{cnt}{n}</code>
Суммирование с n	<code>\advance\reg by n</code>	<code>\addtocounter{cnt}{n}</code>
Умножение на n	<code>\multiply\reg by n</code>	нет
Деление на n	<code>\divide\reg by n</code>	нет
Печать значения арабскими цифрами	<code>\the\reg</code>	<code>\thecnt</code> <code>\arabic{cnt}</code>
строчными римскими цифрами		<code>\roman{cnt}</code>
заглавными римскими цифрами		<code>\Roman{cnt}</code>

VII.3.5. Команды сравнения и цикла

Подробнее о командах `\if... \fi` см. в работе [Дональда Кнута «Все про T_EX»](#).

Мы рассмотрим только один вариант вида: `\ifnum... \fi`

VII.3.5. Команды сравнения и цикла

При определении команд с помощью `\def`, `\newcommand`, и `\renewcommand` для реализации нескольких вариантов, в зависимости от значений параметров $\#1, \#2, \dots$, применяется конструкция

`\ifnum A<B код, если «Да» \else код, если «Нет» \fi`

`\ifnum A=B код, если «Да» \else код, если «Нет» \fi`

`\ifnum A>B код, если «Да» \else код, если «Нет» \fi`

Рассмотрим пример?

VII.3.5. Команды сравнения и цикла

При определении команд с помощью `\def`, `\newcommand`, и `\renewcommand` для реализации нескольких вариантов, в зависимости от значений параметров $\#1, \#2, \dots$, применяется конструкция

`\ifnum A<B код, если «Да» \else код, если «Нет» \fi`

`\ifnum A=B код, если «Да» \else код, если «Нет» \fi`

`\ifnum A>B код, если «Да» \else код, если «Нет» \fi`

Команда выбора из нескольких вариантов:

`\ifcase A {код, если  $A = 0$ } \or... \or {код, если  $A = k$ } \fi`

Рассмотрим пример?

VII.3.5. Команды сравнения и цикла

При определении команд с помощью `\def`, `\newcommand`, и `\renewcommand` для реализации нескольких вариантов, в зависимости от значений параметров $\#1, \#2, \dots$, применяется конструкция

```
\ifnum A<B код, если «Да» \else код, если «Нет» \fi  
\ifnum A=B код, если «Да» \else код, если «Нет» \fi  
\ifnum A>B код, если «Да» \else код, если «Нет» \fi
```

Команда выбора из нескольких вариантов:

```
\ifcase A {код, если  $A = 0$ } \or... \or {код, если  $A = k$ } \fi
```

Для организации цикла применяется конструкция

```
\loop Предварительный код
```

```
\ifnum A<B тело цикла, выполняемое, пока «Да»
```

```
\repeat
```

Обратите внимание на отсутствие в команде цикла команды `\fi`, обычно закрывающей `\ifnum...` **Рассмотрим пример?**

VII.3.6. Регистры: введение новых регистров

TeX и L^AT_EX используют 256 регистров, в которых может храниться целое число.

Автор может определить собственное имя регистра, и с помощью команды `newcounter` для выбранного имени L^AT_EX выбирает свободный регистр.

Например,
`newcounter\Abcde` и `newcounter\Uytr` присваивают именам `\Abcde` и `\Uytr` определённые регистры, или, иначе говоря, присваивают некоторым свободным к этому времени регистрам имена `\Abcde` и `\Uytr`.

VII.3.7. Регистры: печать и изменение значений

TeX и LaTeX используют 256 регистров, в которых может храниться целое число.

Автор может определить собственное имя регистра, и с помощью команды `newcounter` для выбранного имени LaTeX выбирает свободный регистр.

Изменить значения регистра, например, `\Abc` можно командами `advance\Abc n` (увеличить значение `\Abc` на n), `multiply\Abc n` (умножить значение `\Abc` на n), `divide\Abc n` (заменить значение `\Abc` на целую часть от деления его на n), `\the\Abc` (напечатать значение регистра `\Abc`).

Рассмотреть пример?

VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра

Если загрузить файл `MakeNumbb.inp`, например, командой `\input MakeNumbb.inp`, то будет определена команда `\MakeNumbb{name}{k}`, которая при обращении к ней создает команду `\numb`, выдающая на выходе число n .

Требуется, чтобы был определён **регистр** `\CountTmp`.

Этот регистр уже определён в случае использования преамбул `testsample.tex`, `StartTestBigScr.tex`.

VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра

Если загрузить файл `MakeNumbb.inp`, например, командой `\input MakeNumbb.inp`, то будет определена команда `\MakeNumbb{name}{k}`, которая при обращении к ней создает команду `\numb`, выдающая на выходе число n .

Например, результат выполнения кода

```
\setcounter{footnote}{3}   \arabic{footnote}:  
\MakeNumbb{AbCd}{\arabic{footnote}}   \AbCd::  
\setcounter{footnote}{4}   \arabic{footnote},\AbCd:::
```

будет таким:

VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра

Если загрузить файл `MakeNumbb.inp`, например, командой `\input MakeNumbb.inp`, то будет определена команда `\MakeNumbb{name}{k}`, которая при обращении к ней создает команду `\numb`, выдающая на выходе число n .

Например, результат выполнения кода

```
\setcounter{footnote}{3} \arabic{footnote}:  
\MakeNumbb{AbCd}{\arabic{footnote}} \AbCd:::  
\setcounter{footnote}{4} \arabic{footnote},\AbCd:::
```

будет таким:

3: 3:: 4,3:::

VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра

Если загрузить файл `MakeNumbb.inp`, например, командой `\input MakeNumbb.inp`, то будет определена команда `\MakeNumbb{name}{k}`, которая при обращении к ней создает команду `\numb`, выдающая на выходе число n .

Например, результат выполнения кода

```
\setcounter{footnote}{3} \arabic{footnote}:  
\MakeNumbb{AbCd}{\arabic{footnote}} \AbCd::  
\setcounter{footnote}{4} \arabic{footnote},\AbCd:::
```

будет таким:

3: 3:: 4,3:::

Таким образом, результат выполнения команды `\AbCd` не изменился, несмотря на изменение значения **счётчика** `footnote`.

VII.3.8. Команда для запоминания числа, в частности, текущего значения счётчика или регистра

Если загрузить файл `MakeNumbb.inp`, например, командой `\input MakeNumbb.inp`, то будет определена команда `\MakeNumbb{name}{k}`, которая при обращении к ней создает команду `\numb`, выдающая на выходе число n .

Например, результат выполнения кода

```
\setcounter{footnote}{3} \arabic{footnote}:  
\MakeNumbb{AbCd}{\arabic{footnote}} \AbCd::  
\setcounter{footnote}{4} \arabic{footnote},\AbCd:::
```

будет таким:

3: 3:: 4,3:::

Результат был бы иным, например, если определить `\def\AbCd{\arabic{footnote}}`.

VII.3.9. Команда печати простого числа из $[1; 999]$

Если загрузить файл `MakeSimpleNumb.inp`, например, командой `\input MakeSimpleNumb.inp`, то будет определена команда `\MakeSimpleNumb{k}`, которая при $1 \leq n \leq 168$ печатает n -е простое число.

При других значениях n , не кратных 168, будут печататься простые числа с номерами, вычисляемыми по модулю 168, т.е. с номерами, равными остатку от деления на 168.

Для чисел, кратных 168, будет печататься 168-е простое число, т.е. число 997.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Это страница номер 329.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Например, в данном месте мы поставили гипермишень

```
\hypertarget {AboutHypertarget}{} 
```

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

`\hypertarget {имя_гипермишени}{текст}`

Метка ставится командами

`\label {имя_метки}` или `\mylabel {имя_метки}`

Например, в данном месте, на странице 331, мы поставили метку `\mylabel {Aboutlabel}`

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Можно сослаться на гипермишень командами

```
\hyperlink{имя_мишени}{Текст гиперссылки}
```

(если мишень находится в том же документе) или

```
\href{имя_файла#имя_мишени}{Текст гиперссылки}
```

(если мишень находится в файле с именем имя_файла).

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Можно сослаться на гипермишень командами

```
\hyperlink{имя_мишени}{Текст гиперссылки}
```

(если мишень находится в том же документе) или

```
\href{имя_файла#имя_мишени}{Текст гиперссылки}
```

(если мишень находится в файле с именем имя_файла).

Если при компиляции в последнем случае возникает сообщение об ошибке (например, этот текст входит в состав другой команды), следует заменить # на \#.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

`\hypertarget {имя_гипермишени}{текст}`

Метка ставится командами

`\label {имя_метки}` или `\mylabel {имя_метки}`

Значение метки (т.е. запомненное значение счётчика) можно распечатать командой `\ref{имя_метки}`.

Номер страницы, на которой находится метка, можно распечатать командой `\pageref{имя_метки}`.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Данная ссылка на **гипермишень** `AboutHypertarget` была поставлена командой

```
\hyperlink{AboutHypertarget}  
{\textbf{гипермишень} \texttt{AboutHypertarget}}.
```

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Данная ссылка на **гипермишень AboutHypertarget** была поставлена командой

```
\hyperlink{AboutHypertarget}  
{\textbf{гипермишень} \texttt{AboutHypertarget}}.
```

Второй аргумент команды `\hyperlink` мы напечатали с новой строки только из типографских соображений. Новая строка не обязательна, `\hyperlink{AboutHypertarget}{гиперссылка}` тоже скомпилируется как **гиперссылка**.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Ранее мы поставили метку Aboutlabel командой `\mylabel`.

В данном случае ссылку мы сделали командой

```
\hyperlink{HTAboutlabel}{\textbf{Ранее}}
```

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки} или \mylabel {имя_метки}
```

Ранее мы поставили метку Aboutlabel командой `\mylabel`.

Выполнение кода `\hyperlink{HTAboutlabel}{метка}`

```
\ref{Aboutlabel} на стр. \pageref{Aboutlabel}
```

приведёт к следующему результату: «метка VII.4 на стр. 331».

Обратите внимание, что в гиперссылке 331 верно указан номер страницы, но переход осуществляется на ту страницу, на которой в последний раз изменялось значение счётчика, в данном случае, счётчика subsection.

VII.4. Гипермишени и гиперссылки

У автора имеется возможность поставить в тексте метки и гипермишени для организации ссылок и переходов по гиперссылкам.

Для вставки в текст гипермишени применяется команда

```
\hypertarget {имя_гипермишени}{текст}
```

Метка ставится командами

```
\label {имя_метки}    или \mylabel {имя_метки}
```

Ранее мы поставили метку Aboutlabel командой `\mylabel`.

Выполнение кода `\hyperlink{HTAboutlabel}{метка}`

```
\ref{Aboutlabel} на стр. \pageref{Aboutlabel}
```

приведёт к следующему результату: «метка VII.4 на стр. 331».

Рассмотрим пример?

VIII. Секционирование текста для слайдов

В случае, если преамбула определяется файлом `texsample.pdf`, определены команды `\PdfSection`, `\PdfSubSection`, `\PdfSubSubSection`, имеющие по 9 аргументов.

VIII. Секционирование текста для слайдов

В случае, если преамбула определяется файлом `texsample.pdf`, определены команды `\PdfSection`, `\PdfSubSection`, `\PdfSubSubSection`, имеющие по 9 аргументов.

Первый аргумент `#1` — название раздела, второй аргумент `#2` — число 0 или 1. Если `#2=0`, то название раздела будет помещено в оглавление, создаваемое командой `\tableofcontents`, а если `#2=1` — то не будет.

VIII. Секционирование текста для слайдов

В случае, если преамбула определяется файлом `texsample.pdf`, определены команды `\PdfSection`, `\PdfSubSection`, `\PdfSubSubSection`, имеющие по 9 аргументов.

Первый аргумент `#1` — название раздела, второй аргумент `#2` — число 0 или 1. Если `#2=0`, то название раздела будет помещено в оглавление, создаваемое командой `\tableofcontents`, а если `#2=1` — то не будет.

Аргументы `#3–#9` в этих командах пока не задействованы.

VIII. Секционирование текста для слайдов

В случае, если преамбула определяется файлом `texsample.pdf`, определены команды `\PdfSection`, `\PdfSubSection`, `\PdfSubSubSection`, имеющие по 9 аргументов.

Первый аргумент `#1` — название раздела, второй аргумент `#2` — число 0 или 1. Если `#2=0`, то название раздела будет помещено в оглавление, создаваемое командой `\tableofcontents`, а если `#2=1` — то не будет.

Аргументы `#3–#9` в этих командах пока не задействованы.

Рекомендуется на первом слайде нового раздела использовать эти команды с параметром `#2=0`, а на последующих слайдах этого раздела — с параметром `#2=1`.

VIII. Секционирование текста для слайдов

В случае, если преамбула определяется файлом `texsample.pdf`, определены команды `\PdfSection`, `\PdfSubSection`, `\PdfSubSubSection`, имеющие по 9 аргументов.

Первый аргумент `#1` — название раздела, второй аргумент `#2` — число 0 или 1. Если `#2=0`, то название раздела будет помещено в оглавление, создаваемое командой `\tableofcontents`, а если `#2=1` — то не будет.

Аргументы `#3–#9` в этих командах пока не задействованы.

Рекомендуется на первом слайде нового раздела использовать эти команды с параметром `#2=0`, а на последующих слайдах этого раздела — с параметром `#2=1`.

При этом на первом слайде нового раздела, с параметром `#2=0`, можно ставить метки командой `\mylabel{...}`

IX. Окружения и форматирование текста

IX.1. Окружения

В системе $\text{\LaTeX}$ применяется система окружений, т.е. выделения фрагментов кода, компилируемых определённым образом (фиксируется выбор шрифта, изменяются значения соответствующих счётчиков и др.).

Окружение — это часть кода, которая находится между `\begin{имя_окружения}` и `\end{имя_окружения}`.

Основным является окружение `document`. Важными являются окружения `equation`, `picture`, `array` и окружения **типа «теорема»**, описанные в **разделе X**.

IX.2. Форматирование текста

Перейти к новому абзацу можно только в текстовой моде. Для этого достаточно вставить в текст пустую строку или команду `\par`. Перейти к новой строке в текстовой моде можно с помощью команды `\newline`.

IX.2. Форматирование текста

Если необходимо выровнять *одну строку* по центру, следует применить команду `\centerline`:

Прежде чем перейти к строке, выровненной по центру, надо перейти к новому абзацу, или завешить текст командой

`\symbol{92}newline.\newline`

`\centerline{Теперь будет напечатана ОДНА строка по центру.}`

В конце «центрируемой» строки ставить команду

`\symbol{92}newline` не обязательно

после компиляции даёт следующий результат:

Прежде чем перейти к строке, выровненной по центру, надо перейти к новому абзацу, или завешить текст командой `\newline`.

Теперь будет напечатана ОДНА строка по центру.

В конце «центрируемой» строки ставить команду `\newline` не обязательно.

IX.2. Форматирование текста

Если необходимо выровнять по центру *несколько абзацев*, следует включить их в **окружение** «`center`». Например код,

```
«\begin{center}
```

Первый абзац.`\par` Второй абзац.`\par` Абзацы могут занимать несколько строк, содержать формулы и рисунки, даже переходить на следующую страницу.

```
\end{center}» в результате компиляции даёт следующий результат:
```

Первый абзац.

Второй абзац.

Абзацы могут занимать несколько строк, содержать формулы и рисунки, даже переходить на следующую страницу.

IX.2. Форматирование текста

С более сложными вариантами форматирования можно ознакомиться, например, в книге [С.М. Львовского «Набор и верстка в системе \$\text{\LaTeX}\$ »](#), стр. 115-140.

IX.3. Расстояние между абзацами

Вертикальный пробел между абзацами можно увеличить, в частности, с помощью команд

`\medskip`, `\bigskip`, `\vspace`, кратко описанных ниже.

IX.3. Расстояние между абзацами

Вертикальный пробел между абзацами можно увеличить, в частности, с помощью команд

`\medskip`, `\bigskip`, `\vspace`.

Перед этим абзацем расстояние обычное.

Здесь расстояния между абзацами тоже обычное.

Команды `\medskip` и `\bigskip` увеличивают расстояние перед тем абзацем, в котором размещены эти команды. Поэтому, если необходимо увеличить это расстояние, то лучше начинать абзац с одной из этих команд.

IX.3. Расстояние между абзацами

Вертикальный пробел между абзацами можно увеличить, в частности, с помощью команд

`\medskip`, `\bigskip`, `\vspace`.

Перед этим абзацем расстояние обычное.

`\medskip`

Команды `\medskip` и `\bigskip` увеличивают расстояние перед тем абзацем, в котором размещены эти команды. Поэтому, если необходимо увеличить это расстояние, то лучше начинать абзац с одной из этих команд.

Здесь использована команда `\medskip`.

IX.3. Расстояние между абзацами

Вертикальный пробел между абзацами можно увеличить, в частности, с помощью команд

`\medskip`, `\bigskip`, `\vspace`.

Перед этим абзацем расстояние обычное.

`\bigskip`

Команды `\medskip` и `\bigskip` увеличивают расстояние перед тем абзацем, в котором размещены эти команды. Поэтому, если необходимо увеличить это расстояние, то лучше начинать абзац с одной из этих команд.

Здесь использована команда `\bigskip`.

IX.3. Расстояние между абзацами

Вертикальный пробел между абзацами можно увеличить, в частности, с помощью команд

`\medskip`, `\bigskip`, `\vspace`.

Перед этим абзацем расстояние обычное.

```
\par \vspace{10mm}
```

Команды `\medskip` и `\bigskip` увеличивают расстояние перед тем абзацем, в котором размещены эти команды. Поэтому, если необходимо увеличить это расстояние, то лучше начинать абзац с одной из этих команд.

А сейчас между абзацами была вставлена команда `\vspace{10mm}`.

Х. Работа с окружением типа «теорема»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Первый аргумент `#1`, обозначенный как `???`, принимает значения `quest` (для окружения «вопрос»), `zam` (для окружения «замечание»), `lmm` (для окружения «лемма»), `defnt` (для окружения «определение»), `cons` (для окружения «следствие»), `thm` (для окружения «теорема»), `prop` (для окружения «утверждение»), `aim` (для окружения «цель»), `law` (для окружения «закон»), `regul` (для окружения «правило»).

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Первый аргумент `#1`, обозначенный как `???`, принимает значения `quest` (для окружения «вопрос»),

`zam` (для окружения «замечание»),

`lmm` (для окружения «лемма»),

`defnt` (для окружения «определение»),

`cons` (для окружения «следствие»),

`thm` (для окружения «теорема»),

`prop` (для окружения «утверждение»),

`aim` (для окружения «цель»),

`law` (для окружения «закон»),

`regul` (для окружения «правило»).

Каждому из этих окружений соответствует одноимённый **счётчик**, т.е. определены **счётчики** `defnt`, `lmm` и др.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Первый аргумент `#1`, обозначенный как `???`, может также принимать значения

`prim` (для окружения «Пример»),

`zad` (для окружения «Задача»),

хотя для них лучше использовать специальные команды, описанные ниже.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

`\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}`

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

`\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}`

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

```
\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности  
к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}
```

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

`\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}`

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

```
\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности  
к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}
```

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Результат выполнения `\ComTextthmForDefPi{}}{}}{}}{}}{}}{}}`:

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Второй аргумент `#2` представляет фрагмент уникальной метки для рассматриваемого окружения. Например,

`\ThmEnvCom{thm}{ForDefPi}{Отношение длины окружности к ее диаметру равно 3,141592....}}{}}{}}{}}{}}{}}`

после компиляции приведет, во-первых, к определению команды `\ComTextthmForDefPi` с 6 аргументами, во-вторых, к определению метки `thmForDefPi` и гипермишени `HTthmForDefPi`.

Результат выполнения `\ComTextthmForDefPi{}}{}}{}}{}}{}}{}}`:

Теорема 1. *Отношение длины окружности к ее диаметру равно 3,141592....*

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Для ввода нумерованной формулы в тексте определения, теоремы и т.д. в аргументе `#3`, можно применить команду `\ThmEnvEquat{}}{}}{}}{}}`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`

`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}{}}`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`
`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}{}}`.

При выполнении `\ComTextdefntAAA{#1}{0}{3}{4}{5}{6}` будет напечатан текст определения, увеличены счетчики `defnt` и `equation`, новое значение счетчика `equation` будет присвоено метке `EquatdefntAAA` и этим значением счетчика будет отмечена формула $\ln x^3 = 3 \ln x$.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`
`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}`.

При выполнении `\ComTextdefntAAA{#1}{0}{3}{4}{5}{6}` будет напечатан текст определения, увеличены счетчики `defnt` и `equation`, новое значение счетчика `equation` будет присвоено метке `EquatdefntAAA` и этим значением счетчика будет отмечена формула $\ln x^3 = 3 \ln x$.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`
`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}{}}`.

При выполнении `\ComTextdefntAAA{#1}{0}{3}{4}{5}{6}` будет напечатан текст определения, увеличены счетчики `defnt` и `equation`, новое значение счетчика `equation` будет присвоено метке `EquatdefntAAA` и этим значением счетчика будет отмечена формула $\ln x^3 = 3 \ln x$.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`
`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}{}}`.

При выполнении `\ComTextdefntAAA{#1}{0}{3}{4}{5}{6}` будет напечатан текст определения, увеличены счетчики `defnt` и `equation`, новое значение счетчика `equation` будет присвоено метке `EquatdefntAAA` и этим значением счетчика будет отмечена формула $\ln x^3 = 3 \ln x$.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Например, в результате выполнения

`\ThmEnvCom{defnt}{AAA}{Формула в тексте определения:`

`\ThmEnvEquat{#1}{#2}{EquatdefntAAA}{\ln{x^3}=3\ln{x},}`
`\newline` продолжение текста определения.}

будет определена команда `\ComTextdefntAAA{}}{}}{}}{}}{}}`.

В дальнейшем следует применять

`\ComTextdefntAAA{#1}{1}{3}{4}{5}{6}`.

Х.1. Окружения «Вопрос», «Цель», «Определение», «Теорема», «Лемма», «Замечание», «Следствие», «Утверждение», «Закон», «Правило»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\ThmEnvCom{???}{!!!}{}}{}}{}}{}}{}}{}}`

Рассмотрим примеры?

Х.2. Задачи для самостоятельного решения и иллюстративные примеры

Презентации для сопровождения практических занятий и для решения примеров на уроках мы разделяем на два типа:

Х.2. Задачи для самостоятельного решения и иллюстративные примеры

Презентации для сопровождения практических занятий и для решения примеров на уроках мы разделяем на два типа:

— задачи, предназначенные для самостоятельного решения;

Х.2. Задачи для самостоятельного решения и иллюстративные примеры

Презентации для сопровождения практических занятий и для решения примеров на уроках мы разделяем на два типа:

- задачи, предназначенные для самостоятельного решения;
- задачи, предназначенные для иллюстрации теоретического материала, демонстрации образцов решения задач (методов решения и применения этих методов).

Х.3. Требования к оформлению задач для самостоятельного решения

Мы исходим из следующих соображений.

Во-первых,

Х.3. Требования к оформлению задач для самостоятельного решения

Мы исходим из следующих соображений.

Во-первых, условия задач должны быть отделены от ответа или описания решений.

Во-вторых,

Х.3. Требования к оформлению задач для самостоятельного решения

Мы исходим из следующих соображений.

Во-первых, условия задач должны быть отделены от ответа или описания решений.

Во-вторых, для задач, предназначенных для закрепления знаний и умений, возможно ограничиться представлением только результата вычислений, сокращая или опуская промежуточные выкладки.

В-третьих,

Х.3. Требования к оформлению задач для самостоятельного решения

Мы исходим из следующих соображений.

Во-первых, условия задач должны быть отделены от ответа или описания решений.

Во-вторых, для задач, предназначенных для закрепления знаний и умений, возможно ограничиться представлением только результата вычислений, сокращая или опуская промежуточные выкладки.

В-третьих, описание решения задачи, предназначенной для самостоятельного решения, предназначено, в первую очередь для контроля правильности выкладок, проведенных учащимися или студентами. Поэтому допустимо, и даже желательно, чтобы демонстрация *решения* доставляла обучаемым определенный дискомфорт, например, за счет более мелкого шрифта, компактной записи и др.

Х.4. Оформление задач для самостоятельного решения

Во-первых, номер задачи для самостоятельного решения определяется счетчиком `MakExercisePdf`, поэтому перед первой из таких задач следует обнулить счетчик командой `\setcounter{MakExercisePdf}{0}`. Счетчик `MakExercisePdf` является подчиненным счетчику `section`.

Х.4. Оформление задач для самостоятельного решения

Во-первых, номер задачи для самостоятельного решения определяется счетчиком `MakExecisePdf`, поэтому перед первой из таких задач следует обнулить счетчик командой `\setcounter{MakExecisePdf}{0}`. Счетчик `MakExecisePdf` является подчиненным счетчику `section`.

Во-вторых, если все задачи для самостоятельного решения выделены в отдельный раздел, то желательно выделять подразделы с различными блоками задач, например, задачи по определенной теме или задачи, ориентированные на формирование и закрепление определенных умений (отнесение дифференциального уравнения к определенному типу, умение переходить от одного типа моделей к другому и др.).

Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```


Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```

В процессе компиляции
этой команды первоначально
печатается «Текст задачи».

Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```

Для формирования слайдов с ответами и/или решениями необходимо выполнить **специальную команду**, описанную ниже.

Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```

При этом в оглавление, создаваемое командой `\tableofcontents`, вставляется строка «**Задача ...**», где вместо ... вставляется значение счетчика `MakeExercisePdf`, подчиненного счетчику `section`.

Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```

При формировании слайдов с
ответами и решениями:

команда `\MakeExNewpage`
используется для перехода к
новому слайду, на котором по-
вторяется текст задачи и слово
Ответ.

Х.4. Оформление задач для самостоятельного решения

В-третьих, задача оформляется следующим образом.

```
\MakeExercisePdf{Текст задачи.  
                }{
```

Начало ответа или решения.

```
\MakeExNewpage
```

Слайд с фрагментом решения
и условием задачи.

```
\ClearExNewpage
```

Слайд с фрагментом решения
без условия задачи.

```
\MakeExNewpage...}
```

При формировании слайдов с ответами и решениями:

команда `\ClearExNewpage` применяется для перехода к новому слайду, на котором текст условия задачи не повторяется и не печатается слово **Ответ.**

Х.4. Оформление задач для самостоятельного решения

В-четвертых, для «распечатки» (выведения на слайды) ответов ко всем задачам применяется команда `\MakePrinAnsw`.

Эта команда применяется в презентации один раз, поскольку она обеспечивает создание слайдов с ответами сразу ко всем задачам, сформированным командами `\MakeExercisePdf`.

Рассмотрим пример?

Х.5. Окружение «Пример»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\PrimCom{описание...}{!!!}{ }{ }{ }{ }{ }{ }`

Х.5. Окружение «Пример»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\PrimCom{описание...}{!!!}{ }{ }{ }{ }{ }{ }`

Первый аргумент `#1`, обозначенный как «описание» представляет собой краткое описание примера, которое будет потом вынесено в оглавление, создаваемое командой `\tableofcontents`.

Х.5. Окружение «Пример»

В случае, если преамбула определяется файлом `texsample.tex`, определена команда `\PrimCom{описание...}{!!!}{ }{ }{ }{ }{ }{ }`

Первый аргумент `#1`, обозначенный как «описание» представляет собой краткое описание примера, которое будет потом вынесено в оглавление, создаваемое командой `\tableofcontents`.

Второй аргумент `#2`, обозначенный как `!!!` представляет собой фрагмент уникальной метки примера.

Рассмотрим пример?

Х.6. Последовательное применение нескольких од- нотипных окружений типа «теорема»

Рассмотрим случай, когда на одном слайде надо разместить несколько определений, теорем и др.

Х.6. Последовательное применение нескольких од- нотипных окружений типа «теорема»: первый слайд

```
\ThmEnvCom{defnt}{FirstMind}{\textbf{Гениальность}} ---  
это суперпрофессионализм...}{ }{ }{ }{ }{ }
```

```
\ThmEnvCom{defnt}{SecondMind}{\textbf{Мудрость}} ---  
это умение предвидеть...}{ }{ }{ }{ }{ }
```

```
\ComTextdefntFirstMind{1}{2}{3}{4}{5}{6}
```

```
\ComTextdefntSecondMind{1}{2}{3}{4}{5}{6}
```

Определение **1**. *Гениальность* — это суперпрофессионализм...

Определение **2**. *Мудрость* — это умение предвидеть...

Х.6. Последовательное применение нескольких од- нотипных окружений типа «теорема»: следующие слайды

```
\addtocounter{defnt}{-2}\refstepcounter{defnt}  
\ComTextdefntFirstMind{1}{2}{3}{4}{5}{6}  
  \refstepcounter{defnt}  
\ComTextdefntSecondMind{1}{2}{3}{4}{5}{6}
```

Определение **1**. *Гениальность* — это суперпрофессионализм...

Определение **2**. *Мудрость* — это умение предвидеть...

Рассмотрим пример?

XI. Создание и импорт рисунков

Шаблоны для реализации описанных ниже конструкций, находятся в файле `000ris.tex`

XI.1. Импорт рисунков

L^AT_EX позволяет импортировать рисунки различного формата: jpg, png и др.

Для этого применяется команда `\includegraphics`.

Формат команды:

`\includegraphics` $\underbrace{[\text{scale}=\text{«число»}]}$ {имя_файла},

$\underbrace{\hspace{1.5cm}}_{\text{масштаб}}$
`\includegraphics` $\underbrace{[\text{width}=\text{«длина»}]}_{\text{ширина}}$ {имя_файла},

$\underbrace{\hspace{1.5cm}}_{\text{высота}}$
`\includegraphics` $\underbrace{[\text{height}=\text{«длина»}]}_{\text{высота}}$ {имя_файла},

`\includegraphics` $[\text{width}=\text{«длина»}, \text{height}=\text{«длина»}]$ {имя_файла}.

Здесь под «длиной» понимается число с указанием единиц длины или типовая T_EXовская единица с коэффициентом, например, `0.3\textwidth`.

XI.1. Импорт рисунков

L^AT_EX позволяет импортировать рисунки различного формата: jpg, png и др.

Для этого применяется команда `\includegraphics`.

Формат команды:

`\includegraphics` $\underbrace{[\text{scale}=\text{«число»}]}_{\text{масштаб}} \{\text{имя_файла}\},$

`\includegraphics` $\underbrace{[\text{width}=\text{«длина»}]}_{\text{ширина}} \{\text{имя_файла}\},$

`\includegraphics` $\underbrace{[\text{height}=\text{«длина»}]}_{\text{высота}} \{\text{имя_файла}\},$

`\includegraphics` $[\text{width}=\text{«длина»}, \text{height}=\text{«длина»}] \{\text{имя_файла}\}.$

В последнем случае, естественно, возможно искажение пропорций в рисунке, см. последующие слайды.

XI.1. Импорт рисунков

```
\includegraphics[scale=0.15]{yury.jpg}
```



XI.1. Импорт рисунков

```
\includegraphics[scale=0.15]{yury.jpg}
```



```
\includegraphics[scale=0.3]{yury.jpg}
```



XI.1. Импорт рисунков

```
\includegraphics[scale=0.15]{yury.jpg}
```



```
\includegraphics[scale=0.3]{yury.jpg}
```



```
\includegraphics[width=20mm]{yury.jpg}
```



20 mm

XI.1. Импорт рисунков

```
\includegraphics[scale=0.15]{yury.jpg}
```



```
\includegraphics[scale=0.3]{yury.jpg}
```



```
\includegraphics[width=20mm]{yury.jpg}
```



20 mm

```
\includegraphics[height=20mm]{yury.jpg}
```



XI.1. Импорт рисунков

```
\includegraphics[scale=0.15]{yury.jpg}
```



```
\includegraphics[scale=0.3]{yury.jpg}
```



```
\includegraphics[width=20mm]{yury.jpg}
```



20 mm

```
\includegraphics[height=20mm]{yury.jpg}
```

20 mm



```
\includegraphics[width=15mm,height=10mm]{yury.jpg}
```



XI.2. Особенности создания рисунков

Для рисунков, создаваемых непосредственно в T_EX, применяется окружение

```
\begin{picture}(width,height)  
\end{picture},
```

где `width` и `height` — ширина и, соответственно, высота бокса, отводимого компилятором для рисунка. Единица длины хранится в переменной `\unitlength`.

XI.2. Особенности создания рисунков

Для рисунков, создаваемых непосредственно в $\text{\TeX}$, применяется окружение

```
\begin{picture}(width,height)  
\end{picture},
```

где `width` и `height` — ширина и, соответственно, высота бокса, отводимого компилятором для рисунка. Единица длины хранится в переменной `\unitlength`.

Возможно применение необязательных параметров ax , ay

```
\begin{picture}(width,height)(ax,ay)  
\end{picture}
```

Они означают смещение рисунка относительно отведенного ему бокса на $(-ax)$ единиц по горизонтали (положительное направление — вправо) и на $(-ay)$ по вертикали (положительное направление — вверх).

XI.3. Команды для рисования

Для рисования используются команды `\put`, `\line`, `\vector`, `\circle`, `\multiput` `\bigcircle` (при использовании стилевого файла `curves.sty`), `\arc`, `\curve` `\qbezier` и др.

XI.3.1. Псевдорисунки: окружение `picture`

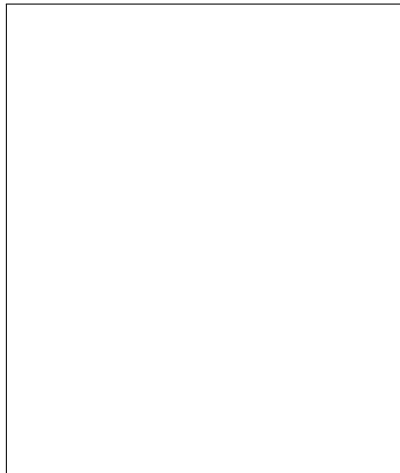


```
\begin{picture}(...)(...)
```

```
...
```

```
\end{picture}
```


XI.3.1. Псевдорисунки: окружение `picture`



```
\begin{picture}(...)(...)
```

```
...
```

```
\end{picture}
```

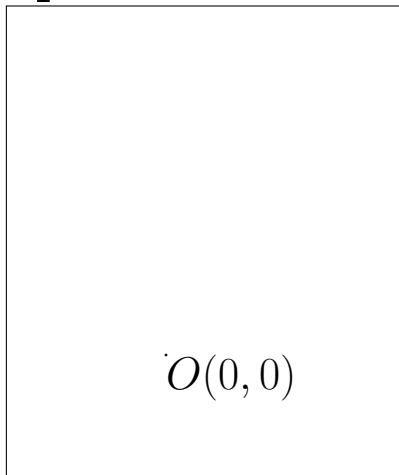
Выполнили команду

```
\begin{picture}(050.00,080.00)(-20.00,-30.00)
```

```
...
```

```
\end{picture}
```

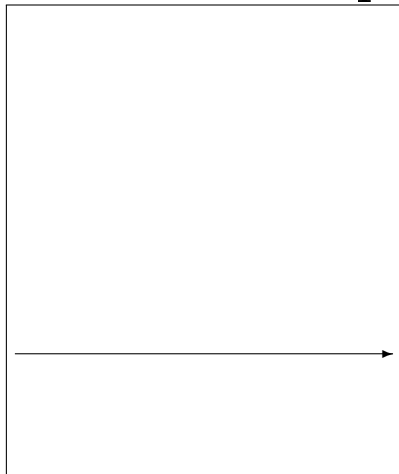
XI.3.2. Помещение объекта в точку с данными координатами: команда `\put`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(000.00,000.00){\circle{0.5}}  
\end{picture}
```

`\put` $(x,y)\{\dots\}$ — поместить объект $\dots$ в точку с координатами $(x; y)$.

XI.3.3. Вектор: команда `\vector`

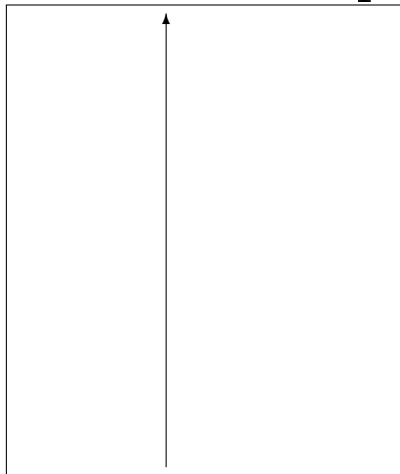


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(-20.00,000.00){\vector( 1, 0){50.00}}  
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0,
причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$,
 d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d —
проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

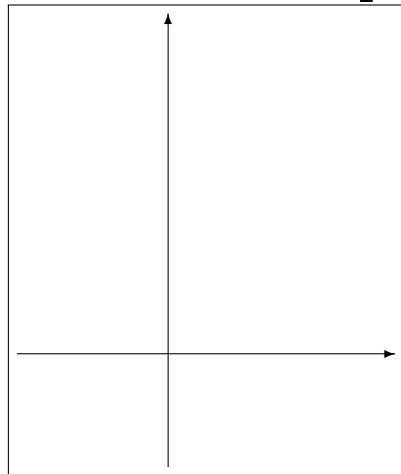


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(000.00,-15.00){\vector( 0, 1){60.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

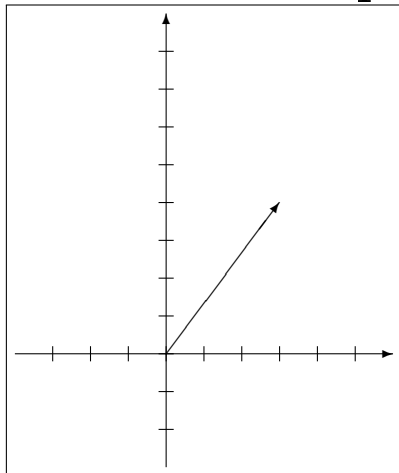


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-20.00,000.00){\vector( 1, 0){50.00}}
  \put(000.00,-15.00){\vector( 0, 1){60.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

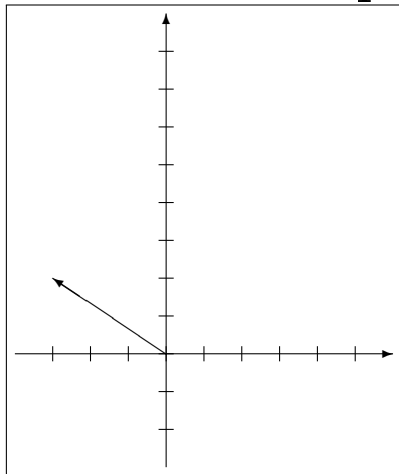


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(000.00,000.00){\vector( 3, 4){15.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

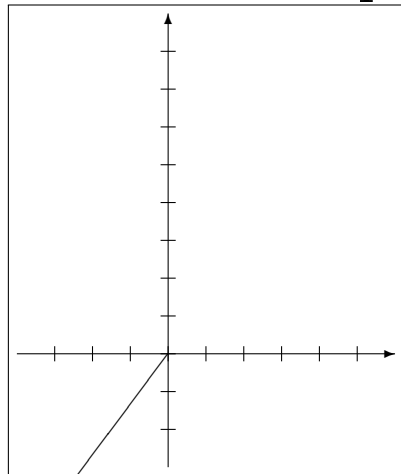


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(000.00,000.00){\vector(-3, 2){15.00}}  
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

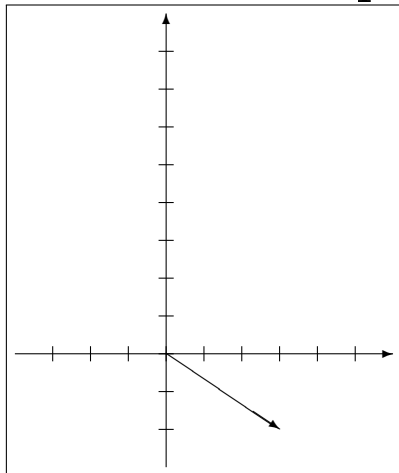


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(000.00,000.00){\vector(-3,-4){15.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

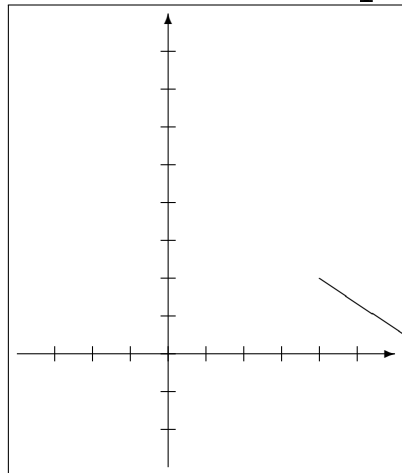


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(000.00,000.00){\vector( 3,-2){15.00}}  
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

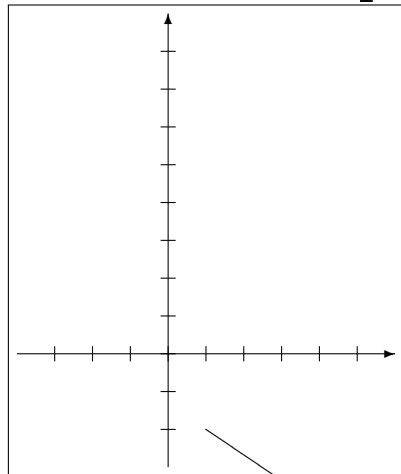


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(020.00,010.00){\vector( 3,-2){15.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.3. Вектор: команда `\vector`

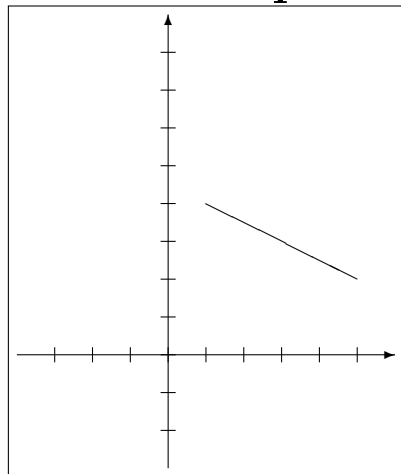


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(005.00,-10.00){\vector( 3,-2){15.00}}
\end{picture}
```

`\vector( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-4 \leq a \leq 4$, $-4 \leq b \leq 4$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.4. Отрезок: команда `\line`

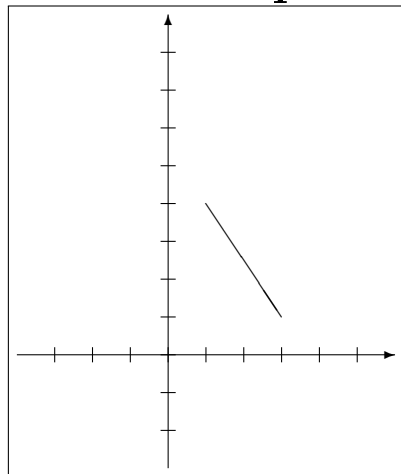


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(005.00,-20.00){\line( 2,-1){20.00}}
\end{picture}
```

`\line( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-6 \leq a \leq 6$, $-6 \leq b \leq 6$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.4. Отрезок: команда `\line`

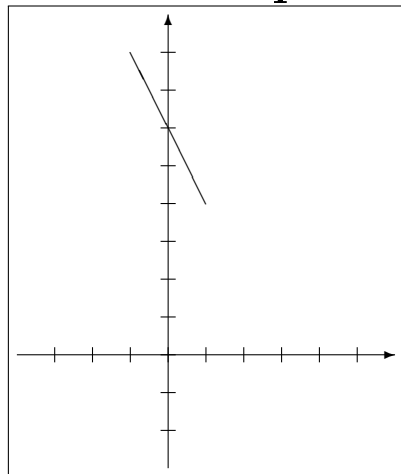


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(005.00,-20.00){\line( 2,-3){10.00}}
\end{picture}
```

`\line( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-6 \leq a \leq 6$, $-6 \leq b \leq 6$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.4. Отрезок: команда `\line`

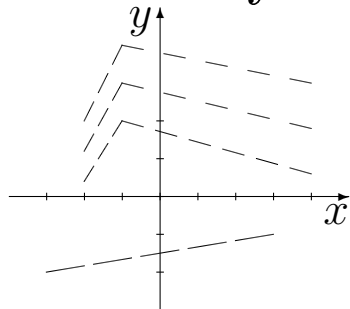


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(005.00,-20.00){\line(-1, 2){10.00}}
\end{picture}
```

`\line( a, b){d}`

Здесь a и b — взаимно простые целые числа или 0, причём $-6 \leq a \leq 6$, $-6 \leq b \leq 6$, d — проекция на ось абсцисс, кроме случая $(a, b) = (0, 1)$, когда d — проекция на ось ординат.

XI.3.5. Пунктирный отрезок: команда `\dashline`



```
\begin{picture}(45,40)(-20,-15)
\dashline{3.0}(-10.0,10.0)(-5.0,20.0)(20.0,15.0)
\dashline[20]{3.0}(-10.0,6.0)(-5.0,15.0)(20.0,9.0)
\dashline[40]{3.0}(-10.0,2.0)(-5.0,10.0)(20.0,3.0)
\dashline[60]{5.0}(-15.0,-10.0)(15.0,-5.0)
\end{picture}
```

`\dashline[stretch]{length}[IDG](x_1,y_1)(x_2,y_2)\dots(x_n,y_n)` —

команда из пакета `epic`. Здесь

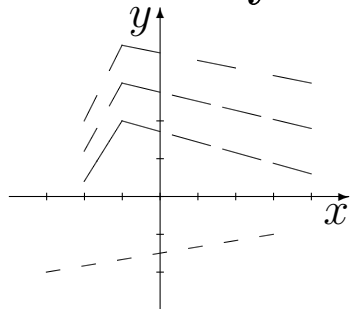
stretch — параметр, определяющий расстояние между пунктирными отрезками, это целое число, большее (-100) ;

length — длина отрезка, составляющего пунктирную линию;

IDG — положительное число, характеризующее просвет между точками в отрезке, составляющем пунктирную линию;

$(x_1, y_1)(x_2, y_2)\dots(x_n, y_n)$ — координаты точек.

XI.3.5. Пунктирный отрезок: команда `\dashline`



```
\begin{picture}(45,40)(-20,-15)
\dashline{5.0}(-10.0,10.0)(-5.0,20.0)(20.0,15.0)
\dashline[20]{5.0}(-10.0,6.0)(-5.0,15.0)(20.0,9.0)
\dashline[40]{5.0}(-10.0,2.0)(-5.0,10.0)(20.0,3.0)
\dashline[-20]{2.0}(-15.0,-10.0)(15.0,-5.0)
\end{picture}
```

`\dashline[stretch]{length}[IDG](x_1,y_1)(x_2,y_2)\dots(x_n,y_n)` —

команда из пакета `epic`. Здесь

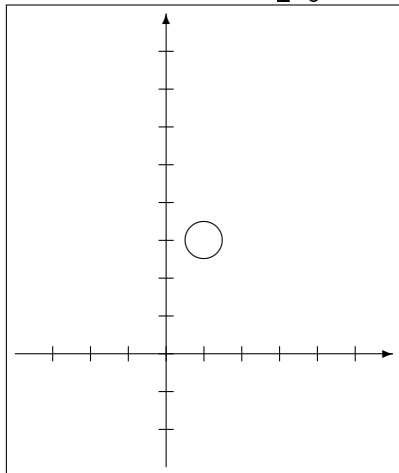
stretch — параметр, определяющий расстояние между пунктирными отрезками, это целое число, большее (-100) ;

length — длина отрезка, составляющего пунктирную линию;

IDG — положительное число, характеризующее просвет между точками в отрезке, составляющем пунктирную линию;

$(x_1,y_1)(x_2,y_2)\dots(x_n,y_n)$ — координаты точек.

XI.3.6. Окружности и круги: команда `\circle`

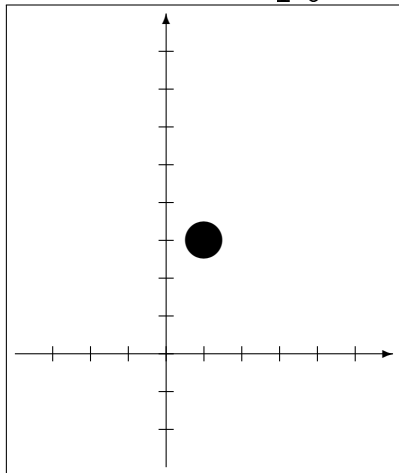


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(005.00,015.00){\circle{05.00}}  
\end{picture}
```

Команда `\circle{D}` изображает окружность диаметра D .

Диапазон значений диаметра ограничен.

XI.3.6. Окружности и круги: команда `\circle`



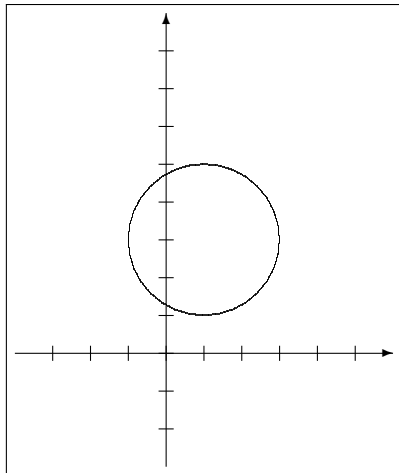
```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(005.00,015.00){\circle*{05.00}}  
\end{picture}
```

Команда `\circle{D}` изображает окружность диаметра D .

Вариант «со звёздочкой».

Диапазон значений диаметра ограничен.

XI.3.7. Окружности: команда `\bigcircle`



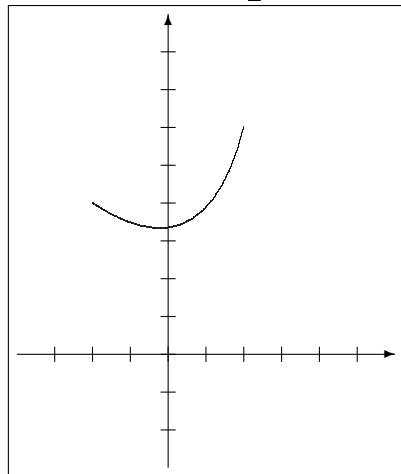
```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \put(005.00,015.00){\bigcircle{20.00}}  
\end{picture}
```

Команда `\bigcircle{D}` из пакета `curves` изображает окружность диаметра D .

D — произвольное положительное число.

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

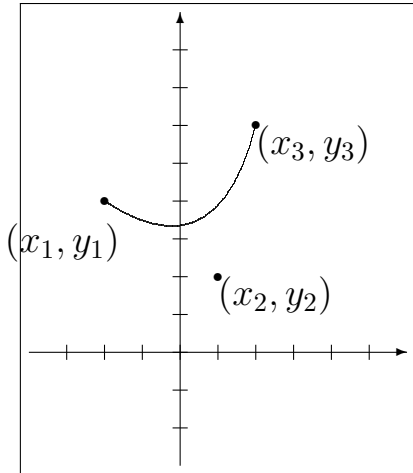
XI.3.8. Кривая Безье: команда `\qbezier`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \qbezier(-10.0,20.0)(05.0,10.0)(10.0,30.0)  
\end{picture}
```

Кривая Безье: `\qbezier`(x_1, y_1)(x_2, y_2)(x_3, y_3)

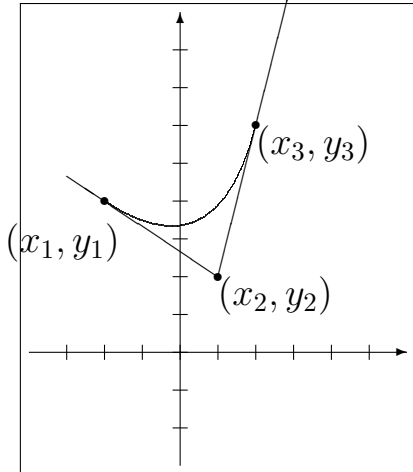
XI.3.8. Кривая Безье: команда `\qBezier`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \qBezier(-10.0,20.0)(05.0,10.0)(10.0,30.0)  
\end{picture}
```

Кривая Безье: `\qBezier` $(x_1, y_1)(x_2, y_2)(x_3, y_3)$

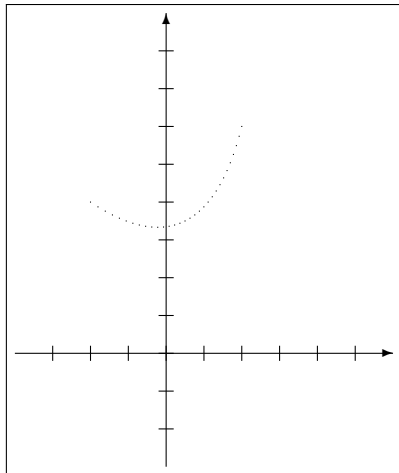
XI.3.8. Кривая Безье: команда `\qbezier`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \qbezier(-10.0,20.0)(05.0,10.0)(10.0,30.0)  
\end{picture}
```

Кривая Безье: `\qbezier`(x_1, y_1)(x_2, y_2)(x_3, y_3)

XI.3.8. Кривая Безье: команда `\qbezier`

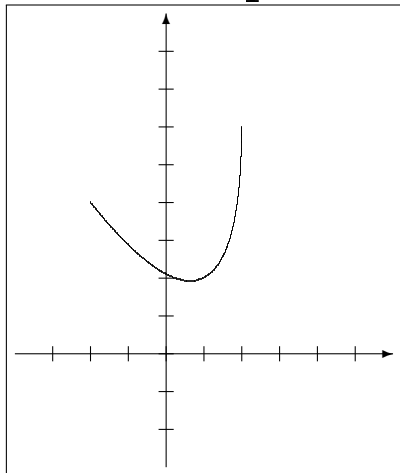


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \qbezier[30](-10.0,20.0)(05.0,10.0)(10.0,30.0)  
\end{picture}
```

Кривая Безье: `\qbezier[n](x_1, y_1)(x_2, y_2)(x_3, y_3)`

Предусмотрен необязательный аргумент — количество точек n .

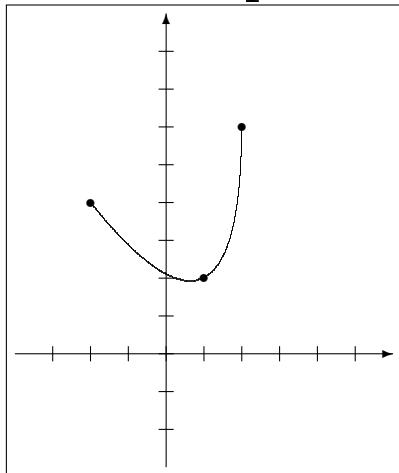
XI.3.9. Кривые: команда `\curve` из пакета `curves`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \curve(-10.0,20.0, 05.0,10.0, 10.0,30.0)  
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

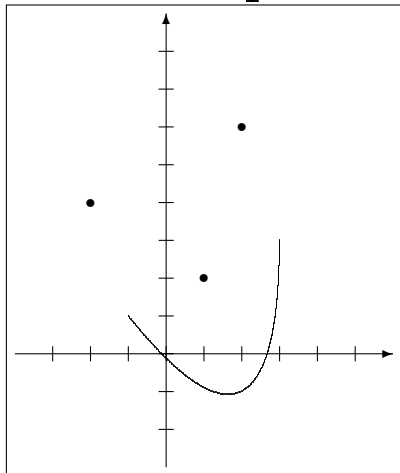
XI.3.9. Кривые: команда `\curve` из пакета `curves`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \curve(-10.0,20.0, 05.0,10.0, 10.0,30.0)  
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

XI.3.9. Кривые: команда `\curve` из пакета `curves`

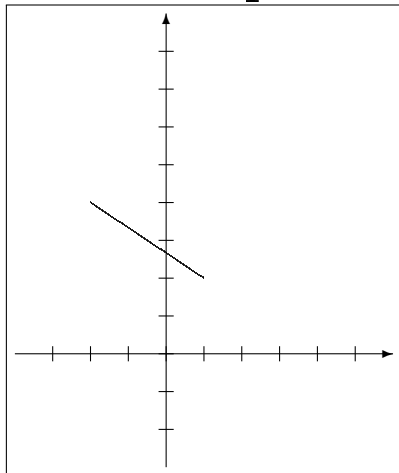


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(05.0,-15.0){\curve(-10.0,20.0,
05.0,10.0, 10.0,30.0)}
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

Команды `\curve` и `\qbezier`, как и другие геометрические объекты, можно подставлять качестве аргумента в команду `\put`

XI.3.9. Кривые: команда `\curve` из пакета `curves`



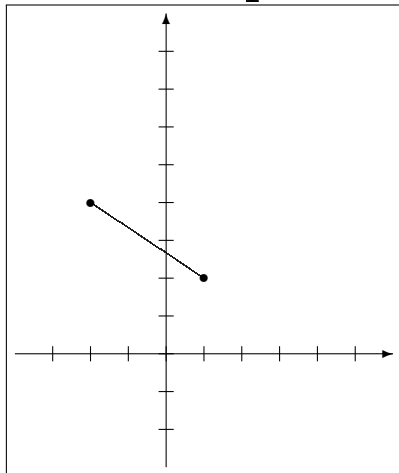
```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \curve(-10.0,20.0, 05.0,10.0)  
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

Если заданы координаты только двух точек, рисует отрезок прямой.

Но команда `\curve` потребляет много ресурсов при компиляции, её использование сильно замедляет компиляцию.

XI.3.9. Кривые: команда `\curve` из пакета `curves`



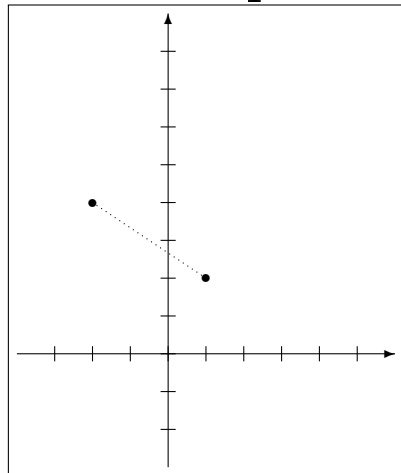
```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \curve(-10.0,20.0, 05.0,10.0)  
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

Если заданы координаты только двух точек, рисует отрезок прямой.

Но команда `\curve` потребляет много ресурсов при компиляции, её использование сильно замедляет компиляцию.

XI.3.9. Кривые: команда `\curve` из пакета `curves`



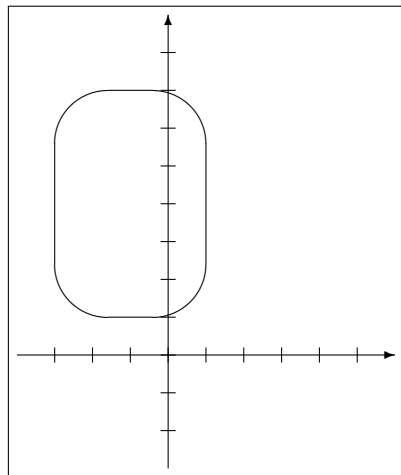
```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \curve[25](-10.0,20.0, 05.0,10.0)  
\end{picture}
```

При использовании `\input texsample.tex` пакет `curves` подключается автоматически.

Если заданы координаты только двух точек, рисует отрезок прямой.

В команде `\curve[25](-10.0,20.0, 05.0,10.0)` предусмотрен необязательный аргумент n — число точек.

XI.3.10. Овал: команда `\oval`

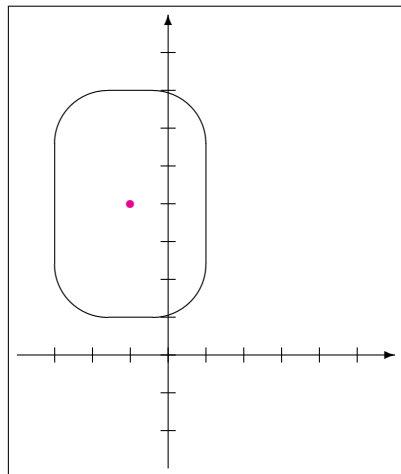


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)}
\end{picture}
```

`\oval(a,b)`

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

XI.3.10. Овал: команда `\oval`

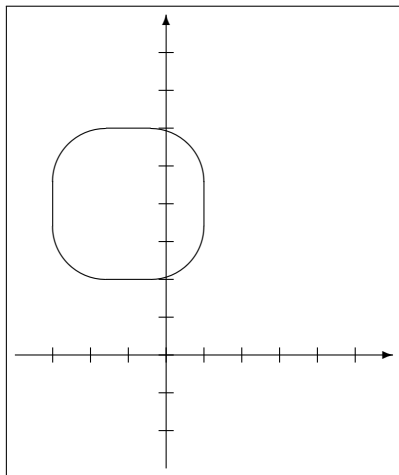


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

В аргументе команды `\put` указываются координаты центра овала.

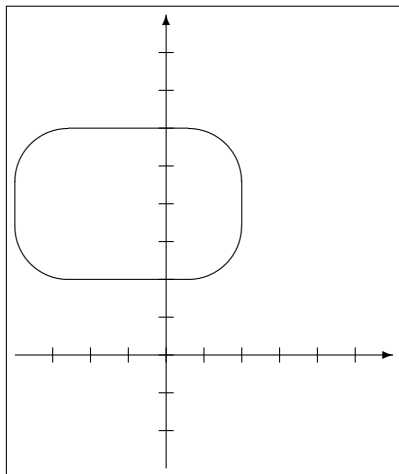
XI.3.10. Овал: команда `\oval`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,20.00)}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

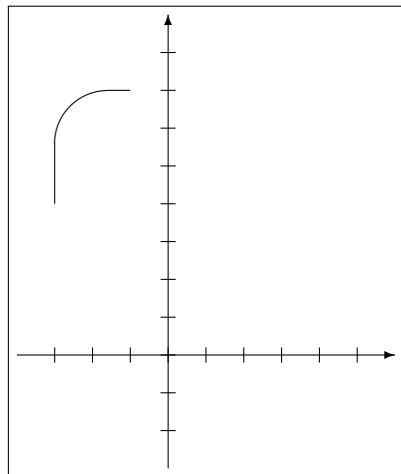
XI.3.10. Овал: команда `\oval`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(30.00,20.00)}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

XI.3.10. Овал: команда `\oval`

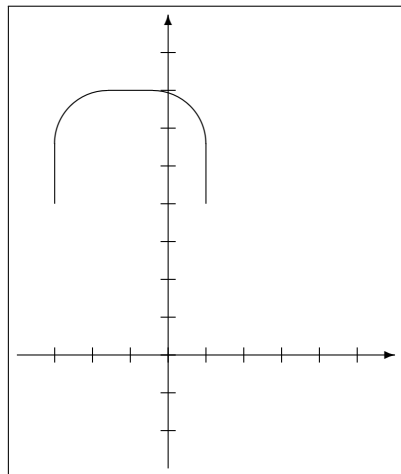


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)[lt]}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

Предусмотрены необязательные параметры `\oval( $a$ , $b$ )[ $pq$ ]`,
где $p \in \{l, c, r\}$ (left, center, right), $q \in \{b, c, t\}$ (bottom, center, top).

XI.3.10. Овал: команда `\oval`

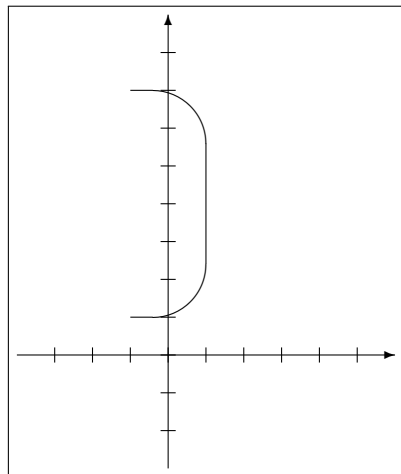


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)[t]}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

Предусмотрены необязательные параметры `\oval( $a, b$ )[ $pq$ ]`,
где $p \in \{l, c, r\}$ (left, center, right), $q \in \{b, c, t\}$ (bottom, center, top).

XI.3.10. Овал: команда `\oval`

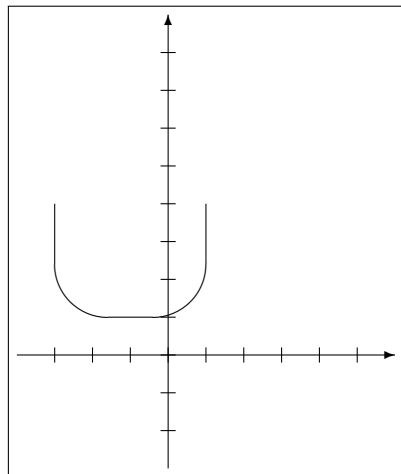


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)[r]}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

Предусмотрены необязательные параметры `\oval( $a$ , $b$ )[ $pq$ ]`,
где $p \in \{l, c, r\}$ (left, center, right), $q \in \{b, c, t\}$ (bottom, center, top).

XI.3.10. Овал: команда `\oval`

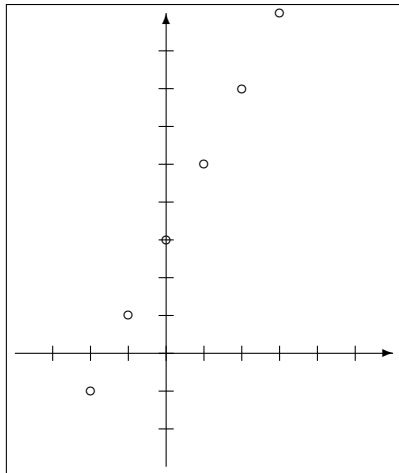


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(-05.00,020.00){\oval(20.00,30.00)[b]}
\end{picture}
```

a — ширина овала (прямоугольника со скруглёнными углами),
 b — высота овала.

Предусмотрены необязательные параметры `\oval( $a, b$ )[ $pq$ ]`,
где $p \in \{l, c, r\}$ (left, center, right), $q \in \{b, c, t\}$ (bottom, center, top).

XI.3.11. Кратное размещение объекта: `\multiput`

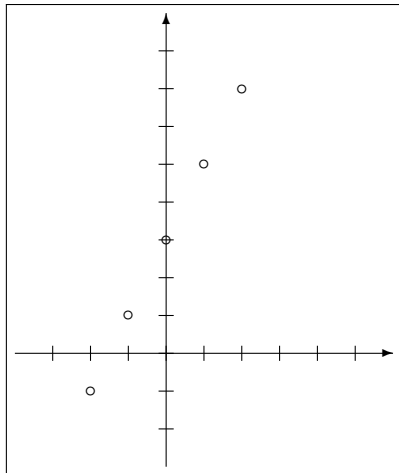


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \multiput(-10.0,-05.0)(5.0,10.0){6}{\circle{1}}
\end{picture}
```

`\multiput` $(x, y) (\Delta x, \Delta y) \{n\} \{ \dots \}$

Здесь (x, y) — координаты начальной точки,
 $(\Delta x, \Delta y)$ — шаг смещения по осям,
 n — количество повторений.

XI.3.11. Кратное размещение объекта: `\multiput`

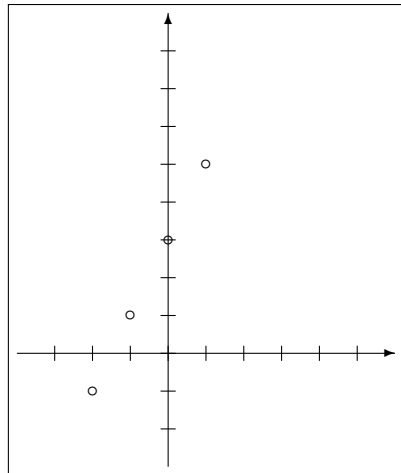


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \multiput(-10.0,-05.0)(5.0,10.0){5}{\circle{1}}  
\end{picture}
```

`\multiput` $(x, y) (\Delta x, \Delta y) \{n\} \{ \dots \}$

Здесь (x, y) — координаты начальной точки,
 $(\Delta x, \Delta y)$ — шаг смещения по осям,
 n — количество повторений.

XI.3.11. Кратное размещение объекта: `\multiput`

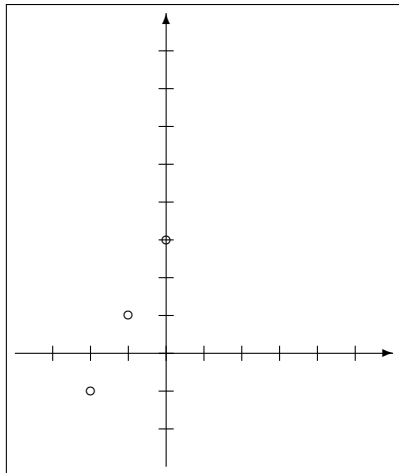


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \multiput(-10.0,-05.0)(5.0,10.0){4}{\circle{1}}  
\end{picture}
```

`\multiput` $(x, y)(\Delta x, \Delta y)\{n\}\{\dots\}$

Здесь (x, y) — координаты начальной точки,
 $(\Delta x, \Delta y)$ — шаг смещения по осям,
 n — количество повторений.

XI.3.11. Кратное размещение объекта: `\multiput`

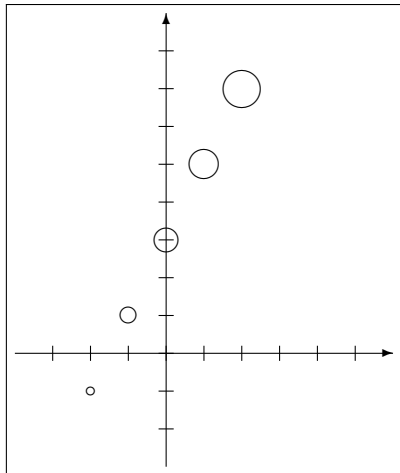


```
\begin{picture}(50.0,60.0)(-20.0,-15.0)  
  \multiput(-10.0,-05.0)(5.0,10.0){3}{\circle{1}}  
\end{picture}
```

`\multiput` $(x, y) (\Delta x, \Delta y) \{n\} \{ \dots \}$

Здесь (x, y) — координаты начальной точки,
 $(\Delta x, \Delta y)$ — шаг смещения по осям,
 n — количество повторений.

XI.3.11. Кратное размещение объекта: `\multiput`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \multiput(-10.0,-05.0)(5.0,10.0){5}{\ldots}
\end{picture}
```

`\multiput` $(x, y) (\Delta x, \Delta y) \{n\} \{ \dots \}$

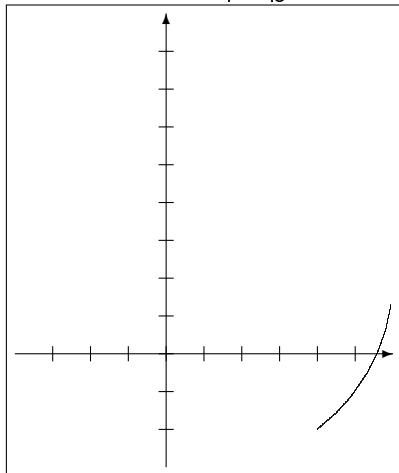
```
\setcounter{CountRisA}{1}
```

```
\multiput(-10.00,-05.00)(005.00,010.00){5}{%
```

```
\circle{\arabic{CountRisA}}\refstepcounter{CountRisA}}
```

Таким образом можно изменять параметры повторяемых объектов.

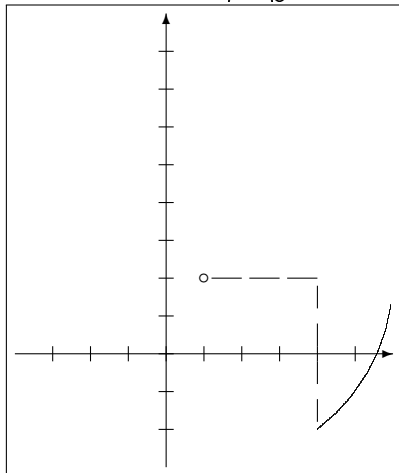
XI.3.12. Дуга окружности: команда `\arc`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(05.0,10.0){\arc(15.0,-20.0){45}}
\end{picture}
```

Команда `\arc( $\Delta x$ ,  $\Delta y$ ){ $\alpha$ }` изображает дугу окружности от точки с координатами $(\Delta x, \Delta y)$ (это координаты смещения точки относительно центра окружности) с углом α , измеряемым в градусах.

XI.3.12. Дуга окружности: команда `\arc`



```
\begin{picture}(50.0,60.0)(-20.0,-15.0)
  \put(05.0,10.0){\arc(15.0,-20.0){45}}
\end{picture}
```

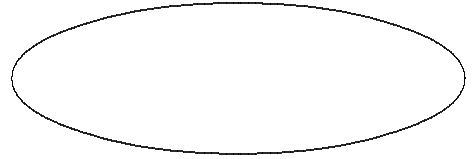
Команда `\arc( $\Delta x$ ,  $\Delta y$ ){ $\alpha$ }` изображает дугу окружности от точки с координатами $(\Delta x, \Delta y)$ (это координаты смещения точки относительно центра окружности) с углом α , измеряемым в градусах.

XI.3.13. Эллипс: команда `\ellipsA`

Если вы используете `texsample.tex`, то вам доступна команда `\ellipsA` изображения эллипса.

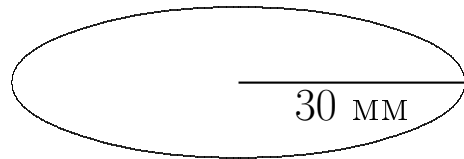
XI.3.13. Эллипс: команда `\ellipsA`

Первые два аргумента обозначают длины полуосей эллипса.



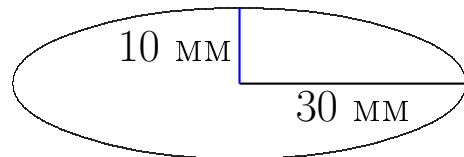
XI.3.13. Эллипс: команда `\ellipsA`

Первые два аргумента обозначают длины полуосей эллипса.



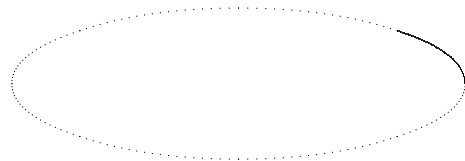
XI.3.13. Эллипс: команда `\ellipsA`

Первые два аргумента обозначают длины полуосей эллипса.



XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.



Результат применения

```
\ellipsA{3000}{1000}{4}{2}{2}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.



Результат применения

```
\ellipsA{3000}{1000}{3}{2}{2}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.

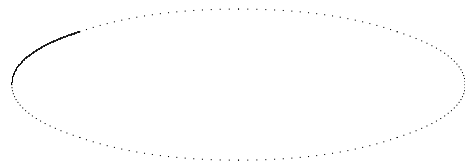


Результат применения

```
\ellipsA{3000}{1000}{2}{3}{2}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.

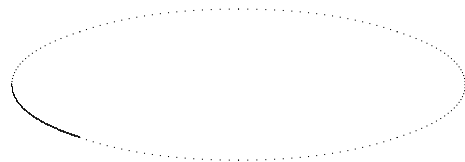


Результат применения

```
\ellipsA{3000}{1000}{2}{4}{2}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.

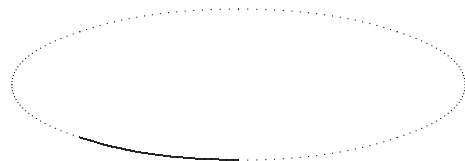


Результат применения

```
\ellipsA{3000}{1000}{2}{2}{4}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.

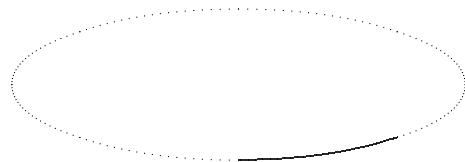


Результат применения

```
\ellipsA{3000}{1000}{2}{2}{3}{2}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.

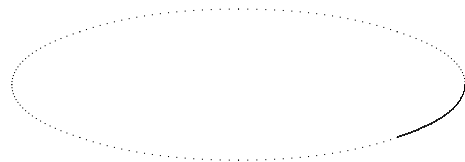


Результат применения

```
\ellipsA{3000}{1000}{2}{2}{2}{3}{20}{magenta}
```

XI.3.13. Эллипс: команда `\ellipsA`

В этой команде эллипс рассматривается как фигура из 8 частей.



Результат применения

```
\ellipsA{3000}{1000}{2}{2}{2}{4}{20}{magenta}
```


XI.3.13. Эллипс: команда `\ellipsA`

При данных значениях параметров $xyzt$ в команде `\ellipsA{1800}{400}{x}{y}{z}{t}{10}{magenta}` получаем следующие результаты:

0777	7077	7707	7770
1777	7177	7717	7771
2777	7277	7727	7772
3777	7377	7737	7773
4777	7477	7747	7774
5777	7577	7757	7775
6777	7677	7767	7776
7777	7777	7777	7777

XI.3.13. Эллипс: команда `\ellipsA`

При данных значениях параметров $xyzt$ в команде `\ellipsA{1800}{400}{x}{y}{z}{t}{10}{magenta}` получаем следующие результаты:

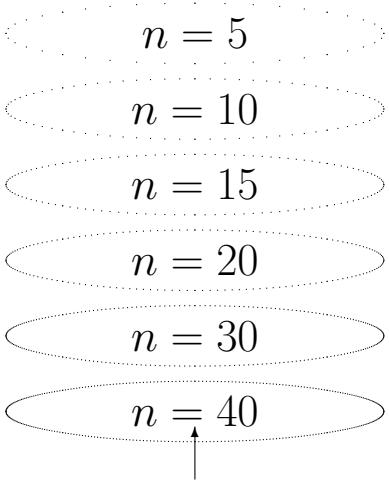
0222	2022	2202	2220
1222	2122	2212	2221
2222	2222	2222	2222
3222	2322	2232	2223
4222	2222	2222	2224
5222	2522	2252	2225
6222	2622	2262	2226
7222	2722	2272	2227

XI.3.13. Эллипс: команда `\ellipsA`

Параметр n в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{magenta}`

обозначает количество точек в каждой восьмой части эллипса.


$$n = 5$$

$$n = 10$$

$$n = 15$$

$$n = 20$$

$$n = 30$$

$$n = 40$$

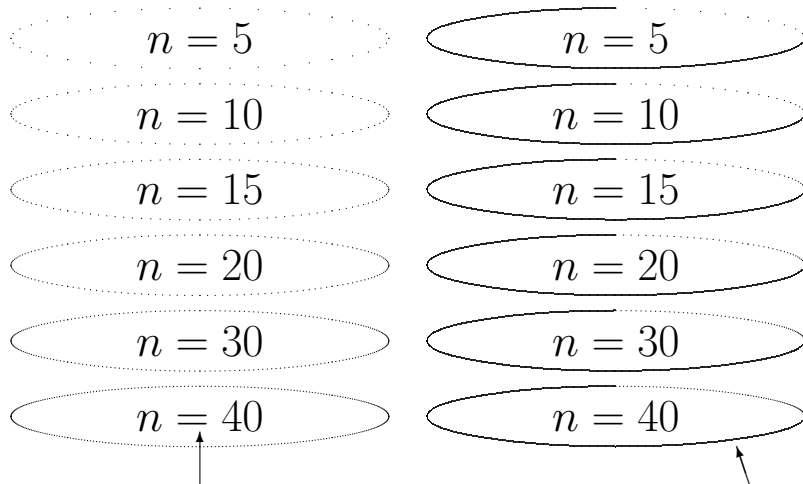
`\ellipsA{2500}{400}{2}{2}{2}{2}{n}{magenta}`

XI.3.13. Эллипс: команда `\ellipsA`

Параметр n в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{magenta}`

обозначает количество точек в каждой восьмой части эллипса.



`\ellipsA{2500}{400}{2}{2}{2}{2}{n}{magenta}`

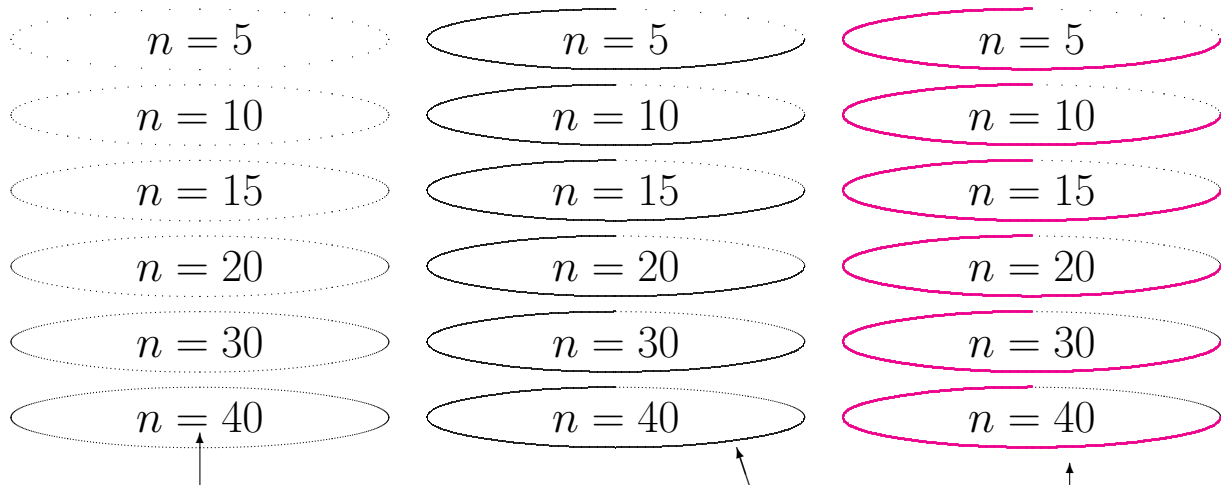
`\ellipsA{2500}{400}{2}{0}{0}{0}{n}{magenta}`

XI.3.13. Эллипс: команда `\ellipsA`

Параметр n в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{magenta}`

обозначает количество точек в каждой восьмой части эллипса.



`\ellipsA{2500}{400}{2}{2}{2}{2}{n}{magenta}`

`\ellipsA{2500}{400}{2}{0}{0}{0}{n}{magenta}`

`\ellipsA{2500}{400}{2}{1}{1}{1}{n}{magenta}`

XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `black`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{black}`

XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

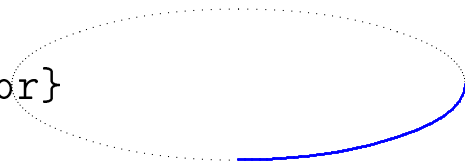
обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `blue`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{blue}`



XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `cyan`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{cyan}`

XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

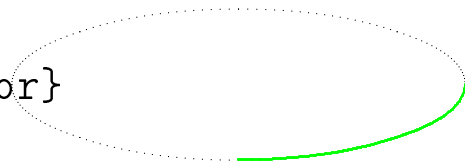
обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `green`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{green}`



XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

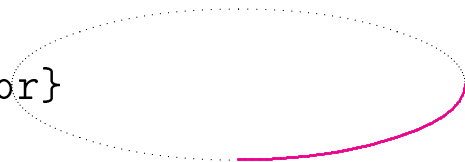
обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `magenta`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{magenta}`



XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

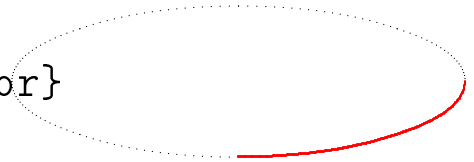
обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `red`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{red}`



XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

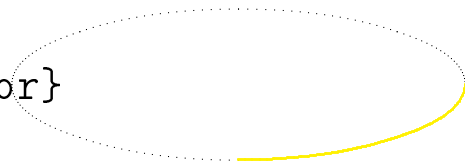
обозначает цвет восьмой части эллипса

при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `yellow`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{yellow}`



XI.3.13. Эллипс: команда `\ellipsA`

Параметр `color` в команде

`\ellipsA{2500}{400}{x}{y}{z}{t}{n}{color}`

обозначает цвет восьмой части эллипса

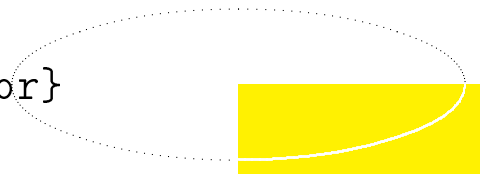
при $x = 1$, $y = 1$, $z = 1$ и/или $t = 1$.

Цвет может принимать значения `black`, `blue`, `cyan`, `green`, `magenta`, `red`, `yellow`, `white`.

В данном случае использован цвет `white`, результат применения

`\ellipsA{3000}{1000}{2}{2}{2}{1}{20}{white}`

Здесь участок эллипса, окрашенный в белый цвет, показан на жёлтом фоне (иначе он был бы невидим на белом фоне).



XI.3.14. Изображение прямоугольной декартовой системы координат: команда `\ComSystCoord`

Систему координат можно изобразить с помощью команд `vector`, `multiput`, `line`, `makebox` и др. Но проще воспользоваться введённой в `texsample.tex` командой `\ComSystCoord`.

Эта команда используется внутри окружения `\picture` или команд, в которые встроено это окружение (например, `\TextPict` и `\PictText`).

XI.3.14. Изображение прямоугольной декартовой системы координат: команда `\ComSystCoord`

В `texsample.tex` введена команда

`\ComSystCoord{a}{b}{m_-}{m_+}{n_-}{n_+}{x}{y}{K}`,

где a — расстояние между метками на оси абсцисс (измеряемое в `\unitlength`, т.е. a — это безразмерное число);

b — расстояние между метками на оси ординат (тоже измеряемое в `\unitlength`);

m_- — число отметок на оси абсцисс слева от начала координат;

m_+ — число отметок на оси абсцисс справа от начала координат;

n_- — число отметок на оси ординат ниже начала координат;

n_+ — число отметок на оси ординат выше начала координат;

x — обозначение оси абсцисс; y — обозначение оси ординат;

K — система команд рисования фигур в данной системе координат.

Рассмотрим пример?

XI.3.15. Штриховка области

К сожалению, среди скромных возможностей рисования в $\text{\LaTeX}$ нет стандартной команды для выполнения штриховки.

XI.3.15. Штриховка области

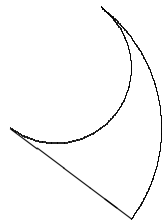
К сожалению, среди скромных возможностей рисования в $\text{\LaTeX}$ нет стандартной команды для выполнения штриховки.

Проблему можно решить с помощью использования `mfpic`.

Но если этот вариант по каким-то соображениям не устраивает, можно применить следующий способ.

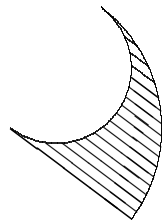
ХІ.3.15. Штриховка области

Допустим, необходимо заштриховать данную область.



ХІ.3.15. Штриховка области

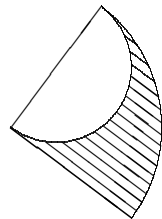
Допустим, необходимо заштриховать данную область.



ХІ.3.15. Штриховка области

Допустим, необходимо заштриховать данную область.

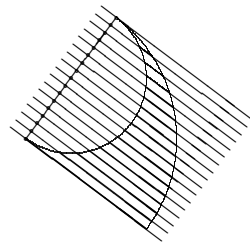
Выберем наклон линий и



ХІ.3.15. Штриховка области

Допустим, необходимо заштриховать данную область.

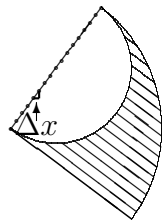
Выберем наклон линий и
прямую, проходящую перпендикулярно им.



ХІ.3.15. Штриховка области

Допустим, необходимо заштриховать данную область.

Выберем наклон линий и
прямую, проходящую перпендикулярно им.

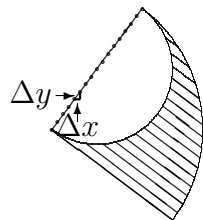


Введём обозначения, характеризующие шаг штриховки: Δx и

ХІ.3.15. Штриховка области

Допустим, необходимо заштриховать данную область.

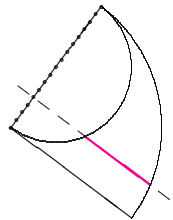
Выберем наклон линий и прямую, проходящую перпендикулярно им.



Введём обозначения, характеризующие шаг штриховки: Δx и Δy .

ХІ.3.15. Штриховка области

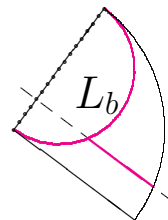
Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .



Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

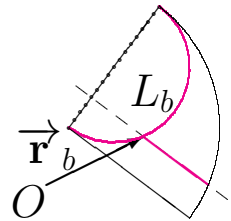
Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .



Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

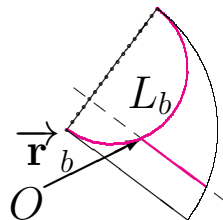


Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

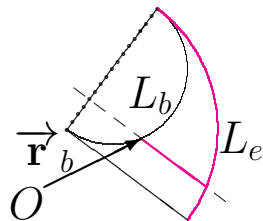


Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

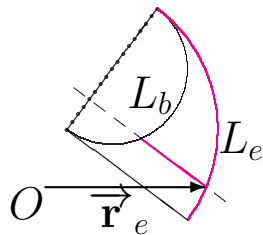


Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$



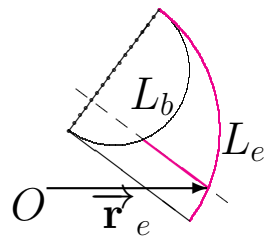
Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



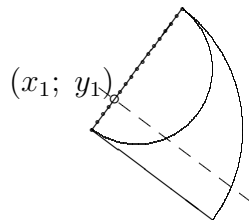
Чтобы описать часть прямой, лежащей внутри нашей области, опишем её границы.

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



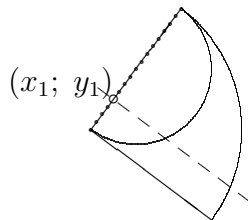
Параметрические уравнения «пунктирной» прямой:

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

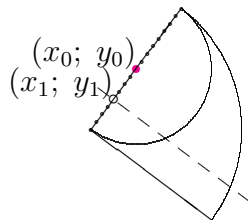
$$\begin{cases} x = x_1 - t\Delta y = \\ y = y_1 + t\Delta x = \end{cases}$$

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

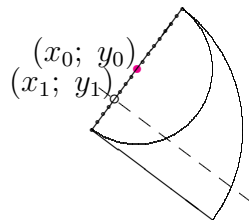
$$\begin{cases} x = x_1 - t\Delta y = \\ y = y_1 + t\Delta x = \end{cases}$$

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

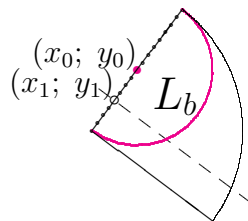
$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

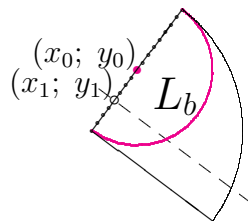
Найдём пересечение с границей L_b :

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

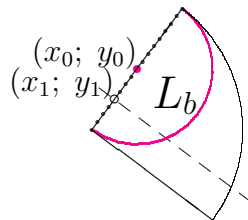
$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow$$

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow$$

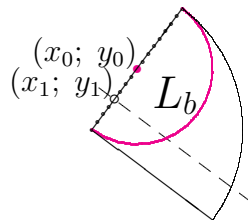
Для того, чтобы «избавиться от t » просуммируем первое уравнение, умноженное на Δx и второе уравнение, умноженное на Δy .

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow \begin{cases} \Delta x x_0 + \Delta y y_0 + n(\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n). \end{cases}$$

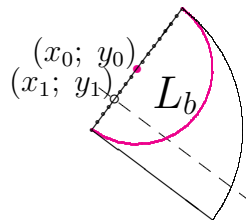
Для того, чтобы «избавиться от t » просуммируем первое уравнение, умноженное на Δx и второе уравнение, умноженное на Δy .

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$



Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow \begin{cases} \Delta x x_0 + \Delta y y_0 + n(\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n). \end{cases}$$

Из последнего уравнения найдём α_n и получим искомые координаты точки пересечения «пунктирной прямой» с границей L_b .

ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \end{cases}$$

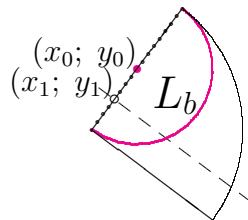
Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow \begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n). \end{cases}$$

Из последнего уравнения найдём α_n и получим искомые координаты точки пересечения «пунктирной прямой» с границей L_b .



ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \end{cases}$$

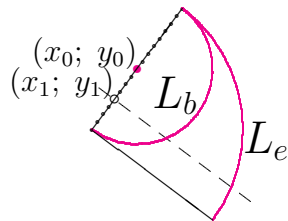
Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow \begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n). \end{cases}$$

Аналогично найдём значение параметра β_n для точки пересечения с границей L_e .



ХІ.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

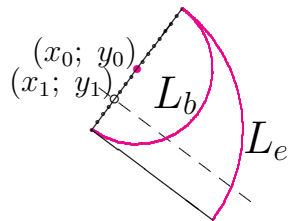
Параметрические уравнения «пунктирной» прямой:

$$\begin{cases} x = x_1 - t\Delta y = x_0 + n\Delta x - t\Delta y, \\ y = y_1 + t\Delta x = y_0 + n\Delta y + t\Delta x. \end{cases}$$

Найдём пересечение с границей L_b :

$$\begin{cases} x_0 + n\Delta x - t\Delta y = x_b(\alpha_n), \\ y_0 + n\Delta y + t\Delta x = y_b(\alpha_n) \end{cases} \Rightarrow \begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n). \end{cases}$$

Аналогично найдём значение параметра β_n для точки пересечения с границей L_e .



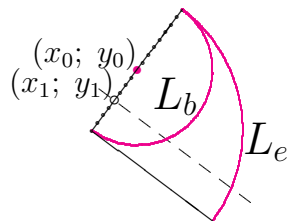
XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

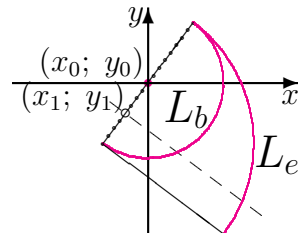
$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

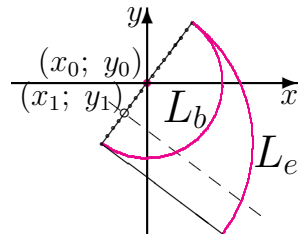
$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$



Из этих уравнений находим α_n и β_n .

Подставим эти выражения в систему уравнений для нахождения значения параметров:

α_n — для точки пересечения линии L_b с «пунктирной прямой» и β_n — для точки пересечения линии L_e с «пунктирной прямой».

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{r}_b(\alpha) = x_b(\alpha) \vec{i} + y_b(\alpha) \vec{j},$$

$$L_e : \vec{r}_e(\beta) = x_e(\beta) \vec{i} + y_e(\beta) \vec{j}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

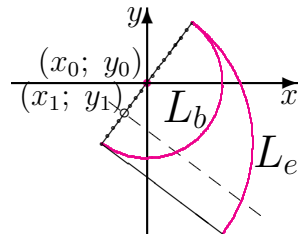
$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$

$$\begin{cases} 25n/16 = 7, 5 \cos \alpha_n + 10 \sin \alpha_n, \\ 25n/16 = -12, 5 + 15 \cos \beta_n + 20 \sin \beta_n, \end{cases} \Rightarrow$$

Подставим эти выражения в систему уравнений для нахождения значения параметров:

α_n — для точки пересечения линии L_b с «пунктирной прямой» и

β_n — для точки пересечения линии L_e с «пунктирной прямой».



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

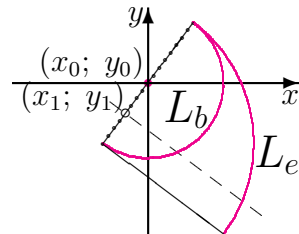
$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$

$$\begin{cases} 25n/16 = 7,5 \cos \alpha_n + 10 \sin \alpha_n, \\ 25n/16 = -12,5 + 15 \cos \beta_n + 20 \sin \beta_n, \end{cases} \Rightarrow \begin{cases} n/8 = 0,6 \cos \alpha_n + 0,8 \sin \alpha_n, \\ (n+8)/16 = 0,6 \cos \beta_n + 0,8 \sin \beta_n, \end{cases}$$



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

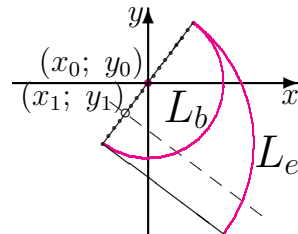
$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$

$$\begin{cases} 25n/16 = 7,5 \cos \alpha_n + 10 \sin \alpha_n, \\ 25n/16 = -12,5 + 15 \cos \beta_n + 20 \sin \beta_n, \end{cases} \Rightarrow \begin{cases} n/8 = 0,6 \cos \alpha_n + 0,8 \sin \alpha_n, \\ (n+8)/16 = 0,6 \cos \beta_n + 0,8 \sin \beta_n, \end{cases}$$

$$\begin{cases} x_b(\alpha_n) = 10 \cos (\pm \arccos(n/8) + \arccos 0,6), \\ y_b(\alpha_n) = 10 \sin (\pm \arccos(n/8) + \arccos 0,6), \end{cases}$$



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{r}_b(\alpha) = x_b(\alpha) \vec{i} + y_b(\alpha) \vec{j},$$

$$L_e : \vec{r}_e(\beta) = x_e(\beta) \vec{i} + y_e(\beta) \vec{j}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

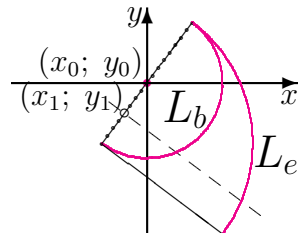
Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \quad \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \quad \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$

$$\begin{cases} 25n/16 = 7,5 \cos \alpha_n + 10 \sin \alpha_n, \\ 25n/16 = -12,5 + 15 \cos \beta_n + 20 \sin \beta_n, \end{cases} \Rightarrow \begin{cases} n/8 = 0,6 \cos \alpha_n + 0,8 \sin \alpha_n, \\ (n+8)/16 = 0,6 \cos \beta_n + 0,8 \sin \beta_n, \end{cases}$$

$$\begin{cases} x_b(\alpha_n) = 10 \cos (\pm \arccos(n/8) + \arccos 0,6), \\ y_b(\alpha_n) = 10 \sin (\pm \arccos(n/8) + \arccos 0,6), \end{cases}$$

$$\begin{cases} x_e(\beta_n) = -6 + 20 \cos (\pm \arccos((n+8)/16) + \arccos 0,6), \\ y_e(\beta_n) = -8 + 20 \sin (\pm \arccos((n+8)/16) + \arccos 0,6). \end{cases}$$



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

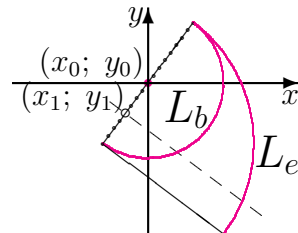
Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, & \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} & \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases} \end{cases}$$

$$\begin{cases} x_b(\alpha_n) = 10 \cos (\pm \arccos(n/8) + \arccos 0, 6), \\ y_b(\alpha_n) = 10 \sin (\pm \arccos(n/8) + \arccos 0, 6), \end{cases}$$

$$\begin{cases} x_e(\beta_n) = -6 + 20 \cos (\pm \arccos((n+8)/16) + \arccos 0, 6), \\ y_e(\beta_n) = -8 + 20 \sin (\pm \arccos((n+8)/16) + \arccos 0, 6). \end{cases}$$

Полученные выражения надо подставить в лист «Исходные данные» в файле `shading.xls` в папке `ShablonPresentations`



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

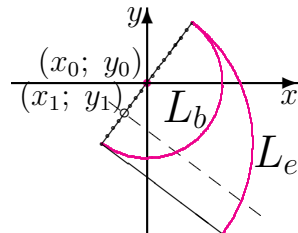
Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, & \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} & \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases} \end{cases}$$

$$\begin{cases} x_b(\alpha_n) = 10 \cos (\pm \arccos(n/8) + \arccos 0, 6), \\ y_b(\alpha_n) = 10 \sin (\pm \arccos(n/8) + \arccos 0, 6), \end{cases}$$

$$\begin{cases} x_e(\beta_n) = -6 + 20 \cos (\pm \arccos((n+8)/16) + \arccos 0, 6), \\ y_e(\beta_n) = -8 + 20 \sin (\pm \arccos((n+8)/16) + \arccos 0, 6). \end{cases}$$

После этого можно будет скопировать нужный фрагмент кода из листа «Итоговый TeXовский код».



Из этих уравнений находим α_n и β_n .

XI.3.15. Штриховка области

Параметры, характеризующие шаг штриховки, обозначили через Δx и Δy .

$$L_b : \vec{\mathbf{r}}_b(\alpha) = x_b(\alpha) \vec{\mathbf{i}} + y_b(\alpha) \vec{\mathbf{j}},$$

$$L_e : \vec{\mathbf{r}}_e(\beta) = x_e(\beta) \vec{\mathbf{i}} + y_e(\beta) \vec{\mathbf{j}}.$$

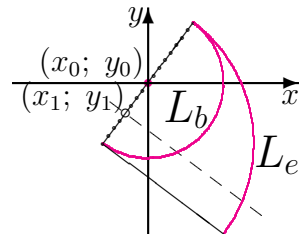
$$\begin{cases} \Delta x x_0 + \Delta y y_0 + n (\Delta x^2 + \Delta y^2) = \\ = \Delta x x_b(\alpha_n) + \Delta y y_b(\alpha_n) = \Delta x x_e(\beta_n) + \Delta y y_e(\beta_n). \end{cases}$$

Например, в изображённой системе координат имеем:

$$\begin{cases} x_b(\alpha) = 10 \cos \alpha, \\ y_b(\alpha) = 10 \sin \alpha, \end{cases} \begin{cases} x_e(\beta) = -6 + 20 \cos \beta, \\ y_e(\beta) = -8 + 20 \sin \beta, \end{cases} \begin{cases} (x_0, y_0) = (0, 0). \\ (\Delta x, \Delta y) = (0, 75; 1). \end{cases}$$

$$\begin{cases} x_b(\alpha_n) = 10 \cos (\pm \arccos(n/8) + \arccos 0, 6), \\ y_b(\alpha_n) = 10 \sin (\pm \arccos(n/8) + \arccos 0, 6), \end{cases}$$

$$\begin{cases} x_e(\beta_n) = -6 + 20 \cos (\pm \arccos((n+8)/16) + \arccos 0, 6), \\ y_e(\beta_n) = -8 + 20 \sin (\pm \arccos((n+8)/16) + \arccos 0, 6). \end{cases}$$



Из этих уравнений находим α_n и β_n .

Компилировать полученный код с помощью `pdflatex` или `latex` можно будет после естественной корректировки (замены десятичной запятой на точку). **Рассмотрим пример?**

XI.4. Импорт рисунков

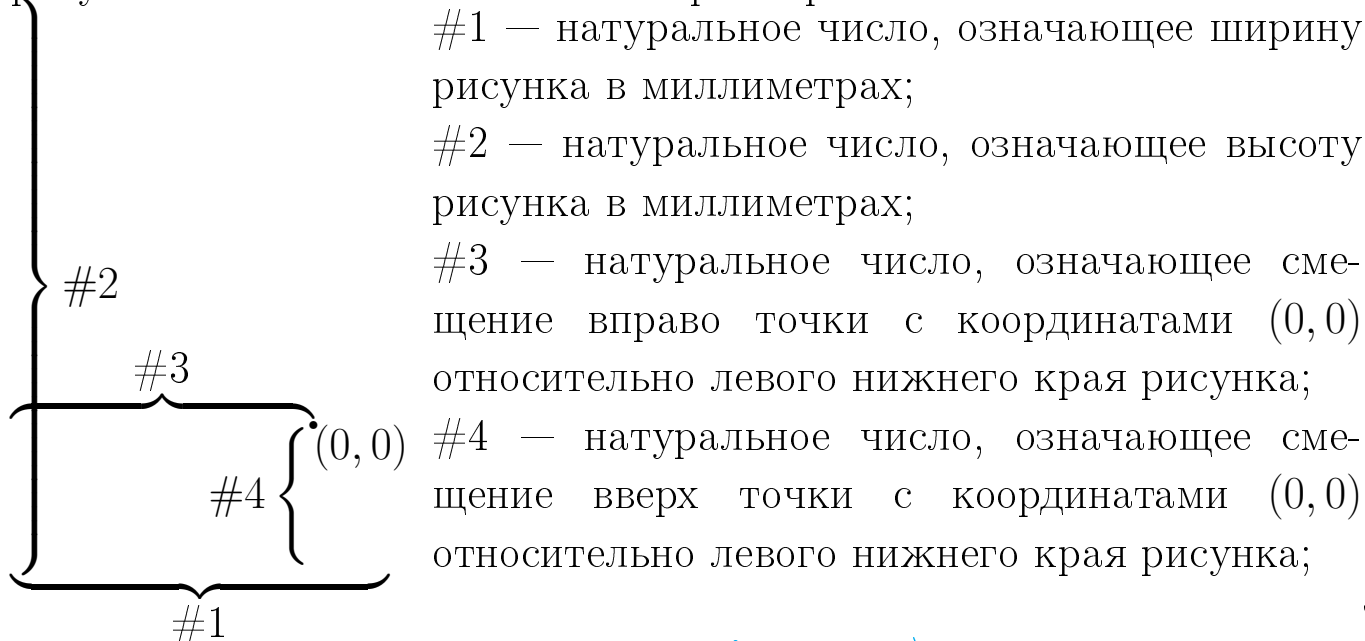
Импорт рисунков можно осуществить с помощью различных команд. Один из вариантов — применение команды `\includegraphics`

Преобразование файлов `*.eps` в формат `*.pdf` можно осуществить с помощью программы `pstopdf.exe`: в командной строке надо набрать

`pstopdf.exe имя_файла` или `pstopdf.exe имя_файла.ps`

XI.5. Размещение текста слева (справа) от рис.

Команды `\PictText` и `\TextPict` вывода текста справа (слева) от рисунка имеют 9 обязательных параметров:

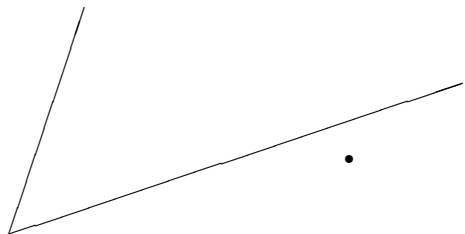


— система команд для окружения `picture`: `\put...` и др.;

#6 — текст, печатаемый справа от рисунка, #7, #8, #9 — свободны.

XI.5.1. Пример применения команды `\PictText`

```
\PictText{60}{30}{45}{10}{  
  \put(-45.00,-10.00){\line( 3, 1){060.00}}  
  \put(-45.00,-10.00){\line( 1, 3){010.00}}  
  \put(000.00,000.00){\circle*{1}}  
}{Текст может содержать несколько абзацев.\par  
  Используются команды  
  \symbol{92}put, \symbol{92}line, \symbol{92}circle,  
  \symbol{92}symbol.  
}{}}{}}
```



Текст может содержать несколько абзацев.

Используются команды

`\put`, `\line`, `\circle`, `\symbol`.

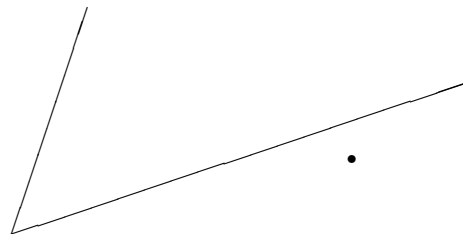
XI.5.2. Пример применения команды `\TextPict`

```
\TextPict{60}{30}{45}{10}{  
  \put(-45.00,-10.00){\line( 3, 1){060.00}}  
  \put(-45.00,-10.00){\line( 1, 3){010.00}}  
  \put(000.00,000.00){\circle*{1}}  
}{Текст может содержать несколько абзацев.\par  
  Использованы команды  
  \symbol{92}put, \symbol{92}line, \symbol{92}circle,  
  \symbol{92}symbol.  
}{}}{}}
```

Текст может содержать несколько абзацев.

Использованы команды

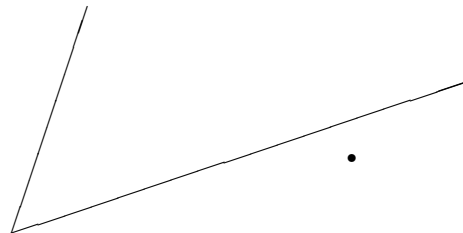
`\put`, `\line`, `\circle`, `\symbol`.



XI.5.3. Пример применения команды `\TextPict`

```
\TextPict{60}{30}{45}{10}{  
  \put(-45.00,-10.00){\line( 3, 1){060.00}}  
  \put(-45.00,-10.00){\line( 1, 3){010.00}}  
  \put(000.00,000.00){\circle*{1}}  
}{Текст может содержать несколько абзацев.\par  
  Используются команды  
  \symbol{92}put, \symbol{92}line, \symbol{92}circle,  
  \symbol{92}symbol.  
}{}}{}}
```

Рассмотрим пример?



XI.6. Создание графиков

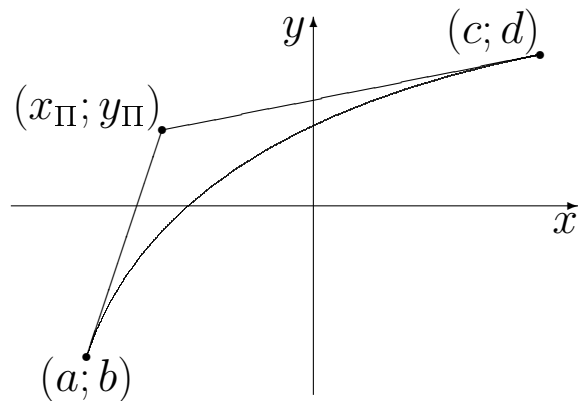
Графики можно создавать с помощью METAFONT и пакета mfpic. Документация по mfpic находится в

Program Files\MikTeX 2.7\doc\latex\mfpic\mfpguide.pdf

Здесь мы рассмотрим «рукодельные» способы создания графиков.

XI.6.1. Построение гладкой кривой по частям с помощью `\qbezier`

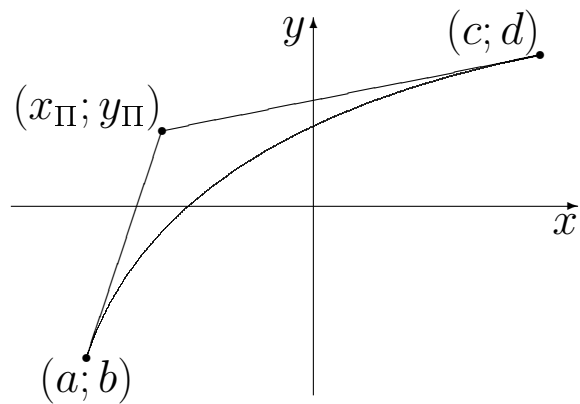
$$\begin{cases} y(a) = b, \\ y(c) = d, \\ y'(a) = k_1, \\ y'(c) = k_2 \end{cases} \Rightarrow \begin{cases} x_{\Pi} = \frac{ak_1 - ck_2 - b + d}{k_1 - k_2}, \\ y_{\Pi} = \frac{(a - c)k_1k_2 + dk_1 - bk_2}{k_1 - k_2}. \end{cases}$$



Значения a, b, c, d и x_{Π}, y_{Π} можно получить с помощью файла `RisByQbezier.xls` из папки `ShablonPresentations`.

XI.6.1. Построение гладкой кривой по частям с помощью `\qbezier`

$$\begin{cases} y(a) = b, \\ y(c) = d, \\ y'(a) = k_1, \\ y'(c) = k_2 \end{cases} \Rightarrow \begin{cases} x_{\Pi} = \frac{ak_1 - ck_2 - b + d}{k_1 - k_2}, \\ y_{\Pi} = \frac{(a - c)k_1k_2 + dk_1 - bk_2}{k_1 - k_2}. \end{cases}$$



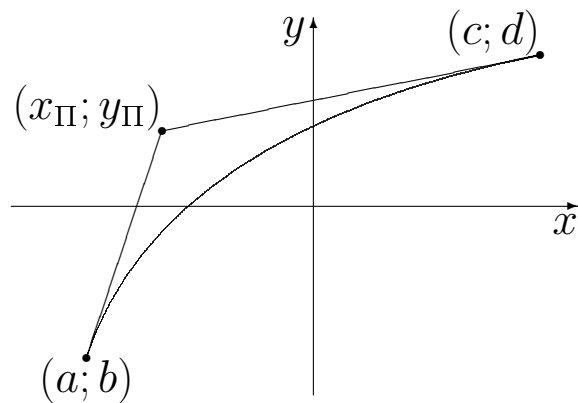
Значения a, b, c, d и x_{Π}, y_{Π} можно получить с помощью файла `RisByQbezier.xls` из папки `ShablonPresentations`.

На листе «Исходные» заполнять можно только клетки жёлтого цвета, а в клетки зеленого цвета заносятся

формулы для вычисления значения функции с аргументом из ячейки $B2, B4, \dots$ и её производной.

XI.6.1. Построение гладкой кривой по частям с помощью `\qbezier`

$$\begin{cases} y(a) = b, \\ y(c) = d, \\ y'(a) = k_1, \\ y'(c) = k_2 \end{cases} \Rightarrow \begin{cases} x_{\Pi} = \frac{ak_1 - ck_2 - b + d}{k_1 - k_2}, \\ y_{\Pi} = \frac{(a - c)k_1k_2 + dk_1 - bk_2}{k_1 - k_2}. \end{cases}$$

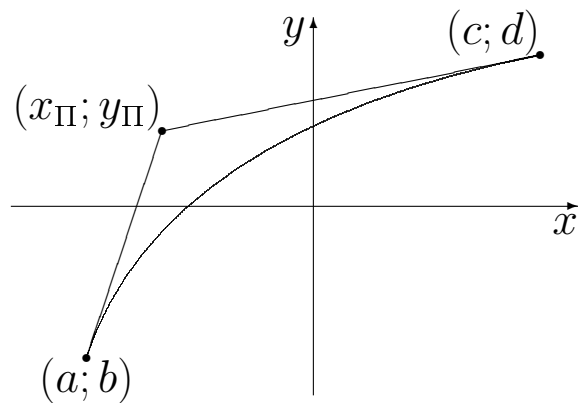


Проще всего после заполнения четвёртой строки скопировать пару строк с номерами 4, 5 (с помощью **Ctrl+C**) и вставить их (с помощью **Ctrl+V**) в строки 6, 7, 8, 9 и др.

На листе «Итоговые команды» формируется $\text{\TeX}$ -овский код, его надо просто скопировать и вставить в документ `*.tex`.

XI.6.1. Построение гладкой кривой по частям с помощью `\qbezier`

$$\begin{cases} y(a) = b, \\ y(c) = d, \\ y'(a) = k_1, \\ y'(c) = k_2 \end{cases} \Rightarrow \begin{cases} x_{\Pi} = \frac{ak_1 - ck_2 - b + d}{k_1 - k_2}, \\ y_{\Pi} = \frac{(a - c)k_1k_2 + dk_1 - bk_2}{k_1 - k_2}. \end{cases}$$



Проще всего после заполнения четвёртой строки скопировать пару строк с номерами 4, 5 (с помощью `Ctrl+C`) и вставить их (с помощью `Ctrl+V`) в строки 6, 7, 8, 9 и др.

Компилировать с помощью `latex` или `pdflatex` можно после замены десятичной запятой на десятичную точку.

XI.6.2. Создание графика с помощью \curve

Допустим, надо построить график функции $f(x) = \frac{1}{x}$ при $x \in (0; 2)$. Учитывая, что наибольшая «неравномерность» сосредоточена вблизи 0, применим неравномерный шаг: $x_n = \frac{2}{2^{n/4}}$.

Для создания рисунка можно использовать, например, Excel и Word. Допустим, Excel настроен на русский язык так, что разделитель целой части и мантиссы — запятая. В Excel создаем «шаблон» для команды `\curve`: на листе A1

[illegible]

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=**J1**»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/**B1**»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=**B1**/СТЕПЕНЬ(2;1/4)»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/F1»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/F1», H2: «=1/F2»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/B1», H2: «=1/B2», ...

J1: «=**F1**/СТЕПЕНЬ(2;1/4)»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/B1», H2: «=1/B2», ...

J1: «=F1/СТЕПЕНЬ(2;1/4)», J2: «=F2/СТЕПЕНЬ(2;1/4)»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/B1», H2: «=1/B2», ...

J1: «=F1/СТЕПЕНЬ(2;1/4)», J2: «=F2/СТЕПЕНЬ(2;1/4)»,

L1: «=1/J1»,

XI.6.2. Создание графика с помощью \curve

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	\curve(	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	\curve(	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)

3 ...

Здесь в ячейках B2, D1-D2-..., F1-F2-..., H1-H2-..., J1-J2-..., L1-L2-... содержатся формулы:

B2: «=J1»,

D1: «=1/B1», D2: «=1/B2», ...

F1: «=B1/СТЕПЕНЬ(2;1/4)», F2: «=B2/СТЕПЕНЬ(2;1/4)»,

H1: «=1/B1», H2: «=1/B2», ...

J1: «=F1/СТЕПЕНЬ(2;1/4)», J2: «=F2/СТЕПЕНЬ(2;1/4)»,

L1: «=1/J1», L2: «=1/J2», ...

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Для того, чтобы создать нужное количество строк, копируем вторую строку (выделяем мышкой и на клавиатуре набираем Ctrl+C) и вставляем ее в качестве третьей, четвертой и др. строк (нажимаем мышкой на ячейку A2, потом на клавиатуре набираем Ctrl+V, затем мышкой на A3 — на клавиатуре Ctrl+V и т.д.)

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

После того, как будет таким образом создано необходимое число строк, «прямоугольник» следует скопировать и вставить в чистый лист Word (в виде таблицы).

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

После того, как будет таким образом создано необходимое число строк, «прямоугольник» следует скопировать и вставить в чистый лист Word (в виде таблицы).

В Word выделяем весь текст (Ctrl+A) и проводим две замены: меняем «,» на «.» и, ПОТОМ «;» — на «,».

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

После того, как будет таким образом создано необходимое число строк, «прямоугольник» следует скопировать и вставить в чистый лист Word (в виде таблицы).

В Word выделяем весь текст (Ctrl+A) и проводим две замены: меняем «,» на «.» и, потом «;» — на «,».

В меню «Таблица» выбираем подменю «Преобразовать» -> «Таблицу в текст». В появившемся диалоговом окне выбираем в качестве разделителя «другой знак» — «пробел» (мышкой выбрать этот пункт и нажать клавишу «пробел» на клавиатуре). Потом нажмите ОК.

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Когда таблица будет преобразована в текст, выделите весь текст (Ctrl+A) и проведите глобальную замену «пробела» на «пустой символ».

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Когда таблица будет преобразована в текст, выделите весь текст (Ctrl+A) и проведите глобальную замену «пробела» на «пустой символ».

Полученный в итоге текст копируем и вставляем в создаваемый файл `*.tex` (в окружение `picture`).

XI.6.2. Создание графика с помощью `\curve`

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<code>\curve(</code>	2	;	0,5	;	1,682	;	0,595	;	1,414	;	0,707	)
2	<code>\curve(</code>	1,414	;	0,707	;	1,297	;	0,771	;	1,215	;	0,823	)
3	...												

Когда таблица будет преобразована в текст, выделите весь текст (Ctrl+A) и проведите глобальную замену «пробела» на «пустой символ».

Полученный в итоге текст копируем и вставляем в создаваемый файл `*.tex` (в окружение `picture`).

Делов-то...

XI.7. Пример получения некоторых целочисленных параметров для рисования

Как показывает опыт, нередко возникает необходимость в получении некоторых значений параметров для выполнения построений средствами $\text{\LaTeX}$.

Например, для построения биссектрисы желательно иметь примеры таких векторов $\vec{\mathbf{a}}$, $\vec{\mathbf{b}}$ с целочисленными координатами, расположенных на плоскости, что вектор $(\vec{\mathbf{a}} + \vec{\mathbf{b}})$ направлен по биссектрисе угла, образованного векторами $\vec{\mathbf{a}}$ и $\vec{\mathbf{b}}$.

Список таких векторов можно получить средствами $\text{\TeX}$ или $\text{\LaTeX}$.

Рассмотрим пример?

XII. Оформление текста в виде презентаций

Создавать презентацию можно «с нуля» или на основе уже имеющегося материала по заданной теме (например, электронных конспектов лекций). Во втором случае имеющийся материал требует определенной переработки для представления в виде презентаций. Текст необходимо разбить на смысловые фрагменты, которые можно разместить на одном слайде с учетом существенных ограничений по объему представляемого материала.

XII.1. Распределение большого фрагмента текста на слайдах

В случае, когда некоторый смысловой фрагмент из-за его большого объема не может быть помещен на один слайд, следует предусмотреть, во-первых,

XII.1. Распределение большого фрагмента текста на слайдах

В случае, когда некоторый смысловой фрагмент из-за его большого объема не может быть помещен на один слайд, следует предусмотреть, во-первых, возможность быстрого переключения между слайдами, например, с помощью [гиперссылок](#), во-вторых,

XII.1. Распределение большого фрагмента текста на слайдах

В случае, когда некоторый смысловой фрагмент из-за его большого объема не может быть помещен на один слайд, следует предусмотреть, во-первых, возможность быстрого переключения между слайдами, например, с помощью [гиперссылок](#), во-вторых, во время занятия выделить время на переписывание слушателями краткого содержания слайдов.

XII.1. Распределение большого фрагмента текста на слайдах

В случае, когда некоторый смысловой фрагмент из-за его большого объема не может быть помещен на один слайд, следует предусмотреть, во-первых, возможность быстрого переключения между слайдами, например, с помощью [гиперссылок](#), во-вторых, во время занятия выделить время на переписывание слушателями краткого содержания слайдов. Рациональным является выделение серии слайдов, предназначенных только для восприятия информации, без записывания,

XII.1. Распределение большого фрагмента текста на слайдах

В случае, когда некоторый смысловой фрагмент из-за его большого объема не может быть помещен на один слайд, следует предусмотреть, во-первых, возможность быстрого переключения между слайдами, например, с помощью [гиперссылок](#), во-вторых, во время занятия выделить время на переписывание слушателями краткого содержания слайдов. Рациональным является выделение серии слайдов, предназначенных только для восприятия информации, без записывания, завершающихся специальными слайдами с краткой версией представленной ранее информации, предназначенной для запоминания и записи в тетради.

XII.1. Распределение большого фрагмента текста на слайдах

Рациональным является выделение на слайде областей «относительно неподвижных», содержание которых не меняется при переходе к следующему слайду,

Это ↑ «относительно неподвижная» область

XII.1. Распределение большого фрагмента текста на слайдах

Рациональным является выделение на слайде областей «относительно неподвижных», содержание которых не меняется при переходе к следующему слайду,

Это ↑ «относительно неподвижная» область

и областей с «динамичным содержанием», изменяющимся на следующем слайде.

Это ↑ область с «динамичным содержанием»

Вот как это выглядит:

XII.1. Распределение большого фрагмента текста на слайдах

Рациональным является выделение на слайде областей «относительно неподвижных», содержание которых не меняется при переходе к следующему слайду,

XII.1. Распределение большого фрагмента текста на слайдах

Рациональным является выделение на слайде областей «относительно неподвижных», содержание которых не меняется при переходе к следующему слайду, и областей с «динамичным содержанием», изменяющимся на следующем слайде.

XII.1. Распределение большого фрагмента текста на слайдах

Рациональным является выделение на слайде областей «относительно неподвижных», содержание которых не меняется при переходе к следующему слайду, и областей с «динамичным содержанием», изменяющимся на следующем слайде.

Обеспечение «относительной неподвижности» области экрана иногда является непростой задачей. Этой теме посвящен **специальный раздел**.

Для выделения «неподвижных блоков» нередко требуется **явное задание длины**. Во многих случаях для этого целесообразно применять **специальные команды**.

XII.2. Единицы длины

В $\text{T}_{\text{E}}\text{X}$ и $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ применяются следующие единицы длины:

pt		— пункт $\approx 0,35$ мм;
pc	■	— пика = 12 pt;
mm	l	— миллиметр;
cm	■	— сантиметр;
in	■	— дюйм (inch) = 25,4 мм;
dd		— пункт Дидо = 1,07 pt;
cc	■	— цицеро = 12 dd;
em	■	— ширина буквы М в текущем шрифте;
ex	■	— высота буквы х в текущем шрифте.

Для того, чтобы установить стандартную единицу длины для команд, использующих безразмерные величины длины, применяется команда `\unitlength{величина длины}`

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\newlength{команда-длина}` — команда для определения команды-длины, т.е. команды, обозначающей длину.

Например, `\newlength{\MyLength}` определяет новую длину `\MyLength`.

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\setlength{команда-длина}{значение длины}` — команда присвоения команде-длине значения длины.

Например, если была определена новая длина командой `\newlength{\MyLength}`, то `\setlength{\MyLength}{12mm}` делает синонимами `\MyLength` и 12 mm.

Компилирование кода:

```
\newlength{\MyLength}
```

```
\setlength{\MyLength}{12mm}
```

```
\rule{5mm}{\MyLength}\quad \rule{5mm}{12mm}
```

приведёт к следующему результату:



XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\addtolength{команда-длина}{добавляемое значение длины}` — команда прибавления к текущему значению команды-длины добавляемого значения длины.

Например, если была определена новая длина командой `\newlength{\MyLength}`, и командой `\setlength{\MyLength}{12mm}` команде `\MyLength` было присвоено значение 12 mm, то после `\addtolength{\MyLength}{7mm}` команда `\MyLength` станет равносильна 19 mm.

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\settodepth`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине (т.е. команде, обозначающей длину) значения, равного глубине текстовой строки, т.е. расстояния от базисной линии до нижнего края этой текстовой строки.

Команда-длина должна быть уже введена (см. `\setlength`).

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\settodepth`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине (т.е. команде, обозначающей длину) значения, равного глубине текстовой строки, т.е. расстояния от базисной линии до нижнего края этой текстовой строки.

Команда-длина должна быть уже введена (см. `\setlength`).

`\settoheight`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине значения, равного высоте текстовой строки, т.е. расстояния от базисной линии до верхнего края этой текстовой строки.

XII.3. Команды со значением длины

TeX позволяет определять команды со значением длины. Мы кратко назовём их **командами-длинами**.

`\settodepth`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине (т.е. команде, обозначающей длину) значения, равного глубине текстовой строки, т.е. расстояния от базисной линии до нижнего края этой текстовой строки.

Команда-длина должна быть уже введена (см. `\setlength`).

`\settoheight`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине значения, равного высоте текстовой строки, т.е. расстояния от базисной линии до верхнего края этой текстовой строки.

`\settowidth`{команда-длина}{Текстовая строка} — команда для присвоения команде-длине (т.е. команде, обозначающей длину) значения, равного ширине текстовой строки.

XII.4. Оверлей

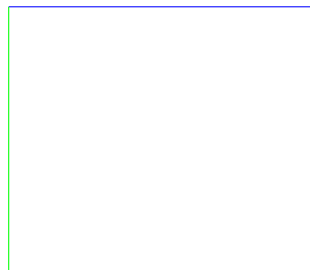
Постепенное появление текста (в частности, формулы), можно организовать с помощью команд `\ComOverlayA`, `\ComOverlayB`, `\ComOverlayX`.

ХII.5. Выделение нескольких неподвижных горизонтальных блоков

Нередко возникает ситуация, когда при смене слайдов, например, рисунок должен оставаться неподвижным, несмотря на изменение величины окружающих компонентов: абзацев, других рисунков. Для этого можно применить, например, команды `\makebox` в сочетании с командой `\parbox`.

Пример использования этих команд содержится в файле `000ris.tex`

XII.5. Выделение нескольких неподвижных горизонтальных блоков



Для печати
текста в пря-
моугольнике
справа от
рисунка при-

меняем данную
конструкцию.

В рассматри-
ваемом случае
введен доволь-
но странный
вариант —

печать в три
колонки

А здесь пече-
тается второй
столбец.


```

{\unitlength=1mm
\begin{picture}(40.00,35.00)
  \put(00.00,00.00){ {\color{blue} \line ( 1, 0){40.00}}
  ...
\end{picture}~%
{\unitlength=1mm
\fbbox{\makebox(40,35)[lt]{\parbox{40mm}{Содержимое второй колонки}}~
{\unitlength=1mm
\fbbox{ \makebox(40,35)[lt]{\parbox{40mm}{Содержимое третьей колонки}}
{\unitlength=1mm
\fbbox{ \makebox(40,35)[lt]{\parbox{40mm}{Содержимое четвертой
колонки}}}

```

ХII.6. Выделение двух неподвижных горизонтальных блоков

В случае необходимости выделения *двух* горизонтальных блоков, можно воспользоваться введенной в файле `texsample.tex` командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}`.

ХII.6. Выделение двух неподвижных горизонтальных блоков

В случае необходимости выделения *двух* горизонтальных блоков, можно воспользоваться введенной в файле `texsample.tex` командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}`.

Первый и второй аргументы `#1` и `#2` означают ширину, соответственно, первого и второго блоков, измеряемую в долях от ширины текста, т.е. от значения переменной `\textwidth`.

ХII.6. Выделение двух неподвижных горизонтальных блоков

В случае необходимости выделения *двух* горизонтальных блоков, можно воспользоваться введенной в файле `texsample.tex` командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}`.

Первый и второй аргументы `#1` и `#2` означают ширину, соответственно, первого и второго блоков, измеряемую в долях от ширины текста, т.е. от значения переменной `\textwidth`.

Третий аргумент `#3` равен высоте блоков (высота у них будет одинаковой) в тех же единицах.

ХII.6. Выделение двух неподвижных горизонтальных блоков

В случае необходимости выделения *двух* горизонтальных блоков, можно воспользоваться введенной в файле `texsample.tex` командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}`.

Первый и второй аргументы `#1` и `#2` означают ширину, соответственно, первого и второго блоков, измеряемую в долях от ширины текста, т.е. от значения переменной `\textwidth`.

Третий аргумент `#3` равен высоте блоков (высота у них будет одинаковой) в тех же единицах.

Значением четвертого `#4` и пятого аргумента `#5` являются содержание первого и, соответственно, второго блока. При этом текст автоматически форматируется по ширине столбца.

ХII.6. Выделение двух неподвижных горизонтальных блоков

В случае необходимости выделения *двух* горизонтальных блоков, можно воспользоваться введенной в файле `texsample.tex` командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}`.

Разница между командами `\TwoBlock{#1}{#2}{#3}{#4}{#5}` и `\fTwoBlock{#1}{#2}{#3}{#4}{#5}` в том, что при использовании второй команды `\fTwoBlock{#1}{#2}{#3}{#4}{#5}` текст блоков окружается рамкой.

XII.6. Выделение двух неподвижных горизонтальных блоков

```
\noindent\fTwoBlock{0.3}{0.65}{0.14}
```

```
{Текст первой колонки. Сейчас <<защита>> ширина столбца  
0.3\textwidth.}
```

```
{Текст второй колонки. В данном случае он более чем вдвое шире  
первого столбца, его ширина  
равна 0.65\textwidth.}
```

Текст первой колонки. Сейчас «защита» ширина столбца 0.3\textwidth.

Текст второй колонки. В данном случае он более чем вдвое шире первого столбца, его ширина равна 0.65\textwidth.

Вставлена команда `\noindent` для подавления абзацного отступа.

XII.6. Выделение двух неподвижных горизонтальных блоков

```
\noindent\TwoBlock{0.3}{0.65}{0.14}
```

```
{Текст первой колонки. Сейчас <<защита>> ширина столбца  
0.3\textwidth.}
```

```
{Текст второй колонки. В данном случае он более чем вдвое шире  
первого столбца, его ширина  
равна 0.65\textwidth.}
```

```
Текст первой колонки. Сейчас «защита» ширина столбца 0.3\textwidth.  
Текст второй колонки. В данном случае он более чем вдвое шире первого столбца, его ширина равна 0.65\textwidth.
```

Вставлена команда `\noindent` для подавления абзацного отступа.

XIII. Оформление слайда

Оформление слайда для учебного пособия, выполненного в формате электронных презентаций, в большой степени представляет собой искусство,

направленное на решение дидактических задач.

Все рассмотренные выше разделы были в той или иной степени ориентированы на решение этой задачи. Здесь мы рассмотрим несколько особых моментов.

XIII.1. «Неподвижность» фрагмента при переходе к другому слайду

Выше были рассмотрены приемы для **сохранения номера раздела**, сохранения **номера теоремы, определения и др.**

Для сохранения размеров объекта, например, высоты строки или ее длины можно применять следующие приемы:

XIII.1. «Неподвижность» фрагмента при переходе к другому слайду

Выше были рассмотрены приемы для **сохранения номера раздела**, сохранения **номера теоремы, определения и др.**

Для сохранения размеров объекта, например, высоты строки или ее длины можно применять следующие приемы:

- использование команды `\makebox`;
- вставка невидимых «линеек» `\rule`;
- использование команды `\phantom`.
- Выделение фрагментов текста в отдельную строку с помощью команды `\newline` или выделением текста в отдельный параграф (вставка пустой строки или команды `\par`)

XIII.2. Фоновый рисунок

По нашему мнению использование фонового рисунка с дидактической точки зрения, как правило, нецелесообразно. Если, тем не менее, есть необходимость использовать фоновый рисунок, это можно сделать, используя команды

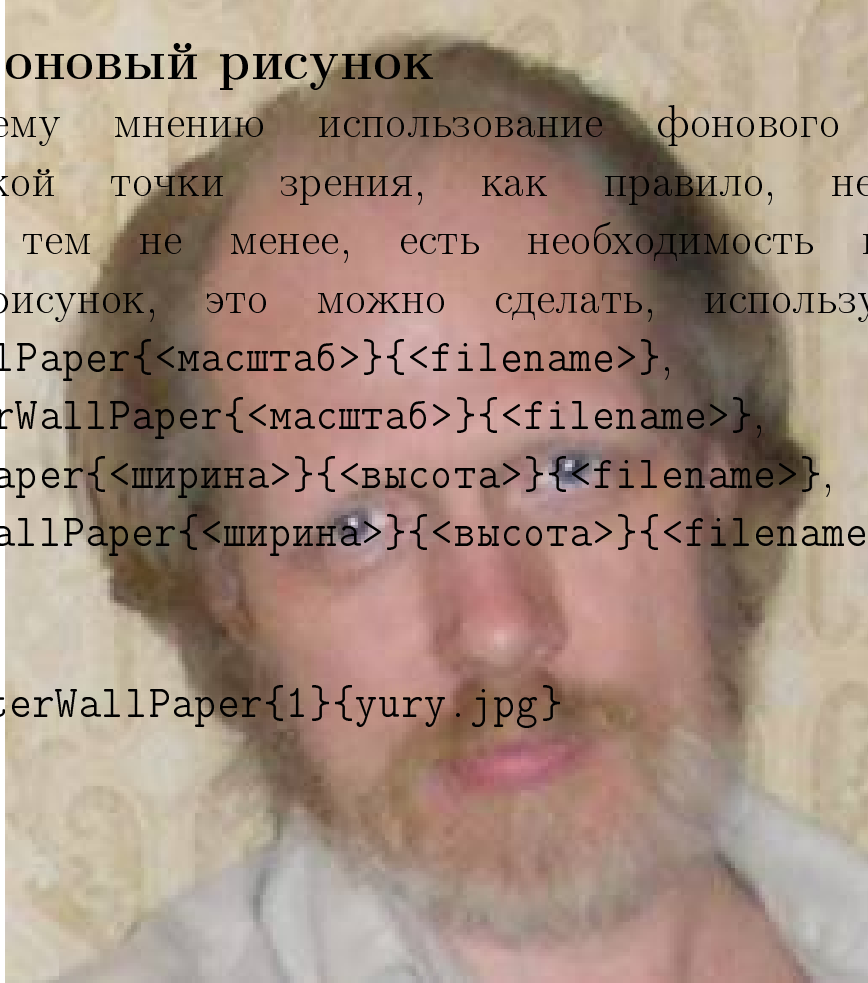
```
\CenterWallPaper{<масштаб>}{<filename>},  
\ThisCenterWallPaper{<масштаб>}{<filename>},  
\TileWallPaper{<ширина>}{<высота>}{<filename>},  
\ThisTileWallPaper{<ширина>}{<высота>}{<filename>}.
```

XIII.2. Фоновый рисунок

По нашему мнению использование фонового рисунка с дидактической точки зрения, как правило, нецелесообразно. Если, тем не менее, есть необходимость использовать фоновый рисунок, это можно сделать, используя команды

```
\CenterWallPaper{<масштаб>}{<filename>},  
\ThisCenterWallPaper{<масштаб>}{<filename>},  
\TileWallPaper{<ширина>}{<высота>}{<filename>},  
\ThisTileWallPaper{<ширина>}{<высота>}{<filename>}.
```

```
\ThisCenterWallPaper{1}{yury.jpg}
```



XIII.2. Фоновый рисунок

По нашему мнению использование фонового рисунка с дидактической точки зрения, как правило, нецелесообразно. Если, тем не менее, есть необходимость использовать фоновый рисунок, это можно сделать, используя команды

```
\CenterWallPaper{<масштаб>}{<filename>},  
\ThisCenterWallPaper{<масштаб>}{<filename>},  
\TileWallPaper{<ширина>}{<высота>}{<filename>},  
\ThisTileWallPaper{<ширина>}{<высота>}{<filename>}.
```

```
\ThisTileWallPaper{\textwidth}{\textheight}{yury.jpg}
```

XIII.2. Фоновый рисунок

Если по каким-то причинам использование команды `\wallpaper` невозможно, можно применить следующий набор команд:

```
\def\newpagemy{\newpage\noindent  
\makebox(0,0)[lt]{\includegraphics[height=1.0\textheight]  
{yury.jpg}}}  
  
\newpagemy
```

Текст страницы.

После компиляции получается следующий результат.

XIII.2. Фоновый рисунок

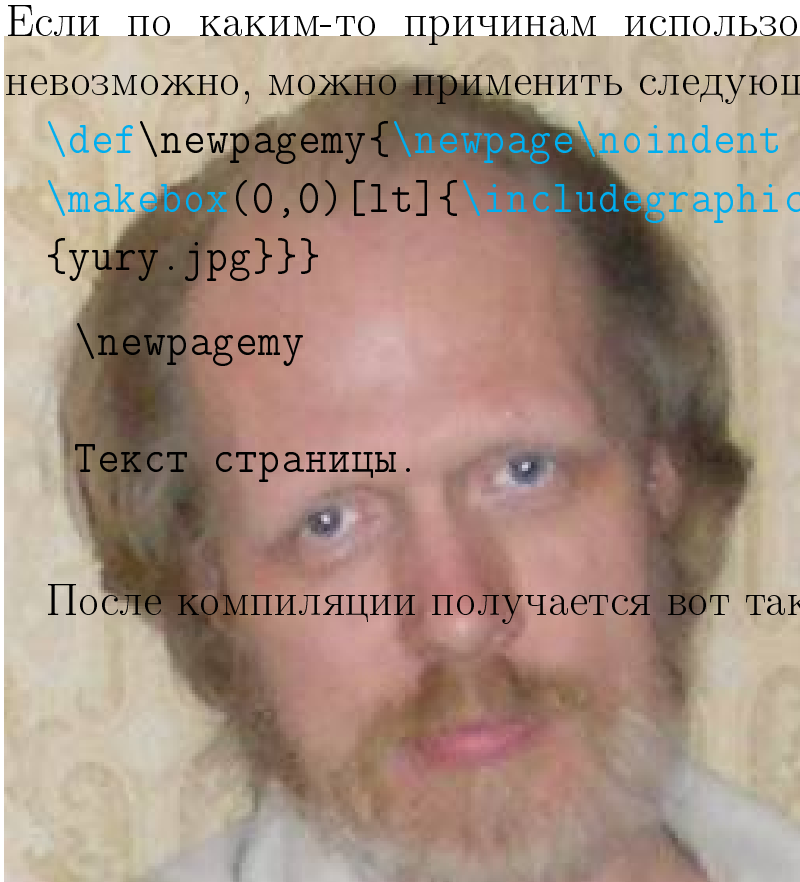
Если по каким-то причинам использование команды `\wallpaper` невозможно, можно применить следующий набор команд:

```
\def\newpagemy{\newpage\noindent  
\makebox(0,0)[lt]{\includegraphics[height=1.0\textheight]  
{yury.jpg}}}
```

```
\newpagemy
```

Текст страницы.

После компиляции получается вот такой результат!



XIII.2. Фоновый рисунок

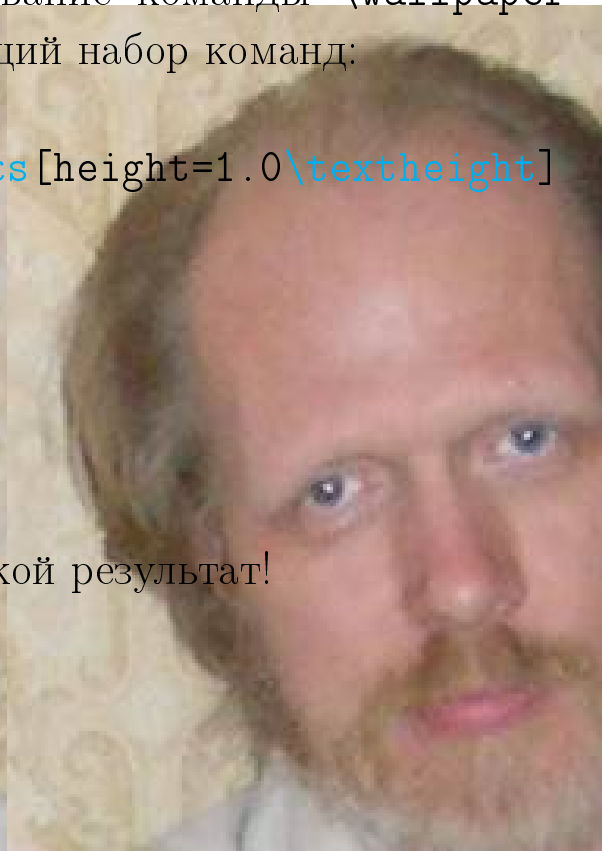
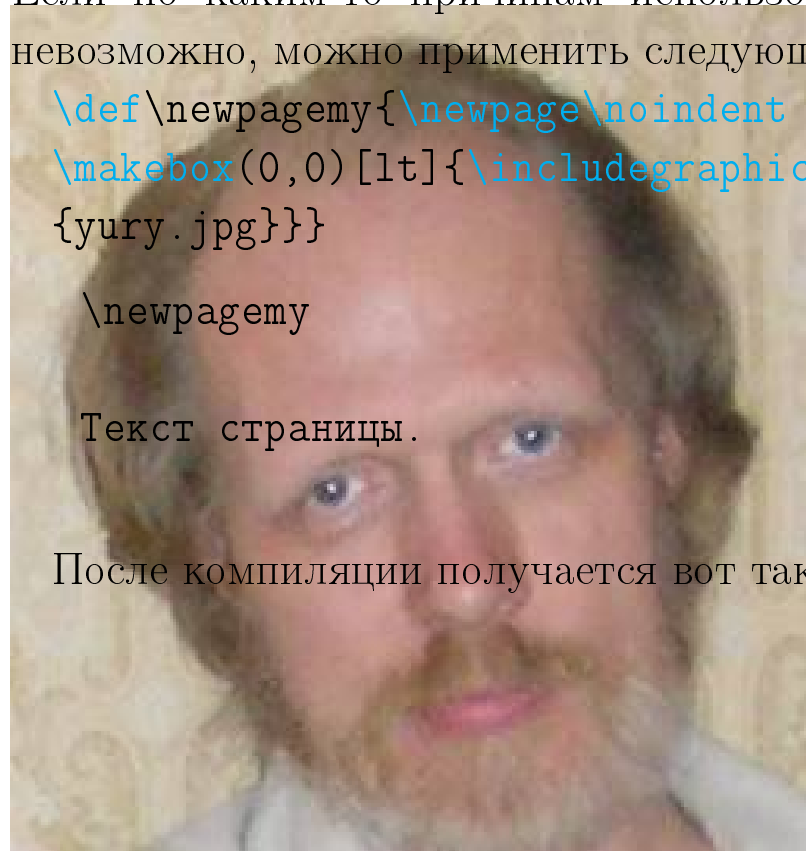
Если по каким-то причинам использование команды `\wallpaper` невозможно, можно применить следующий набор команд:

```
\def\newpagemy{\newpage\noindent  
\makebox(0,0)[lt]{\includegraphics[height=1.0\textheight]  
{yury.jpg}}}
```

```
\newpagemy
```

Текст страницы.

После компиляции получается вот такой результат!



Если по каким-то причинам использование команды `\wallpaper` невозможно, можно применить следующий набор команд:

```
\def\newpagemy{\newpage\noindent  
\makebox(0,0)[lt]{%  
\begin{picture}(190,110)  
\put(-5.00,-10.00){%  
\includegraphics[height=1.0\textheight]{zakat.jpg}}  
\end{picture}}}
```

```
\newpagemy
```

Текст страницы.

После компиляции получается вот такой результат!

XIV. Разработка тестов в системе L^AT_EX

Средства издательской системы L^AT_EX позволяют разрабатывать тесты, в частности, с помощью пакета `acrotex`.

XIV.1. Тесты для лекционного или практического занятия

В настоящий момент учебные занятия по математике, физике и др. редко проводятся в компьютерных классах. Поэтому актуальным является проведение тестов, в которых задание предъявляется на билетах или с помощью проектора.

Рассмотрим создание тестов, предъявление которых студентам осуществляется на лекции или практическом занятии с помощью проектора.

XIV.2. Преамбула `StartTestBigScr.tex`

Высокую эффективность показала система тестирования, осуществляется на лекции или практическом занятии с помощью проектора.

Задания оформляются в виде, представленном в приложении в файле `BasaBigScrTmp.tex`, а непосредственно тест создаётся по образцу, представленному в файле `BigScrTestTmp.tex`.

XIV.2. Преамбула `StartTestBigScr.tex`

Не забудьте переименовать файлы `BasaBigScrTmp.tex` и `BigScrTestTmp.tex`, прежде чем создавать собственные тесты!

Высокую эффективность показала система тестирования, осуществляется на лекции или практическом занятии с помощью проектора.

Задания оформляются в виде, представленном в приложении в файле `BasaBigScrTmp.tex`, а непосредственно тест создаётся по образцу, представленному в файле `BigScrTestTmp.tex`.

XIV.2. Преамбула `StartTestBigScr.tex`

Задания заносятся в файл `BasaBigScr?????.tex`, полученный переименованием файла `BasaBigScrTmp.tex`.

Например, задания по тригонометрии можно занести в файл с названием `BasaBigScrTrigonA.tex`, т.е. `?????` в данном случае означает `TrigonA`.

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScrTmp.tex}
```

```
\global\def\ThisCommandName{BasaBigScrTmp.tex}
```

```
\newcommand{\BasaBigScrTmp.tex}[3]{...
```


XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScrTmp.tex}
```

```
\global\def\ThisCommandName{BasaBigScrTmp.tex}
```

```
\newcommand{\BasaBigScrTmp.tex}[3]{...
```

Заменим на реальное имя файла (вместо знаков вопроса должна быть цепочка букв латинского алфавита)...

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScr?????.tex}  
\global\def\ThisCommandName{BasaBigScrTmp.tex}  
\newcommand{\BasaBigScrTmp.tex}[3]{...
```

Заменим на реальное имя файла (вместо знаков вопроса должна быть цепочка букв латинского алфавита)...

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScr?????.tex}  
\global\def\ThisCommandName{BasaBigScrTmp.tex}  
\newcommand{\BasaBigScrTmp.tex}[3]{...
```

Заменим на реальное имя макрокоманды, определяемой в данном файле (знаки вопроса надо заменить на ту же цепочку букв латинского алфавита, что и в имени строкой выше)...

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScr?????.tex}
```

```
\global\def\ThisCommandName{BasaBigScr?????.tex}
```

```
\newcommand{\BasaBigScr?????.tex}[3]{...
```

Заменим на реальное имя макрокоманды, определяемой в данном файле (знаки вопроса надо заменить на ту же цепочку букв латинского алфавита, что и в имени строкой выше)...

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\global\def\ThisFileName{BasaBigScr?????.tex}  
\global\def\ThisCommandName{BasaBigScr?????.tex}  
\newcommand{\BasaBigScr?????.tex}[3]{...
```

Таким образом получим указанное начало файла BasaBigScr?????.tex (знаки вопроса надо заменить на ту же цепочку букв латинского алфавита, что и в имени строкой выше)...

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\or{Вопрос
```

```
    первого типа варианта~1 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 1 в базе.

```
\fi}}
```

```
\or{Вопрос
```

```
    первого типа варианта~2 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 2 в базе.

```
\fi}}...
```

XIV.2. Преамбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\or{Вопрос
```

```
    первого типа варианта~1 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 1 в базе.

```
\fi}}
```

```
\or{Вопрос
```

```
    первого типа варианта~2 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 2 в базе.

```
\fi}}...
```

Предполагается, что все задачи одного типа отличаются друг от друга незначительно.

XIV.2. Пreamбула StartTestBigScr.tex

Задания заносятся в файл BasaBigScr?????.tex, полученный переименованием файла BasaBigScrTmp.tex.

```
\or{Вопрос
```

```
первого типа варианта~1 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 1 в базе.

```
\fi}}
```

```
\or{Вопрос
```

```
первого типа варианта~2 в базе.
```

```
{\ifnum#3>0\newline\textbf{Ответ:}}
```

Ответ к вопросу первого типа варианта 2 в базе.

```
\fi}}...
```

Однотипные задания предъявляются студентам в разное время, поэтому «одинаковость» заданий не бросается в глаза, а воспользоваться подсказкой соседа всё равно не получится, поскольку он в этот момент решает другую задачу при дефиците времени.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Рассмотрим пример?

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Пробный тест}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Пробный тест}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

Укажите количество заданий, предъявляемых студенту.

В **рассмотренном примере** их было 5, поскольку таким было определение команды \QuantityOfVariants.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{7}  
\newcommand{\ThemeOfThisTest}{Пробный тест}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

Укажите количество заданий, предъявляемых студенту.

При данном определении команды \QuantityOfVariants получим 7 заданий.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Пробный тест}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

Измените определение команды \ThemeOfThisTest, например, вместо **Пробный тест** напомним

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Пробный тест}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

Измените определение команды \ThemeOfThisTest, например, вместо **Пробный тест** напомним **Тригонометрия**.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Тригонометрия}  
\newcommand{\FileNameMini}{BigScrTestTmp}  
\input BasaBigScrTmp.tex
```

Измените определение команды \ThemeOfThisTest, например, вместо **Пробный тест** напомним **Тригонометрия**.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Тригонометрия}  
\newcommand{\FileNameMini}{BigScrTest????}  
\input BasaBigScrTmp.tex
```

Теперь надо отредактировать определение команды \FileNameMini, заменив BigScrTestTmp на имя итогового файла с тестом. Это имя, вообще говоря, может быть не связано с BasaBigScr....

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\QuantityOfVariants}{5}  
\newcommand{\ThemeOfThisTest}{Тригонометрия}  
\newcommand{\FileNameMini}{BigScrTestXXX}  
\input BasaBigScr?????.tex
```

Теперь следует указать имена файлов с заданиями. Можно загружать несколько файлов.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestaa}[2]{\ifcase#1{}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{2}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\fi}
```

Теперь отредактируем список предъявляемых заданий и порядок их предъявления.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestaa}[2]{\ifcase#1{}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{2}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\or{\BasaBigScrTmp{1}{1}{#2}}  
\fi}
```

Исправьте имена макрокоманд, подгружаемых из файлов BasaBigScr?????.tex.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestaa}[2]{\ifcase#1{}  
\or{\BasaBigScr???{1}{1}{#2}}  
\or{\BasaBigScr???{2}{1}{#2}}  
\or{\BasaBigScr???{1}{1}{#2}}  
\or{\BasaBigScr???{1}{1}{#2}}  
\or{\BasaBigScr???{1}{1}{#2}}  
\fi}
```

Исправьте имена макрокоманд, подгружаемых из файлов BasaBigScr?????.tex.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestaa}[2]{\ifcase#1{}  
\or{\BasaBigScr???{1}{1}{#2}}  
\or{\BasaBigScr???{2}{1}{#2}}  
\or{\BasaBigScr???{3}{1}{#2}}  
\or{\BasaBigScr???{4}{1}{#2}}  
\or{\BasaBigScr???{5}{1}{#2}}  
\fi}
```

Укажите значения типов задач.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestab}[2]{\ifcase#1{}  
\or{\BasaBigScr???{5}{2}{#2}}  
\or{\BasaBigScr???{3}{2}{#2}}  
\or{\BasaBigScr???{1}{2}{#2}}  
\or{\BasaBigScr???{2}{2}{#2}}  
\or{\BasaBigScr???{4}{2}{#2}}  
\fi}
```

Укажите значения типов задач. Отредактируйте \MiniTestab и проследите, чтобы на экране они у соседей не повторялись.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestba}[2]{\ifcase#1{}  
\or{\BasaBigScr???{3}{3}{#2}}  
\or{\BasaBigScr???{1}{3}{#2}}  
\or{\BasaBigScr???{4}{3}{#2}}  
\or{\BasaBigScr???{5}{3}{#2}}  
\or{\BasaBigScr???{2}{3}{#2}}  
\fi}
```

Укажите значения типов задач. Отредактируйте \MiniTestba и проследите, чтобы на экране они у соседей не повторялись.

XIV.2. Пreamбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Для этого вы должны откорректировать файл BigScrTest?????.tex:

```
\newcommand{\MiniTestbbb}[2]{\ifcase#1{}  
\or{\BasaBigScr???{5}{4}{#2}}  
\or{\BasaBigScr???{3}{4}{#2}}  
\or{\BasaBigScr???{1}{4}{#2}}  
\or{\BasaBigScr???{2}{4}{#2}}  
\or{\BasaBigScr???{4}{4}{#2}}  
\fi}
```

Укажите значения типов задач. Отредактируйте \MiniTestbbb и проследите, чтобы на экране они у соседей не повторялись.

XIV.2. Преамбула StartTestBigScr.tex

Компиляция файла BigScrTest?????.tex, полученного переименованием файла BigScrTestTmp.tex, приводит к формированию **системы заданий и ответов**.

Осталось скомпилировать файл BigScrTest???.tex, записать его на флешку и, приняв ядовидый вид, в нужный момент на лекции или практическом занятии проести подготовленный вами тест.

XIV.3. Преамбула StartTestE.tex для пакета exerquiz

Рассмотрим пакет AcroTeX, стилевой файл exerquiz, который позволяет создавать тесты для самоконтроля, который студент будет проходить самостоятельно за компьютером, смартфоном и др.

Совместить некоторые возможности пакета exerquiz и texsample можно с помощью загрузки преамбулы StartTestE.tex. Как обычно, загрузка осуществляется командой

```
\input StartTestE.tex
```

Рассмотрим систему подготовки электронных тестов, принятую на кафедре Прикладной математики?

XIV.4. Пакет exerquiz, окружение shortquiz

Применяется для тестов, применяемых для самопроверки.

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

2 — это количество вариантов ответа.

Может быть, например, **3**: да, нет, не всегда, и т.п.

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

\Ans1 означает, что этот вариант ответа верен (в данном случае это ответ «Да»).

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

\Ans0 означает, что этот вариант ответа неверен (в данном случае это ответ «Нет»).

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Текст ответа.

```
\end{solution}
```

```
\end{shortquiz}
```

Ответ можно выделить, сделав отступы от краев листа с помощью окружения quote

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
```

Текст вопроса.

```
\begin{answers}[qz:МеткаОтвета]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

```
\begin{quote}Текст ответа.\end{quote}
```

```
\end{solution}
```

```
\end{shortquiz}
```

Ответ можно выделить, сделав отступы от краев листа с помощью окружения quote

XIV.4. Пакет exerquiz, окружение shortquiz

```
\begin{shortquiz}
  Текст вопроса.
  \begin{answers}[qz:МеткаОтвета]{2}
    \bChoices
    \Ans1 Да\еAns
    \Ans0 Нет\еAns
  \eChoices
\end{answers}
\begin{solution}
\begin{quote}Текст ответа.\end{quote}
\end{solution}
\end{shortquiz}
```

Рассмотрим результат компиляции?

XIV.5. Пакет exerquiz, окружение quiz

Это окружение для создания тестов.

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{2}
```

Второй вопрос (выбор одного ответа из 4 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaB]{4}
```

```
\bChoices
```

```
\Ans0 Да\еAns
```

```
\Ans1 Нет\еAns
```

```
\Ans0 Очень\еAns
```

```
\Ans0 Не совсем\еAns
```

```
\еChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на второй вопрос (с выбором одного ответа из 4 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{7}
```

Третий вопрос (выбор одного ответа из 5 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaC]{5}
```

```
\bChoices
```

```
\Ans0 5\eAns
```

```
\Ans0 7\eAns
```

```
\Ans0 10\eAns
```

```
\Ans1 13\eAns
```

```
\Ans1 15\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на третий вопрос (с выбором одного ответа из 5 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{7}
```

Третий вопрос (выбор одного ответа из 5 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaC]{5}
```

```
\bChoices
```

```
\Ans0 5\eAns
```

```
\Ans0 7\eAns
```

```
\Ans0 10\eAns
```

```
\Ans1 13\eAns
```

```
\Ans1 15\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на третий вопрос (с выбором одного ответа из 5 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

```
\begin{questions}
```

```
\item\PTs{7}
```

Третий вопрос (выбор одного ответа из 5 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaC]{5}
```

```
\bChoices
```

```
\Ans0 5\eAns
```

```
\Ans0 7\eAns
```

```
\Ans0 10\eAns
```

```
\Ans1 13\eAns
```

```
\Ans1 15\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на третий вопрос (с выбором одного ответа из 5 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

Можно назначить количество баллов

```
\begin{questions}
```

за каждое задание. В данном случае

```
\item\PTs{7}
```

7 баллов.

Третий вопрос (выбор одного ответа из 5 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaC]{5}
```

```
\bChoices
```

```
\Ans0 5\eAns
```

```
\Ans0 7\eAns
```

```
\Ans0 10\eAns
```

```
\Ans1 13\eAns
```

```
\Ans1 15\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на третий вопрос (с выбором одного ответа из 5 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

Можно назначить количество баллов

```
\begin{questions}
```

за каждое задание. В данном случае

```
\item\PTs{2}
```

2 балла.

Второй вопрос (выбор одного ответа из 4 вариантов, но правильных ответов может быть несколько).

```
\begin{answers}[qz:testAbaB]{4}
```

```
\bChoices
```

```
\Ans0 Да\еAns
```

```
\Ans1 Нет\еAns
```

```
\Ans0 Очень\еAns
```

```
\Ans0 Не совсем\еAns
```

```
\еChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на второй вопрос (с выбором одного ответа из 4 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

Можно назначить количество баллов

```
\begin{questions}
```

за каждое задание. В данном случае

```
\item\PTs{1}
```

1 балл.

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

```
Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz
```

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов. Рассмотрим результат компиляции?

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

```
Points: \PointsField\currQuiz\    Percent: \PercentField\currQuiz
```

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

Можно ввести команды для демонстрации верных ответов и показа верного ответа на каждый вопрос.

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad
```

```
Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz
```

XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов.

Можно ввести команды для демонстрации верных ответов и показа верного ответа на каждый вопрос.

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad \ScoreField\currQuiz\eqButton\currQuiz
```

```
Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz
```


XIV.5.1. Тесты с выбором одного варианта

```
\begin{quiz}*{qztestAba}
```

Тест из нескольких вопросов. Рассмотрим результат компиляции?

```
\begin{questions}
```

```
\item\PTs{1}
```

Первый вопрос (выбор одного ответа из 2 вариантов).

```
\begin{answers}[qz:testAbaA]{2}
```

```
\bChoices
```

```
\Ans1 Да\eAns
```

```
\Ans0 Нет\eAns
```

```
\eChoices
```

```
\end{answers}
```

```
\begin{solution}
```

Ответ на первый вопрос (с выбором одного ответа из 2 вариантов).

```
\end{solution}
```

```
\end{questions}
```

```
\end{quiz}\quad \ScoreField\currQuiz\eqButton\currQuiz
```

```
Points: \PointsField\currQuiz\    Percent: \PercentField\currQuiz
```

XIV.5.2. Тесты с выбором и демонстрацией ответов

```
\begin{quiz}*{qztestAbc} Преамбула, некоторый текст.  
\begin{questions}  
  \item \PTs{2}  
    Вопрос с выбором нескольких верных вариантов ответа.  
    Не понимает слова на русском языке в качестве ответа.  
    \begin{manswers}[qz:testAbcA]{2}  
      \bChoices[4]  
        \Ans0 \mbox{$x'-2x=0$} \eAns          \Ans1 Слон\eAns  
        \Ans1 $\bigtriangleup$ \eAns          \Ans0 Слово\eAns  
      \eChoices  
    \end{manswers}\CorrAnsButton{$\bigtriangleup$, Слон}  
  \end{questions}  
\end{quiz} \ScoreField{testAbc} \eqButton\currQuiz\  
Percent: \PercentField\currQuiz  
  
\noindent  Answers: \AnswerField\currQuiz
```

XIV.5.2. Тесты с выбором и демонстрацией ответов

`\begin{quiz}*{qztestAbc}` Преамбула, некоторый текст.

`\begin{questions}` В данном случае указан максимальный балл за зада-
`\item \PTs{2}` ние.

Вопрос с выбором нескольких верных вариантов ответа.

Не понимает слова на русском языке в качестве ответа.

`\begin{manswers}[qz:testAbcA]{2}`

`\bChoices[4]`

`\Ans0 \mbox{$x'-2x=0$}` `\eAns`

`\Ans1 Слон\eAns`

`\Ans1 $\bigtriangleup$` `\eAns`

`\Ans0 Слово\eAns`

`\eChoices`

`\end{manswers}\CorrAnsButton{$\bigtriangleup$, Слон}`

`\end{questions}`

`\end{quiz}` `\ScoreField{testAbc}` `\eqButton\currQuiz\`

`Percent: \PercentField\currQuiz`

`\noindent` `Answers: \AnswerField\currQuiz`

XIV.5.2. Тесты с выбором и демонстрацией ответов

`\begin{quiz}*{qztestAbc}` Преамбула, некоторый текст.

`\begin{questions}` В данном случае указан максимальный балл за зада-
`\item \PTs{2}` ние.

Вопрос с выбором нескольких верных вариантов ответа.

Не понимает слова на русском языке в качестве ответа.

`\begin{manswers}[qz:testAbcA]{2}`

`\bChoices[4]`

`\Ans0 \mbox{$x'-2x=0$}` `\eAns`

`\Ans1 Слон\eAns`

`\Ans1 $\bigtriangleup$` `\eAns`

`\Ans0 Слово\eAns`

`\eChoices`

`\end{manswers}\CorrAnsButton{$\bigtriangleup$, Слон}`

`\end{questions}`

`\end{quiz}` `\ScoreField{testAbc}` `\eqButton\currQuiz\`

`Percent: \PercentField\currQuiz`

`\noindent` `Answers: \AnswerField\currQuiz`

XIV.5.2. Тесты с выбором и демонстрацией ответов

`\begin{quiz}*{qztestAbc}` Преамбула, некоторый текст.

`\begin{questions}` Максимальный балл за задание не указан.

`\item`

Вопрос с выбором нескольких верных вариантов ответа.

Не указано число баллов за ответ.

`\begin{manswers}[qz:testAbcB]{2}`

`\bChoices[4]`

`\Ans1  $2x+1$ \eAns` `\Ans0 Yes\eAns`

`\Ans1 No\eAns` `\Ans1 \mbox{Слово}\eAns`

`\eChoices`

`\end{manswers}\CorrAnsButton{ $2x+1$ ,No, Слово}`

`\end{questions}`

`\end{quiz}` `\ScoreField{testAbc}` `\eqButton\currQuiz\`

Percent: `\PercentField\currQuiz`

`\noindent` Answers: `\AnswerField\currQuiz`

XIV.5.2. Тесты с выбором и демонстрацией ответов

```
\begin{quiz}*{qztestAbc} Преамбула, некоторый текст.  
\begin{questions} К сожалению, при демонстрации ответов русские  
  \item  
    символы нормально не воспроизводятся.  
    Вопрос с выбором нескольких верных вариантов ответа.  
    Не указано число баллов за ответ.  
  \begin{manswers}[qz:testAbcB]{2}  
    \bChoices[4]  
      \Ans1  $2x+1$ \eAns          \Ans0 Yes\eAns  
      \Ans1 No\eAns              \Ans1 \mbox{Слово}\eAns  
    \eChoices  
  \end{manswers}\CorrAnsButton{ $2x+1$ ,No, Слово}  
\end{questions}  
\end{quiz} \ScoreField{testAbc} \eqButton\currQuiz\  
Percent: \PercentField\currQuiz  
  
\noindent Answers: \AnswerField\currQuiz
```

XIV.5.2. Тесты с выбором и демонстрацией ответов

```
\begin{quiz}*{qztestAbc} Преамбула, некоторый текст.  
\begin{questions}    Рассмотрим результат компиляции?  
  \item  
    Вопрос с выбором нескольких верных вариантов ответа.  
    Не указано число баллов за ответ.  
    \begin{manswers}[qz:testAbcB]{2}  
      \bChoices[4]  
        \Ans1  $2x+1$ \eAns          \Ans0 Yes\eAns  
        \Ans1 No\eAns              \Ans1 \mbox{Слово}\eAns  
      \eChoices  
    \end{manswers}\CorrAnsButton{ $2x+1$ ,No, Слово}  
  \end{questions}  
\end{quiz}  \ScoreField{testAbc}  \eqButton\currQuiz\  
Percent: \PercentField\currQuiz  
  
\noindent  Answers: \AnswerField\currQuiz
```

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}`

`\item`

Задания с вводом текста. Количество баллов не указано.`\newline`

`\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

`\CorrAnsButton{Слово}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd} \eqButton\currQuiz`

Points: `\PointsField\currQuiz\` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

```
\begin{quiz}*{qztestAbd} Преамбула к тесту, некоторый текст.
```

`\begin{questions}` Опция **[calc]** означает, что вводится алгебраическое
`\item` выражение.

Задания с вводом текста. Количество баллов не указано.\newline

`\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

\CorrAnsButton{Слово}

\end{questions}

\end{quiz}

\ScoreField{qztestAbd} \eqButton\currQuiz

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

Answers: \AnswerField\currQuiz

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}` Предусмотрено **6** вариантов ответа.

`\item`

Задания с вводом текста. Количество баллов не указано.`\newline`

`\RespBoxTxt{0}{0}[calc]` **{6}** `{-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

`\CorrAnsButton{Слово}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd}` `\eqButton\currQuiz`

Points: `\PointsField\currQuiz` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}` Предусмотрено **6** вариантов ответа.

`\item`

Задания с вводом текста. Количество баллов не указано.`\newline`

`\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

`\CorrAnsButton{Слово}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd} \eqButton\currQuiz`

Points: `\PointsField\currQuiz\` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}`

`\item`

Задания с вводом текста. Количество баллов не указано.`\newline`

`\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

`\CorrAnsButton{Слово}`

`\item\PTs{1}`

Завершать тест следует вопросом на выбор варианта.

`\begin{answers}[qz:testAbdC]{2}`

`\bChoices` `\Ans1 Да\еAns` `\Ans0 Нет\еAns` `\eChoices`

`\end{answers}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd} \eqButton\currQuiz`

Points: `\PointsField\currQuiz\` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}`

`\item`

Задания с вводом текста. Количество баллов не указано.`\newline`

`\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}`

`\CorrAnsButton{Слово}`

`\item\PTs{1}`

Завершать тест следует вопросом на выбор варианта.

`\begin{answers}[qz:testAbdC]{2}`

`\bChoices` `\Ans1 Да\Ans` `\Ans0 Нет\Ans` `\eChoices`

`\end{answers}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd} \eqButton\currQuiz`

Points: `\PointsField\currQuiz` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

```
\begin{quiz}*{qztestAbd} Преамбула к тесту, некоторый текст.
```

`\begin{questions}` В данном случае первое задание оценено в 1 балл (по
`\item` умолчанию).

Задания с вводом текста. Количество баллов не указано.\newline

\RespBoxTxt{0}{0}[calc] {6} {-2}{(-2)}{-2,}{-2,0}{-2.}{-2.0}

\CorrAnsButton{Слово}

\item\PTs{1}

Завершать тест следует вопросом на выбор варианта.

$$\backslash\mathrm{begin}\{\mathrm{answers}\}[qz:\mathrm{testAbdC}]\{2\}$$

\bChoices \Ans1 Да\Ans0 Нет\Ans \eChoices

\end{answers}

\end{questions}

\end{quiz}

\ScoreField{qztestAbd} \eqButton\currQuiz

Points: \PointsField\currQuiz\ Percent: \PercentField\currQuiz

Answers: \AnswerField\currQuiz

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}` А в данном случае первое задание оценено в 2 балла.
`\item\PTs{2}`

Задания с вводом текста. Указано количество баллов.`\newline`

`\RespBoxTxt{0}{0}[calc]{5}{3}{3.0}{3,}{3,0}{3.}`

`\CorrAnsButton{3}`

`\item\PTs{1}`

Завершать тест следует вопросом на выбор варианта.

`\begin{answers}[qz:testAbdC]{2}`

`\bChoices` `\Ans1 Да\еAns` `\Ans0 Нет\еAns` `\еChoices`

`\end{answers}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd}` `\eqButton\currQuiz`

Points: `\PointsField\currQuiz` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}` К сожалению, в настоящее время ответ в виде текста
`\item\PTs{2}` на кириллице не предусмотрен.

Задания с вводом текста. Указано количество баллов.`\newline`

`\RespBoxTxt{0}{0}[calc]{5}{3}{3.0}{3,}{3,0}{3.}`

`\CorrAnsButton{3}`

`\item\PTs{1}`

Завершать тест следует вопросом на выбор варианта.

`\begin{answers}[qz:testAbdC]{2}`

`\bChoices` `\Ans1 Да\еAns` `\Ans0 Нет\еAns` `\еChoices`

`\end{answers}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd}` `\eqButton\currQuiz`

Points: `\PointsField\currQuiz` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.5.3. Тесты с текстовым ответом

`\begin{quiz}*{qztestAbd}` Преамбула к тесту, некоторый текст.

`\begin{questions}` **Рассмотрим результат компиляции?**

`\item\PTs{2}`

Задания с вводом текста. Указано количество баллов.`\newline`

`\RespBoxTxt{0}{0}[calc]{5}{3}{3.0}{3,}{3,0}{3.}`

`\CorrAnsButton{3}`

`\item\PTs{1}`

Завершать тест следует вопросом на выбор варианта.

`\begin{answers}[qz:testAbdC]{2}`

`\bChoices` `\Ans1 Да\еAns` `\Ans0 Нет\еAns` `\еChoices`

`\end{answers}`

`\end{questions}`

`\end{quiz}`

`\ScoreField{qztestAbd} \eqButton\currQuiz`

Points: `\PointsField\currQuiz` Percent: `\PercentField\currQuiz`

Answers: `\AnswerField\currQuiz`

XIV.6. Инструменты преамбулы `testsample.tex`

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz,   \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz,   \CountVvx,\CountVvy,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,\CountVvy,\CountVvz,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,\CountVvy,\CountVvz,  
\CountSca,
```


XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz,  \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz,  \CountVvx,\CountVvy,\CountVvz,  
\CountSca,  \CountTmp,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,\CountVvy,\CountVvz,  
\CountSca, \CountTmp, \countvarnumb,
```

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,\CountVvy,\CountVvz,  
\CountSca, \CountTmp, \countvarnumb,
```

а также **счётчики** countSign,CountVariants,varntnumber,

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

```
\CountVax,\CountVay,\CountVaz, \CountVbx,\CountVby,\CountVbz,  
\CountVcx,\CountVcy,\CountVcz, \CountVvx,\CountVvy,\CountVvz,  
\CountSca, \CountTmp, \countvarnumb,
```

а также **счётчики** countSign,CountVariants,varntnumber,
countVAX,countVAY,countVAZ, countVBx,countVBy,countVBz,
countVCx,countVCy,countVCz, countVDx,countVDy,countVDz,
countPAX,countPAY,countPAZ, countPBx,countPBy,countPBz,
countPCx,countPCy,countPCz, countPDx,countPDy,countPDz.

XIV.6.1. testsample.tex: регистры и счётчики

В преамбуле testsample.tex определены **регистры**

`\CountVax, \CountVay, \CountVaz, \CountVbx, \CountVby, \CountVbz,`
`\CountVcx, \CountVcy, \CountVcz, \CountVvx, \CountVvy, \CountVvz,`
`\CountSca, \CountTmp, \countvarnumb,`

а также **счётчики** `countSign, CountVariants, varntnumber,`
`countVAX, countVAY, countVAZ, countVBx, countVBy, countVBz,`
`countVCx, countVCy, countVCz, countVDx, countVDy, countVDz,`
`countPAX, countPAY, countPAZ, countPBx, countPBy, countPBz,`
`countPCx, countPCy, countPCz, countPDx, countPDy, countPDz.`

В принципе, хотя и возможны осложнения, можно использовать **регистры**
`\CountSystCoordX, \CountSystCoordY,`

и **счётчики**

`MakExecisePdf, MakExecisePdfPrint, MakRemarkPdf, MakRemarkPdfPrint,`
`zaplatka, zaplatPrint, MyCommentPdfMake, MyCommentPdfPrint, setka,`
`PictureWidthAn, ParagraphWidthAn, CountEllipsAx, CountEllipsAy.`

XIV.6.2. testsample.tex: корректировка печати

В результате компиляции кода

```
$\left[2\mathbf{v}_i+3\mathbf{v}_j-3\mathbf{v}_k, \mathbf{v}_i+\mathbf{v}_k\right]=
```

```
\ComVectProd{2}{3}{-3}{1}{0}{1}
```

```
\the\CountVvx\mathbf{v}_i+\the\CountVvy\mathbf{v}_j+\the\CountVvz\mathbf{v}_k}.$
```

получили некорретную запись:

$$\left[2\overrightarrow{\mathbf{i}} + 3\overrightarrow{\mathbf{j}} - 3\overrightarrow{\mathbf{k}}, \overrightarrow{\mathbf{i}} + \overrightarrow{\mathbf{k}}\right] = 3\overrightarrow{\mathbf{i}} + -5\overrightarrow{\mathbf{j}} + -3\overrightarrow{\mathbf{k}}.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

В результате компиляции кода

```
$\left[2\mathrm{vectr}\{i\}+3\mathrm{vectr}\{j\}-3\mathrm{vectr}\{k\}, \mathrm{vectr}\{i\}+\mathrm{vectr}\{k\}\right]=
```

```
\ComVectProd{2}{3}{-3}{1}{0}{1}
```

```
\the\CountVvx\mathrm{vectr}\{i\}+\the\CountVvy\mathrm{vectr}\{j\}+\the\CountVvz\mathrm{vectr}\{k\}.$
```

получили некорретную запись:

$$\left[2\vec{\mathbf{i}} + 3\vec{\mathbf{j}} - 3\vec{\mathbf{k}}, \vec{\mathbf{i}} + \vec{\mathbf{k}}\right] = 3\vec{\mathbf{i}} + -5\vec{\mathbf{j}} + -3\vec{\mathbf{k}}.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

В команде `\ComPlusL{i}{k}{e}` и `\ComPlusN{k}{e}` считается, что k — это целочисленный коэффициент перед выражением e , i — целочисленный индикатор, определяющий характер печати выражения.

Мы использовали букву e как сокращение от expression.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{3}{x}\ComPlusL{1}{-4}{}`

будет

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{3}{x}\ComPlusL{1}{-4}{}`

будет

$$2x^2+3x-4.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{3}{x}\ComPlusL{1}{-4}{}`

будет

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{3}{x}\ComPlusL{1}{-4}{}`

будет

$+3x-4$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{1}{-4}{}`

будет

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{1}{-4}{}`

будет

-4 .

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{1}{0}{}`

будет

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{1}{0}{}`

будет

. (Т.е. ничего не напечатано).

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

будет

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

При $k = 0$ выражение $0e$ не печатается ни при каком значении i , кроме случая, когда $i = 3$ и все предыдущие двучлены в печатаемом выражении были нулевыми.

Например, результатом компиляции выражения

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

будет

0.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:
во-первых, при $k > 0$ знак $+$ не ставится;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:
во-первых, при $k > 0$ знак $+$ не ставится;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:
во-первых, при $k > 0$ знак $+$ не ставится;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$2x^2+2x+3$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:
во-первых, при $k > 0$ знак $+$ не ставится;

`\ComPlusL{0}{-2}{x^{2}}\ComPlusL{1}{-2}{x}\ComPlusL{1}{3}{}`
при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

`\ComPlusL{0}{-2}{x^{2}}\ComPlusL{1}{-2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$$-2x^2 - 2x + 3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

`\ComPlusL{0}{1}{x^{2}}\ComPlusL{1}{1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

`\ComPlusL{0}{1}{x^{2}}\ComPlusL{1}{1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

x^2+x+3 .

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{-1}{x^{2}}\ComPlusL{1}{-1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusLi{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 0$ применяется для печати первого двучлена $k \cdot e$:

во-первых, при $k > 0$ знак $+$ не ставится;

во-вторых, при $k = 1$ вместо $1e$ печатается e ;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{-1}{x^{2}}\ComPlusL{1}{-1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$-x^2 - x + 3$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$2x^2+2x+3$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

`\ComPlusL{0}{-2}{x^{2}}\ComPlusL{1}{-2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

`\ComPlusL{0}{-2}{x^{2}}\ComPlusL{1}{-2}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$$-2x^2 - 2x + 3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

`\ComPlusL{0}{1}{x^{2}}\ComPlusL{1}{1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

`\ComPlusL{0}{1}{x^{2}}\ComPlusL{1}{1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

x^2+x+3 .

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{-1}{x^{2}}\ComPlusL{1}{-1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{-1}{x^{2}}\ComPlusL{1}{-1}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$$-x^2 - x + 3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{0}{x^{2}}``\ComPlusL{1}{0}{x}``\ComPlusL{1}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 1$ применяется для печати двучлена $k \cdot e$, который может оказаться не первым ненулевым:

во-первых, при $k \geq 2$ вместо ke печатается $+ke$;

во-вторых, при $k = 1$ вместо $1e$ печатается $+e$;

в-третьих, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{1}{3}{}`

при компиляции даёт

$+3$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{3}{}`

при компиляции даёт

$$2x^2+2x+3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{-3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{-3}{}`

при компиляции даёт

$$2x^2+2x-3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{1}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{1}{}`

при компиляции даёт

$2x^2+2x+1$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{-1}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{-1}{}`

при компиляции даёт

$2x^2+2x-1$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ ничего не печатается.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ ничего не печатается.

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{0}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 2$ применяется для печати двучлена $k \cdot e$ при пустом e , т.е. печати k :

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ ничего не печатается.

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{2}{0}{}`

при компиляции даёт

$2x^2+2x$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{3}{}`

при компиляции даёт

$$2x^2+2x+3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{-3}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{-3}{}`

при компиляции даёт

$$2x^2+2x-3.$$

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{1}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{2}{x}\ComPlusL{3}{1}{}`

при компиляции даёт

$2x^2+2x+1$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается,

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды**

`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается,

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды**
`\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается,

`\ComPlusL{0}{2}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

при компиляции даёт

$2x^2$.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле `testsample.tex` введены **команды** `\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается, либо печатается 0, если все предыдущие коэффициенты были нулевыми.

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды** `\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается, либо печатается 0, если все предыдущие коэффициенты были нулевыми.

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

при компиляции даёт

XIV.6.2. testsample.tex: корректировка печати

В преамбуле testsample.tex введены **команды** `\ComPlusL{i}{k}{e}`, `\ComPlusV{x}{y}{z}` и `\ComPlusVb{x}{y}{z}`.

Значение $i = 3$ отличается от случая $i = 2$ тем, что в случае, когда все остальные слагаемые линейной комбинации имели нулевые коэффициенты, печатается 0 (применяется для последнего слагаемого в сумме):

во-первых, при $k \geq 1$ вместо k печатается $+k$;

во-вторых, при $k = -1$ вместо $-1e$ печатается $-e$;

в-третьих, при $k = 0$ либо ничего не печатается, либо печатается 0, если все предыдущие коэффициенты были нулевыми.

`\ComPlusL{0}{0}{x^{2}}\ComPlusL{1}{0}{x}\ComPlusL{3}{0}{}`

при компиляции даёт

0.

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

`\ComScalProd{u_x}{u_y}{u_z}{v_x}{v_y}{v_z}` применяется для вычисления скалярного произведения векторов $\vec{u} = u_x \vec{i} + u_y \vec{j} + u_z \vec{k}$ и $\vec{v} = v_x \vec{i} + v_y \vec{j} + v_z \vec{k}$, значение скалярного произведения присваивается **регистру** `\CountSca`.

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

`\ComScalProd{u_x}{u_y}{u_z}{v_x}{v_y}{v_z}` применяется для вычисления скалярного произведения векторов $\vec{u} = u_x \vec{i} + u_y \vec{j} + u_z \vec{k}$ и $\vec{v} = v_x \vec{i} + v_y \vec{j} + v_z \vec{k}$, значение скалярного произведения присваивается **регистру** `\CountSca`.

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

Например, результат компиляции кода

```

$$\left(2\mathbf{v}_i+3\mathbf{v}_j-3\mathbf{v}_k, \mathbf{v}_i+\mathbf{v}_k\right)=$$
\ComScalProd{2}{3}{-3}{1}{0}{1} \the\CountSca.$
```

будет таким:

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

Например, результат компиляции кода

```
$\left(2\mathrm{vectr}\{i\}+3\mathrm{vectr}\{j\}-3\mathrm{vectr}\{k\}, \mathrm{vectr}\{i\}+\mathrm{vectr}\{k\}\right)=  
\ComScalProd{2}{3}{-3}{1}{0}{1} \the\CountSca.$
```

будет таким:

$$\left(2\vec{i} + 3\vec{j} - 3\vec{k}, \vec{i} + \vec{k}\right) = -1.$$

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

`\ComVectProd{u_x}{u_y}{u_z}{v_x}{v_y}{v_z}` применяется для вычисления векторного произведения векторов $\vec{u} = u_x \vec{i} + u_y \vec{j} + u_z \vec{k}$ и $\vec{v} = v_x \vec{i} + v_y \vec{j} + v_z \vec{k}$, координаты векторного произведения присваиваются **регистрам** `\CountVvx`, `\CountVvy`, `\CountVvz`.

XIV.6.3. testsample.tex: функции на векторах

Например, результат компиляции кода

```

$$\left[2\mathbf{v}_i+3\mathbf{v}_j-3\mathbf{v}_k, \mathbf{v}_i+\mathbf{v}_k\right]=$$

$$\text{ComVectProd}\{2\}{3}\{-3\}\{1\}\{0\}\{1\}$$

$$\frac{\text{CountVvx}\mathbf{v}_i+\text{CountVvy}\mathbf{v}_j+\text{CountVvz}\mathbf{v}_k}{\text{CountV}}.$$

```

будет таким:

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

Например, результат компиляции кода

```
$\left[2\mathrm{vectr}\{i\}+3\mathrm{vectr}\{j\}-3\mathrm{vectr}\{k\}, \mathrm{vectr}\{i\}+\mathrm{vectr}\{k\}\right]=
```

```
\ComVectProd{2}{3}{-3}{1}{0}{1}
```

```
\the\CountVvx\mathrm{vectr}\{i\}+\the\CountVvy\mathrm{vectr}\{j\}+\the\CountVvz\mathrm{vectr}\{k\}.$
```

будет таким:

$$\left[2\vec{i} + 3\vec{j} - 3\vec{k}, \vec{i} + \vec{k}\right] = 3\vec{i} + -5\vec{j} + -3\vec{k}.$$

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

Например, результат компиляции кода

```
$\left[2\mathrm{vectr}\{i\}+3\mathrm{vectr}\{j\}-3\mathrm{vectr}\{k\}, \mathrm{vectr}\{i\}+\mathrm{vectr}\{k\}\right]=
```

```
\ComVectProd{2}{3}{-3}{1}{0}{1}
```

```
\the\CountVvx\mathrm{vectr}\{i\}+\the\CountVvy\mathrm{vectr}\{j\}+\the\CountVvz\mathrm{vectr}\{k\}.$
```

будет таким:

$$\left[2\vec{i} + 3\vec{j} - 3\vec{k}, \vec{i} + \vec{k}\right] = 3\vec{i} + -5\vec{j} + -3\vec{k}.$$

Запись
получилась

неграмотная!?

XIV.6.3. testsample.tex: функции на векторах

С помощью **перечисленных выше** счетчиков и регистров введены функции `\ComScalProd` и `\ComVectProd`.

Например, результат компиляции кода

```
$\left[2\backslash\mathrm{vectr}\{i\}+3\backslash\mathrm{vectr}\{j\}-3\backslash\mathrm{vectr}\{k\}, \backslash\mathrm{vectr}\{i\}+\backslash\mathrm{vectr}\{k\}\right]=
```

```
\ComVectProd{2}{3}{-3}{1}{0}{1}
```

```
\the\CountVvx\mathrm{vectr}\{i\}+\the\CountVvy\mathrm{vectr}\{j\}+\the\CountVvz\mathrm{vectr}\{k\}.$
```

будет таким:

$$\left[2\vec{i} + 3\vec{j} - 3\vec{k}, \vec{i} + \vec{k}\right] = 3\vec{i} + -5\vec{j} + -3\vec{k}.$$

Запись
получилась

неграмотная!?

Следует применить `\ComPlusV`.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}` — команда для вычисления детерминанта матрицы $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$. Значение детерминанта присваивается **регистру** `\CountSca`.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca}.$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}` — команда для вычисления детерминанта матрицы $\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$. Значение детерминанта присваивается **регистру** `\CountSca`.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca}.$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}` — команда для вычисления наибольшего общего делителя чисел m и n . Результат присваивается **регистру** `\CountSca`.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca}.$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \backslash\text{CountSca}.$

`\ComNumbOfPlaceA{m}{n}` — команда для вычисления числа сочетаний $A_n^m = (m+1)(m+2)\dots(n-1)n = \frac{n!}{m!}$. Результат будет в **регистре** `\CountSca`.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}` — команда для вычисления цифр числа n .

Здесь k — количество знаков числа, команда определяет команды

`\Namea, \Nameb, ..., \Nameбукваk`

$$n = \backslash\text{namea} \cdot 10^{k-1} + \backslash\text{nameb} \cdot 10^{k-2} + \backslash\text{namec} \cdot 10^{k-3} + \dots$$

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}` — команда для вычисления цифр числа n .

Здесь k — количество знаков числа, команда определяет команды

`\Namea, \Nameb, ..., \Nameбукваk`

$$n = \backslash\text{namea} \cdot 10^{k-1} + \backslash\text{nameb} \cdot 10^{k-2} + \backslash\text{namec} \cdot 10^{k-3} + \dots$$

`\ComCypherOfNumbA{Abc}{4}{2569}` приводит к появлению команд `\Abca`, равное 2, `\Abcb`, равное 5, `\Abcc`, равное 6, `\Abcd`, равное 9.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \text{\CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{4 \text{ знака}}$

— команда для сложения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealPlus({1}{2}{3}{4}{Abc}{Xx})` $1\text{\cdot}10^{\{2\}} + 3\text{\cdot}10^{\{4\}} = \text{\Abc}\text{\cdot}10^{\{\text{\Xx}\}}$ получим $1 \cdot 10^2 + 3 \cdot 10^4 = 301 \cdot 10^2$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}$

— команда для сложения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealPlus({6}{4}{3}{2}{Uvw}{Yy})` $6 \backslash\text{cdot} 10^{\{4\}} + 3 \backslash\text{cdot} 10^{\{2\}} =$
`\Abc \backslash\text{cdot} 10^{\{Xx\}}` получим $6 \cdot 10^4 + 3 \cdot 10^2 = 603 \cdot 10^2$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \text{\CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{4 \text{ знака}}$

— команда для сложения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealPlus({9}{6}{3}{3}{Pqr}{Zz})` $9 \cdot 10^6 + 3 \cdot 10^3 =$
`\Abc \cdot 10^{\Xx}` получим $9 \cdot 10^6 + 3 \cdot 10^3 = 9003 \cdot 10^3$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: НОД(m, n) $\mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}$.

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}$

— команда для умножения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealMult({1}{2}{3}{4}{Abc}{Xx})` $1 \backslash\text{cdot} 10^{\{2\}} \cdot 3 \backslash\text{cdot} 10^{\{4\}} =$
`\Abc \backslash\text{cdot} 10^{\{Xx\}}` получим $1 \cdot 10^2 \cdot 3 \cdot 10^4 = 3 \cdot 10^6$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: НОД(m, n) $\mapsto \text{\CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{4 \text{ знака}}$.

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{4 \text{ знака}}$

— команда для умножения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealMult({6}{4}{3}{2}{Uvw}{Yy})` `6\cdot10^{4} \cdot 3\cdot10^{2} =`
`\Abc\cdot10^{\Xx}` получим $6 \cdot 10^4 \cdot 3 \cdot 10^2 = 18 \cdot 10^6$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: НОД(m, n) $\mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}$.

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}$

— команда для умножения двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!).

Например, в результате компиляции:

`\ComRealMult({9}{6}{3}{3}{Pqr}{Zz})` $9 \backslash\text{cdot} 10^{\{6\}} \cdot 3 \backslash\text{cdot} 10^{\{3\}} =$
`\Abc \backslash\text{cdot} 10^{\{Xx\}}` получим $9 \cdot 10^6 \cdot 3 \cdot 10^3 = 27 \cdot 10^9$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca}.$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \text{\CountSca}.$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca}.$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\substack{\text{4 знака}}}$.

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\substack{\text{4 знака}}}$.

`\ComRealDivi{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b : (c \cdot 10^d) = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\substack{\text{4 знака}}}$

— команда для деления двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!). Например, в результате компиляции:

`\ComRealDivi({1}{2}{3}{4}{Abc}{Xx})` `\displaystyle\frac{1\cdotp10^2}{3\cdotp10^4} =`
`\Abc\cdotp10^{\Xx}` получим $\frac{1 \cdot 10^2}{3 \cdot 10^4} = 3333 \cdot 10^{-6}.$

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: НОД(m, n) $\mapsto \text{\CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}$.

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}$.

`\ComRealDivi{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b : (c \cdot 10^d) = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}$

— команда для деления двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!). Например, в результате компиляции:

`\ComRealDivi({6}{4}{3}{2}{Uvw}{Yy})` $\displaystyle \frac{6 \cdot 10^4}{3 \cdot 10^2} =$
`\Abc \cdotp 10^{\Xx}` получим $\frac{6 \cdot 10^4}{3 \cdot 10^2} = 2000 \cdot 10^{-1}$.

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \text{\CountSca}.$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \text{\CountSca}.$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \text{\CountSca}.$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}.$

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}.$

`\ComRealDivi{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b : (c \cdot 10^d) = \underbrace{\text{\nameA} \cdot 10^{\text{\nameB}}}_{\text{4 знака}}$

— команда для деления двух чисел в «научной форме» с четырьмя значащими цифрами (не округление!). Например, в результате компиляции:

`\ComRealDivi({9}{6}{3}{3}{Pqr}{Zz})` `\displaystyle\frac{9\cdotp10^6}{3\cdotp10^3}` =
`\Abc\cdotp10^{\Xx}` получим $\frac{9 \cdot 10^6}{3 \cdot 10^3} = 3000 \cdot 10^0.$

XIV.6.4. testsample.tex: вычисление некоторых целочисленных функций

`\ComDetTwo{a}{b}{c}{d}`: $\begin{vmatrix} a & b \\ c & d \end{vmatrix} \mapsto \backslash\text{CountSca.}$

`\ComDetThree{a11}{a12}{a13}{a21}{a22}{a23}{a31}{a32}{a33}`.

`\NOD{m}{n}`: $\text{НОД}(m, n) \mapsto \backslash\text{CountSca.}$

`\ComNumbOfPlaceA{m}{n}`: $A_n^m \mapsto \backslash\text{CountSca.}$

`\ComCypherOfNumbA{Name}{k}{n}`: разложение по цифрам k -значного числа n .

`\ComRealPlus{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b + c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}.$

`\ComRealMult{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b \cdot c \cdot 10^d = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}.$

`\ComRealDivi{a}{b}{c}{d}{nameA}{nameB}`: $a \cdot 10^b : (c \cdot 10^d) = \underbrace{\backslash\text{nameA} \cdot 10^{\backslash\text{nameB}}}_{4 \text{ знака}}.$

XIV.6.5. `testsample.tex`: генерирование «псевдослучайных» значений

Для генерирования тестов необходимо получить набор некоторых «входных параметров».

При этом использование традиционных генераторов «псевдослучайных чисел» нерационально, поскольку желательно, чтобы при повторной генерации заданий для группы или класса получались бы те же значения, что и ранее, поскольку обычно необходимость в повторной генерации обусловлена или появлением новых обучаемых, или обнаружением ошибок в коде и т.п.

При этом желательно, чтобы остальные задания или задания других студентов не изменялись.

Поэтому значения вычисляются как комбинация значений параметров, уникальных для каждой группы или класса.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero{n}{a}{b}` — команда для преобразования целого числа n в число $n \in \{b; b + 1; \dots; b + a - 1\}$, где b — неотрицательное целое число, $a \in \mathbb{N}$.

Результат присваивается **регистру** `\CountTmp`.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero{n}{a}{b}`: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero{n}{a}{b}` — команда для преобразования целого числа n в число

$m \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

Результат присваивается **регистру** `\CountTmp`.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

`\MakeParamPositiv` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$ — команда для генерирования значения натурального числа из диапазона $\{p; p+1; \dots; q\}$.

Результат присваивается **регистру** `\CountTmp`.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

`\MakeParamPositiv` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{p; p+1; \dots; q\}$.

`\MakeParamAnysign` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$ — коман-

да для генерирования значения натурального числа из диапазона $\{-q; -q+1; \dots; -p-1\} \cup \{p+1; p+2; \dots; q\}$.

Результат присваивается **регистру** `\CountTmp`.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

`\MakeParamPositiv` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{p; p+1; \dots; q\}$.

`\MakeParamAnysign` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{-q; -q+1; \dots; -p-1\} \cup \{p+1; p+2; \dots; q\}$.

`\GenerateNonZeroParamsZ` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{a_8\}\{a_9\}$ —

команда для генерирования ненулевых значений **регистров**

`\CountVax`, `\CountVay`, `\CountVaz`, `\CountVbx`, `\CountVby`, `\CountVbz`,
`\CountVcx`, `\CountVcy`, `\CountVcz`, `\CountVvx`, `\CountVvy`, `\CountVvz`.

Эти значения определяются с помощью параметров

$a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

`\MakeParamPositiv` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{p; p+1; \dots; q\}$.

`\MakeParamAnysign` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{-q; -q+1; \dots; -p-1\} \cup \{p+1; p+2; \dots; q\}$.

`\GenerateNonZeroParamsZ` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{a_8\}\{a_9\}$ —

команда для генерирования ненулевых значений **регистров**

`\CountVax`, `\CountVay`, `\CountVaz`, `\CountVbx`, `\CountVby`, `\CountVbz`,
`\CountVcx`, `\CountVcy`, `\CountVcz`, `\CountVvx`, `\CountVvy`, `\CountVvz`.

Гарантированно несовпадают `\CountVvx`, `\CountVvy`, `\CountVvz`:

`\CountVvx` $\neq$ `\CountVvy` $\neq$ `\CountVvz` $\neq$ `\CountVvx`.

XIV.6.5. testsample.tex: генерирование «псевдослучайных» значений

`\ComModbyNumbSingleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{b; b+1; \dots; b+a-1\}$.

`\ComModbyNumbDoubleNonZero` $\{n\}\{a\}\{b\}$: $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{0\}$,

$n \mapsto \backslash\text{CountTmp} \in \{-b-a, -b-a+1, \dots; -b-1\} \cup \{b+1, b+2; \dots; b+a\}$.

`\MakeParamPositiv` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{p; p+1; \dots; q\}$.

`\MakeParamAnysign` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{p\}\{q\}$:

$\backslash\text{CountTmp} \in \{-q; -q+1; \dots; -p-1\} \cup \{p+1; p+2; \dots; q\}$.

`\GenerateNonZeroParamsZ` $\{a_1\}\{a_2\}\{a_3\}\{a_4\}\{a_5\}\{a_6\}\{a_7\}\{a_8\}\{a_9\}$:

$\backslash\text{CountVax}, \backslash\text{CountVay}, \backslash\text{CountVaz}, \backslash\text{CountVbx}, \backslash\text{CountVby}, \backslash\text{CountVbz},$

$\backslash\text{CountVcx}, \backslash\text{CountVcy}, \backslash\text{CountVcz}, \backslash\text{CountVvx} \neq \backslash\text{CountVvy} \neq \backslash\text{CountVvz} \neq \backslash\text{CountVvx}.$

XIV.6.6. `testsample.tex`: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в $\text{\TeX}$ проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\SummaRealA{p}{k}{q}{m}{X}{Y}`:

$p \cdot 10^{-k} + q \cdot 10^{-m} = \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\SummaRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\SummaRealA`.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\SummaRealA{p}{k}{q}{m}{X}{Y}`:

$p \cdot 10^{-k} + q \cdot 10^{-m} = \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\SummaRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\SummaRealA`.

Например, в результате выполнения `\SummaRealA{1234}{1}{567}{2}{Fr}{Deg}` будут определены: команда `\Fr`, печатающая число 12907 и команда `\Deg`, печатающая число 2, где значения этих команд определяются равенством $123,4 + 5,67 = 129,07$.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\MultiRealA{p}{k}{q}{m}{X}{Y}{n}`, $p \cdot 10^{-k} \cdot q \cdot 10^{-m} \approx \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\MultiRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\MultiRealA`.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\MultiRealA{p}{k}{q}{m}{X}{Y}{n}`, $p \cdot 10^{-k} \cdot q \cdot 10^{-m} \approx \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\MultiRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\MultiRealA`.

Например, в результате выполнения `\MultiRealA{1234}{2}{5678}{3}{M}{D}{2}` будут определены: команда `\M`, печатающая число 12907 и команда `\D`, печатающая число 2, где значения этих команд определяются равенством $12,34 \cdot 5,678 = 70,06652 \approx 70,07$. Таким образом, команда `\D` печатает значение седьмого аргумента команды `\MultiRealA`, т.е. n , причем результат умножения округляется до `\D`-й (т.е. до n -й) цифры мантиссы.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

`\SummaRealA{p}{k}{q}{m}{X}{Y}`, `\MultiRealA{p}{k}{q}{m}{X}{Y}{n}`,
`\DivideRealA{p}{k}{q}{m}{X}{Y}{n}`, `\PrintRealA{p}{k}`,
`\PrintRealB{p}{k}`.

`\DivideRealA{p}{k}{q}{m}{X}{Y}{n}`, $\frac{p \cdot 10^{-k}}{q \cdot 10^{-m}} \approx \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\DivideRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\DivideRealA`.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в `TeX` проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

`\SummaRealA{p}{k}{q}{m}{X}{Y}`, `\MultiRealA{p}{k}{q}{m}{X}{Y}{n}`,
`\DivideRealA{p}{k}{q}{m}{X}{Y}{n}`, `\PrintRealA{p}{k}`,
`\PrintRealB{p}{k}`.

`\DivideRealA{p}{k}{q}{m}{X}{Y}{n}`, $\frac{p \cdot 10^{-k}}{q \cdot 10^{-m}} \approx \backslash X \cdot 10^{-\backslash Y}$, где `\X` и `\Y` — команды, определяемыми во время работы `\DivideRealA`, с идентификаторами, указанными в качестве пятого и шестого аргументов этой команды `\DivideRealA`.

В результате выполнения `\DivideRealA{1243}{2}{56789}{3}{M}{D}{4}` будут определены: команда `\M`, печатающая число 2189 и команда `\D`, печатающая число 4, где значения этих команд определяются равенством

$\frac{12,43}{56,789} = 0,21888 \approx 0,2189$. Таким образом, команда `\D` печатает значение седьмого аргумента команды `\DivideRealA`, т.е. n , причем результат умножения округляется до `\D`-й (т.е. до n -й) цифры мантиссы.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в $\text{\TeX}$ проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\PrintDivideRealA{p}{k}` — команда, печатающая в виде позиционной десятичной дроби число $p \cdot 10^{-k}$. Отметим, что сам $\text{\TeX}$ полученную строку числом не считает!

В качестве разделителя мантиссы и целой части числа используется точка.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в $\text{\TeX}$ проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\PrintDivideRealA{p}{k}` — команда, печатающая в виде позиционной десятичной дроби число $p \cdot 10^{-k}$. Отметим, что сам $\text{\TeX}$ полученную строку числом не считает!

В качестве разделителя мантиссы и целой части числа используется точка.

Например, компиляция команды `\PrintDivideRealA{987654321}{4}` приведет к тому, что будет напечатано выражение 98765.4321, которое, как известно, равно $987654321 \cdot 10^{-4}$.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в $\text{\TeX}$ проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\PrintDivideRealB{p}{k}`. команда, печатающая в виде позиционной десятичной дроби число $p \cdot 10^{-k}$. Отметим, что сам $\text{\TeX}$ полученную строку числом не считает!

В качестве разделителя мантиссы и целой части числа используется запятая.

XIV.6.6. testsample.tex: операции с десятичными дробями

Дональд Кнут предусмотрел в $\text{\TeX}$ проведение вычислений только с целыми числами. Поэтому в `testsample.tex` мы ввели несколько инструментов для проведения операций с десятичными дробями:

```
\SummaRealA{p}{k}{q}{m}{X}{Y}, \MultiRealA{p}{k}{q}{m}{X}{Y}{n},  
\DivideRealA{p}{k}{q}{m}{X}{Y}{n}, \PrintRealA{p}{k},  
\PrintRealB{p}{k}.
```

`\PrintDivideRealB{p}{k}`. команда, печатающая в виде позиционной десятичной дроби число $p \cdot 10^{-k}$. Отметим, что сам $\text{\TeX}$ полученную строку числом не считает!

В качестве разделителя мантиссы и целой части числа используется запятая.

Например, компиляция команды `\PrintDivideRealA{987654321}{4}` приведет к тому, что будет напечатано выражение 98765,4321, которое, как известно, равно $987654321 \cdot 10^{-4}$.

XIV.6.7. testsample: окружение mathGrp

Пусть `\Abs` определено как «два», значение **счетчика** `countVCz` равно 2, значение **регистра** `\CountVax` равно 7, а `\CountVay` — 4.

Тогда результатом компиляции фрагмента файла `STest...`:

```
\begin{mathGrp}\PTs*{1}
```

```
Число «два» пишется как \coeff{2}.\newline
```

```
$\the\CountVax= 3+ \coeff{\the\CountVay}}.$
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\arabic{countVCz},\the\CountVay}
```

будет тестовое задание

XIV.6.7. testsample: окружение mathGrp

Пусть `\Abs` определено как «два», значение **счетчика** `countVCz` равно 2, значение **регистра** `\CountVax` равно 7, а `\CountVay` — 4.

Тогда результатом компиляции фрагмента файла `STest...`:

```
\begin{mathGrp}\PTs*{1}
```

Число «два» пишется как `\coeff{2}.\newline`

`$\the\CountVax= 3+ \coeff{\the\CountVay}}.$`

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\arabic{countVCz},\the\CountVay}
```

будет тестовое задание.

Число «два» пишется как .

$7 = 3 +$.

XIV.6.7. testsample: окружение mathGrp

Пусть `\Abs` определено как «два», значение **счетчика** `countVCz` равно 2, значение **регистра** `\CountVax` равно 7, а `\CountVay` — 4.

Тогда результатом компиляции фрагмента файла `STest...`:

```
\begin{mathGrp}\PTs*{1}
```

Число «два» пишется как `\coeff{2}.\newline`

`$\the\CountVax= 3+ \coeff{\the\CountVay}}.$`

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\arabic{countVCz},\the\CountVay}
```

будет тестовое задание.

В команде `\CorrAnsButtonGrp` разработчику теста необходимо последовательно перечислить значения, указанные в `\coeff`.

XIV.6.7. testsample: окружение mathGrp

Пусть `\Abs` определено как «два», значение **счетчика** `countVCz` равно 2, значение **регистра** `\CountVax` равно 7, а `\CountVay` — 4.

Тогда результатом компиляции фрагмента файла `STest...`:

```
\begin{mathGrp}\PTs*{1}
```

```
Число «два» пишется как \coeff{2}.\newline
```

```
$\the\CountVax= 3+ \coeff{\the\CountVay}}.$
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\arabic{countVCz},\the\CountVay}
```

будет тестовое задание.

В команде `\CorrAnsButtonGrp` разработчику теста необходимо последовательно перечислить значения, указанные в `\coeff`.

Эти значения будут выводиться в поле коррекции, которое будет появляться в тесте,

если в файле `IDZfinish.tex` была строка `\def\PrintCorr{1}`.

XIV.6.7. testsample: окружение mathGrp

Пусть `\Abs` определено как «два», значение **счетчика** `countVCz` равно 2, значение **регистра** `\CountVax` равно 7, а `\CountVay` — 4.

Тогда результатом компиляции фрагмента файла `STest...`:

```
\begin{mathGrp}\PTs*{1}
```

```
Число «два» пишется как \coeff{2}.\newline
```

```
$\the\CountVax= 3+ \coeff{\the\CountVay}}.$
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\arabic{countVCz},\the\CountVay}
```

будет тестовое задание.

В команде `\CorrAnsButtonGrp` разработчику теста необходимо последовательно перечислить значения, указанные в `\coeff`.

Поле коррекции не выводится,

если в файле `IDZfinish.tex` была строка `\def\PrintCorr{0}`.

XIV.6.8. testsample.tex: окружение `manswers`

Окружение «`manswers`» формирует в тесте блока вариантов ответов с множественными вариантами верных ответов.

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}
```

Окружение «`manswers`» формирует в тесте блока вариантов ответов с множественными вариантами верных ответов.

```
\end{manswers}
```

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}
```

Окружение «`manswers`» формирует в тесте блока вариантов ответов с множественными вариантами верных ответов.

```
\end{manswers}
```

XIV.6.8. testsample.tex: окружение `answers`

```
\begin{answers}{m}
```

Здесь число *m* означает количество колонок, отводимых для вариантов ответа.

```
\end{answers}
```


XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}
```

Здесь число m означает количество колонок, отводимых для вариантов ответа.

```
\end{manswers}
```

Предполагается, что если вместо m поставлено число 2, компиляция приведет к результату типа:

☐ вариант ₁	☐ вариант ₂
☐ вариант ₃	☐ вариант ₄
☐ вариант ₅	☐ вариант ₆

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}
```

Здесь число m означает количество колонок, отводимых для вариантов ответа.

```
\end{manswers}
```

Предполагается, что если вместо m поставлено число 2, компиляция приведет к результату типа:

☐ вариант ₁	☐ вариант ₂	☐ вариант ₃
☐ вариант ₄	☐ вариант ₅	☐ вариант ₆
☐ вариант ₇	☐ вариант ₈	☐ вариант ₉

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}  
  \bChoices
```

«Скобки» `\bChoices...` `\eChoices` «обрамляют» список вариантов ответа.

```
  \eChoices  
\end{manswers}
```

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}  
  \bChoices[nCols=2]
```

Параметр `nCols` означает реально заданное число колонок с вариантами ответов. Этот параметр необязательный и его значение этого параметра не должно превосходить m .

```
  \eChoices  
\end{manswers}
```

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}
```

«Скобки» `\Ans...` `\eAns` «обрамляют»

каждый вариант ответа.

```
\bChoices[nCols=2]
```

```
\Ans          ... \eAns
```

```
\Ans          ... \eAns
```

```
...
```

```
\Ans          ... \eAns
```

```
\eChoices
```

```
\end{manswers}
```

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}  
  \bChoices[nCols=2]  
    \Ans      ... \eAns  
    \Ans      ... \eAns  
    ...  
    \Ans      ... \eAns  
  \eChoices  
\end{manswers}
```

«Скобки» `\Ans...` `\eAns` «обрамляют»
каждый вариант ответа.

В этом примере вместо `...` нахо-
дится `вариант$_{1}$`, соответственно,
`вариант$_{2}$` и т.д.

Предполагается, что если вместо m поставлено число 2, компиляция приведет к результату типа:

☐ вариант ₁	☐ вариант ₂
☐ вариант ₃	☐ вариант ₄
☐ вариант ₅	☐ вариант ₆

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}  
  \bChoices[nCols=2]  
    \Ans      ... \eAns  
    \Ans      ... \eAns  
    ...  
    \Ans      ... \eAns  
  \eChoices  
\end{manswers}
```

«Скобки» `\Ans...` `\eAns` «обрамляют»
каждый вариант ответа.

В этом примере вместо ... находится вариант $_{1}$, соответственно, вариант $_{2}$ и т.д.

Предполагается, что если вместо m поставлено число 2, компиляция приведет к результату типа:

☐ вариант ₁	☐ вариант ₂	☐ вариант ₃
☐ вариант ₄	☐ вариант ₅	☐ вариант ₆
☐ вариант ₇	☐ вариант ₈	☐ вариант ₉

XIV.6.8. testsample.tex: окружение manswers

```
\begin{manswers}{m}
  \bChoices[nCols=2]
    \Ans      {a1} ... \eAns
    \Ans      {a2} ... \eAns
    ...
    \Ans      {ak} ... \eAns
  \eChoices
\end{manswers}
```

Вместо $a_1, a_2, \dots$ следует поставить либо число 0, если этот вариант ответа неверен либо 1, если этот вариант ответа является верным.

XIV.6.8. testsample.tex: окружение `manswers`

```
\begin{manswers}{m}
  \bChoices[nCols=2]
    \Ans[b_1]{a_1} ... \eAns
    \Ans[b_2]{a_2} ... \eAns
    ...
    \Ans[b_k]{a_k} ... \eAns
  \eChoices
\end{manswers}
```

Вместо необязательных параметров $b_1, b_2, \dots$ можно поставить количество баллов, начисляемых за выбор данного варианта ответа. Ясно, что значение этого параметра тестируемому не предъявляется.

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{ }( ){ }{ }{ }$}`

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}( ) { } { } { } { }`

F — формула в обозначениях mprost (практически совпадает с JavaScript), например,

`cos(x)` вместо $\cos x$, т.е., в обозначениях T_EX, `\cos{x}`,

`x^(2*x-1)` вместо x^{2x-1} , т.е., в обозначениях T_EX, `x^{2x-1}` и др.

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}(V){ }{ }{ }$}`

V — список переменных, которым в процессе сравнения введенного тестируемым выражения с эталонным придаются определенные значения, например, (V) надо заменить на (x) , на (xy) , на (xyz) , на (abc) .

Можно уточнить тип переменной, например, если V заменить на $(r:x, r:y, i:z)$, то это означает, что при сравнении значений выражения, введенного тестируемым, с эталонным выражением, вместо x и y будут подставляться действительные числа (на практике рациональные), а вместо z — целые.

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}(V){ }{ }{Set}$}`

V — список переменных, которым в процессе сравнения введенного тестируемым выражения с эталонным придаются определенные значения, например, (V) надо заменить на (x) , на (xy) , на (xyz) , на (abc) .

Set — описание множества, на котором проводится сравнение формулы, введенной тестируемым, с эталонной формулой.

Вместо Set надо ввести, например, $[0,1]$ или $[2,3] \times [3,5]$, или $[-1,2] \times [1,4] \times [1,5]$ и т.п., где «крестик» между обозначениями отрезков — это латинская буква «икс».

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}(V){n}{ }{Set}$}`

V — список переменных, которым в процессе сравнения введенного тестируемым выражения с эталонным придаются определенные значения, например, (V) надо заменить на (x) , на (xy) , на (xyz) , на (abc) .

Set : например, $[0, 1]$, $[2, 3] \times [3, 5]$ или $[-1, 2] \times [1, 4] \times [1, 5]$ и т.п.

n — количество точек (видимо, на каждом из отрезков, указанных в Set), в которых вычисляется значение выражения, введенного тестируемым, и эталонного выражения, для того, чтобы сделать вывод о том, совпадают ли они.

Обычно вместо n ставят 4 или 5, этого, как правило, достаточно.

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}(V){n}{\varepsilon}{Set}$}`

V — список переменных, которым в процессе сравнения введенного тестируемым выражения с эталонным придаются определенные значения, например, (V) надо заменить на (x) , на (xy) , на (xyz) , на (abc) .

Set : например, $[0, 1]$, $[2, 3] \times [3, 5]$ или $[-1, 2] \times [1, 4] \times [1, 5]$ и т.п.

n — количество точек (видимо, на каждом из отрезков, указанных в Set), в которых вычисляется значение выражения, введенного тестируемым, и эталонного выражения, для того, чтобы сделать вывод о том, совпадают ли они.

ε — точность, с которой в каждой из n точек сравниваются выражения: введенное тестируемым и эталонное.

Обычно достаточно вместо ε поставить .0001

XIV.6.9. testsample: окружение RespBoxMath

`\mbox{$...=`

`\RespBoxMath{F}(V){n}{\varepsilon}{Set}\$}`

Например, компиляция фрагмента

`$f(x,y)=\RespBoxMath{y*atan(x)}(xy){4}{.00001}{[0,1]x[-1,1]}$`

приводит к результату вида

$$f(x, y) = \boxed{},$$

где в поле для ввода в качестве правильного ответа следует ввести `*atan(x)`, что означает $y \arctg x$.

XIV.7. Макрокоманды для задания функций в StartDiff.inp

XIV.7.1. StartDiff.inp: \FncA{...}{...}{...}{...}

\FncA{argum_{TEX}}{agrum_{Java}}{n}{PartOfName} — команда для выбора конкретной функции с указанием ее производной.

Выбор функции определяется значением #3, точнее, после «нормировки», значением счётчика \countvarnumb

Значения x для проверки совпадения с $\beta(\alpha(x))$

		0	1	2	3	4	5	6	7
$\alpha \backslash \beta$		$\sqrt[3]{\#1}$	$\sin \#1$	$\cos \#1$	$\sqrt{\#1}$	$\operatorname{tg} \#1$	$\operatorname{arctg} \#1$	$\operatorname{arcsin} \#1$	$\operatorname{arccos} \#1$
0	$\sqrt[3]{\#1}$	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0; 0.48]	[0; 1]	[0, 0.59]	[0.16, 1]
1	$\sin \#1$	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0, 0.9]	[0; 1]	[0; 1]	[0.58, 1]
2	$\cos \#1$	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0.67, 1]	[0; 1]	[0.58, 1]	[0; 1]
3	$\sqrt{\#1}$	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0, 0.61]	[0; 1]	[0, 0.7]	[0.3, 1]
4	$\operatorname{tg} \#1$	[0, 0.78]	[0; 1]	[0; 1]	[0, 0.78]	[0, 0.66]	[0; 1]	[0, 0.69]	[0.5, 0.78]
5	$\operatorname{arctg} \#1$	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0; 1]	[0.6; 1]
6	$\operatorname{arcsin} \#1$	[0, 0.84]	[0; 1]	[0; 1]	[0; 0.84]	[0; 0.7]	[0, 0.99]	[0, 0.74]	[0.52, 0.84]
7	$\operatorname{arccos} \#1$	[0.55; 1]	[0; 1]	[0; 1]	[0.55; 1]	[0.71; 1]	[0.02, 1]	[0.67, 1]	[0.55, 0.85]

XIV.7.2. StartDiff.inp: \FncB{\dots}{\dots}{\dots}{\dots}

При значениях #3, точнее, после «нормировки», в счётчике \countvarnumb, выбираются следующие функции:

$$\backslash\text{countvarnumb}=0: \quad \#1^{\#5} = x^{\#5};$$

$$\backslash\text{countvarnumb}=1: \quad \log_{\#5} \#1 = \log_{\#5} x;$$

$$\backslash\text{countvarnumb}=2: \quad \#5^{\#1} = \#5^x.$$

XIV.7.3. StartDiff.inp: $\backslash\text{FncC}\{\dots\}\{\dots\}\{\dots\}\{\dots\}$

При значениях #3, точнее, после «нормировки», в счётчике $\backslash\text{countvarnumb}$, выбираются следующие функции:

$\backslash\text{countvarnumb}=0$: $\sin \#1$;

$\backslash\text{countvarnumb}=1$: $\cos \#1$;

$\backslash\text{countvarnumb}=2$: $e^{\#1}$.

XIV.7.4. StartDiff.inp: \FncD{...}{...}{...}{...}

При значениях #3, точнее, после «нормировки», в счётчике \countvarnumb, выбираются следующие функции:

\countvarnumb=0: $\ln \#1$;

\countvarnumb=1: $\arcsin \#1$;

\countvarnumb=2: $\operatorname{arctg} \#1$;

\countvarnumb=3: $\arccos \#1$.

XIV.8. Система разработки средств контроля

Разработанная нами система, созданная на базе профессиональной издательской системы L^AT_EX, создаёт документы в формате pdf со включением скриптов, в частности, кода Java.

XIV.8.1. Преимущества pdf

Преимущества pdf:

- богатые возможности для интерактивности (гипермишени и гиперссылки, возможность анимации, вставка кода JavaScript, вызов внешних приложений и др.);
- реализация просмотра на всех распространённых аппаратных и программных платформах;
- оптимальная защищённость от несанкционированных изменений;
- аутентичное воспроизведение;
- богатый набор средств разработки pdf-документов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

Рассмотрим структуру и предназначение файлов в системе подготовки электронных тестов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

001.tex

002.tex

.....

00*n*.tex

.....

035.tex

.....

Изначально используются эти файлы, они одинаковы для всех тестов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

`001.tex` $\xrightarrow{\text{pdflatex}}$ `001.pdf`

`002.tex`

.....

`00n.tex`

.....

`035.tex`

.....

Запускается пакетная компиляция всех этих файлов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

`001.tex` $\xrightarrow{\text{pdflatex}}$ `001.pdf`

`002.tex` $\xrightarrow{\text{pdflatex}}$ `002.pdf`

.....

`00n.tex`

.....

`035.tex`

.....

Запускается пакетная компиляция всех этих файлов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

`001.tex` $\xrightarrow{\text{pdflatex}}$ `001.pdf`

`002.tex` $\xrightarrow{\text{pdflatex}}$ `002.pdf`

.....

`00n.tex` $\xrightarrow{\text{pdflatex}}$ `00n.pdf`

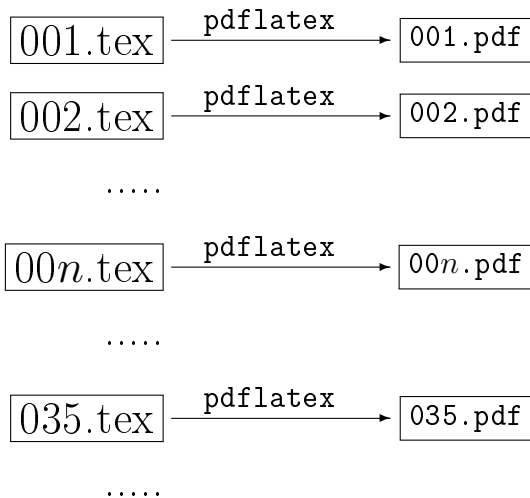
.....

`035.tex`

.....

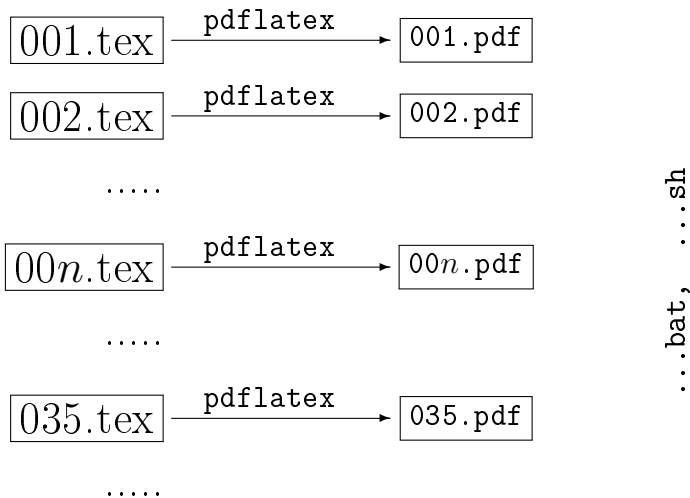
Запускается пакетная компиляция всех этих файлов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



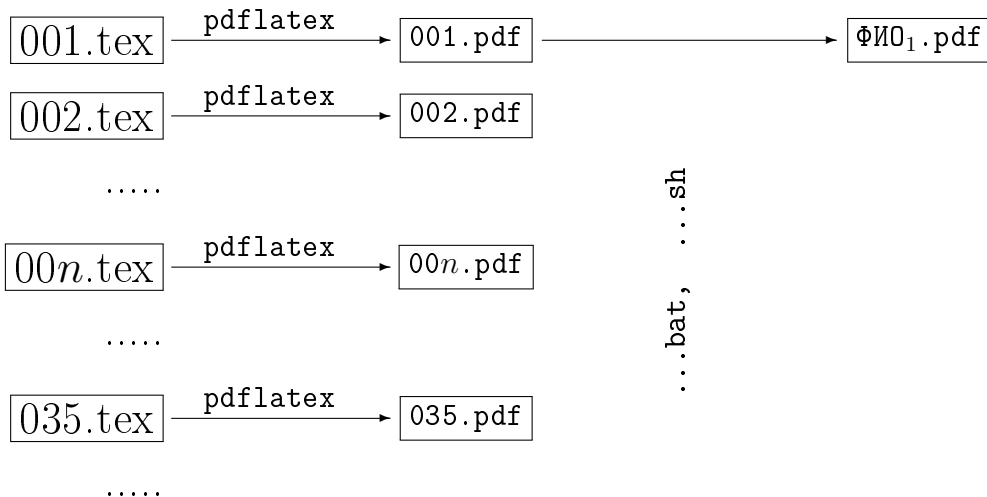
Запускается пакетная компиляция всех этих файлов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



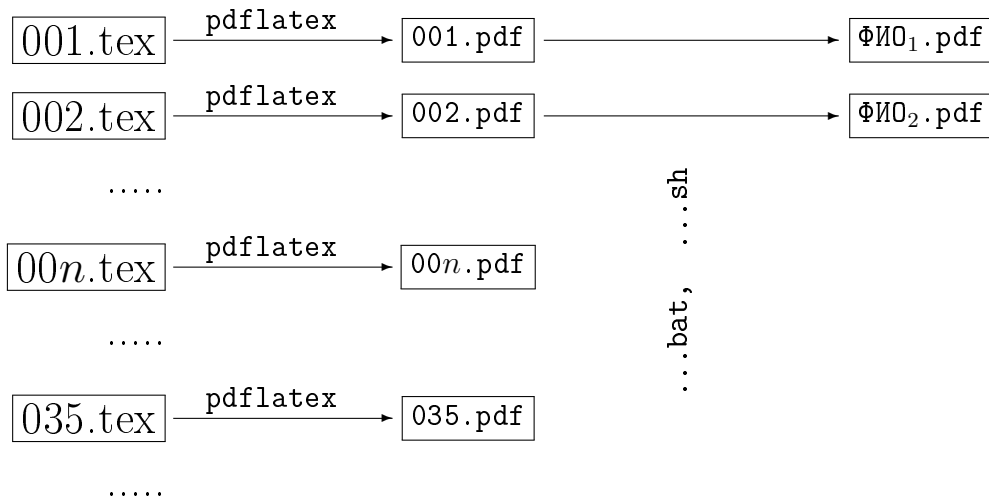
В этом же пакетном файле запускается копирование файлов `00n.pdf` в файлы, именами которых являются ФИО студентов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



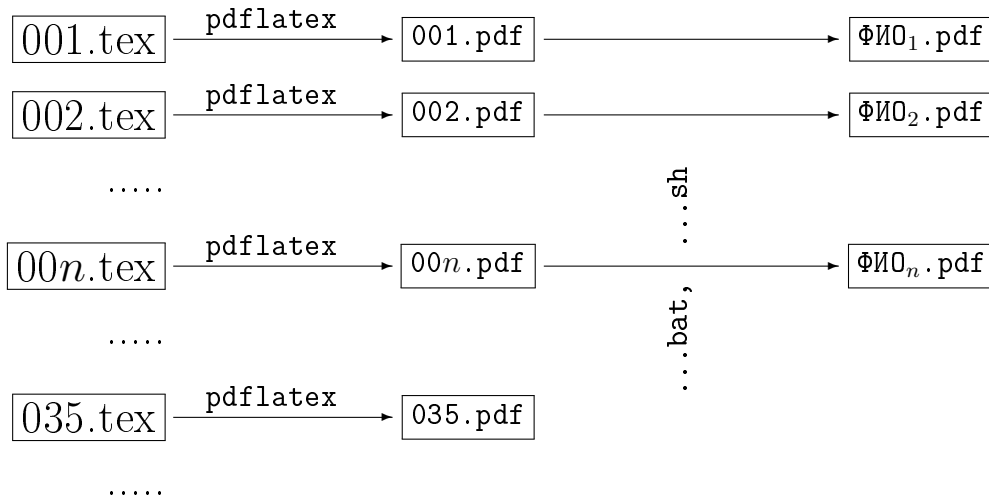
В этом же пакетном файле запускается копирование файлов `00n.pdf` в файлы, именами которых являются ФИО студентов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



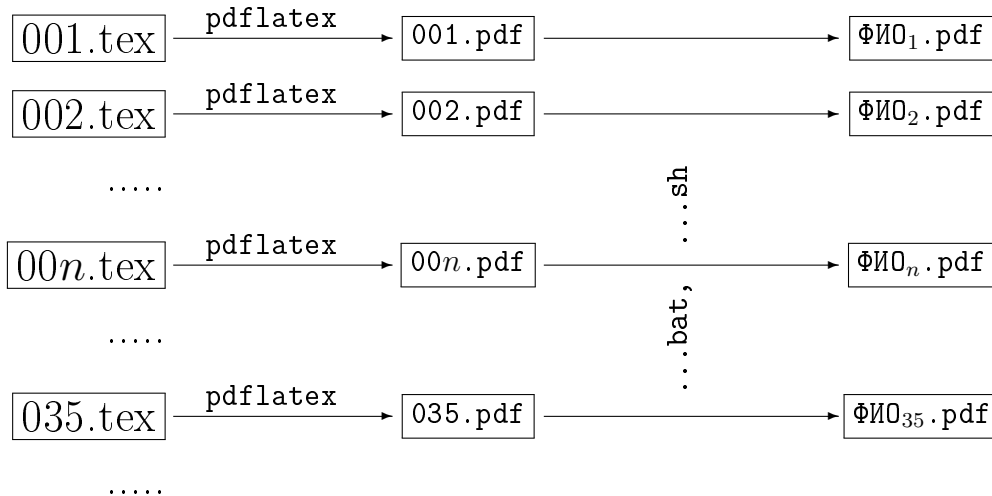
В этом же пакетном файле запускается копирование файлов $00n.pdf$ в файлы, именами которых являются ФИО студентов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



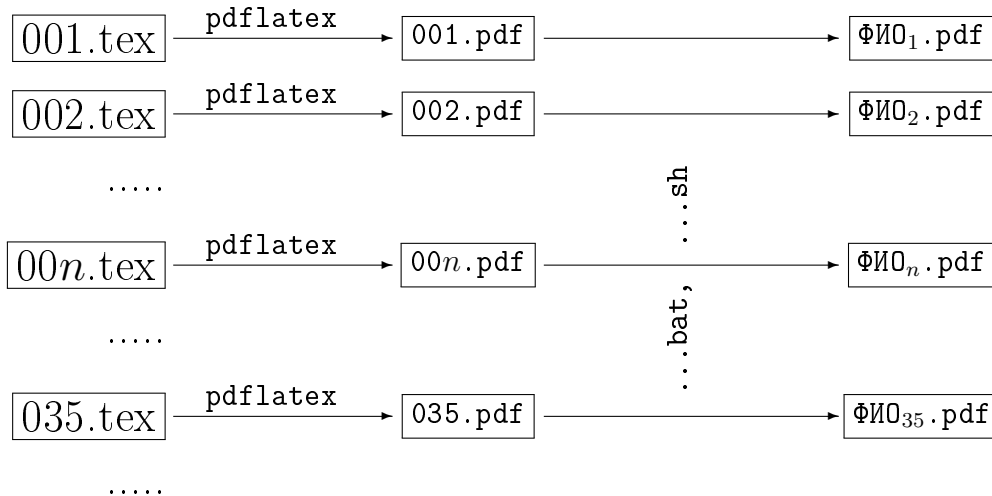
В этом же пакетном файле запускается копирование файлов `00n.pdf` в файлы, именами которых являются ФИО студентов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



В этом же пакетном файле запускается копирование файлов $00n.pdf$ в файлы, именами которых являются ФИО студентов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



Полученные файлы `ФИО.pdf` объединяются в архив, который выкладываем на портал.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

001.tex

002.tex

.....

00*n*.tex

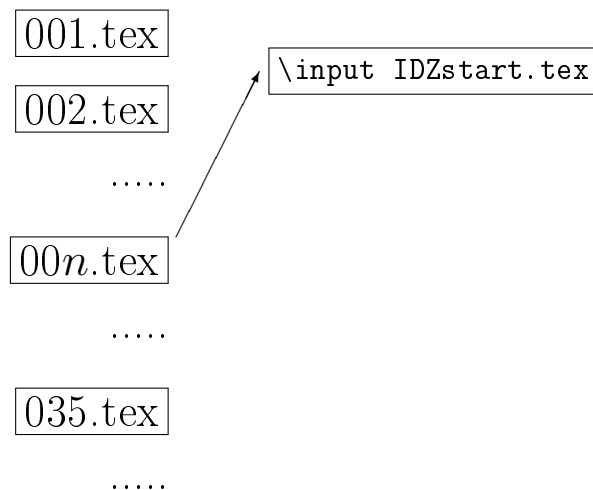
.....

035.tex

.....

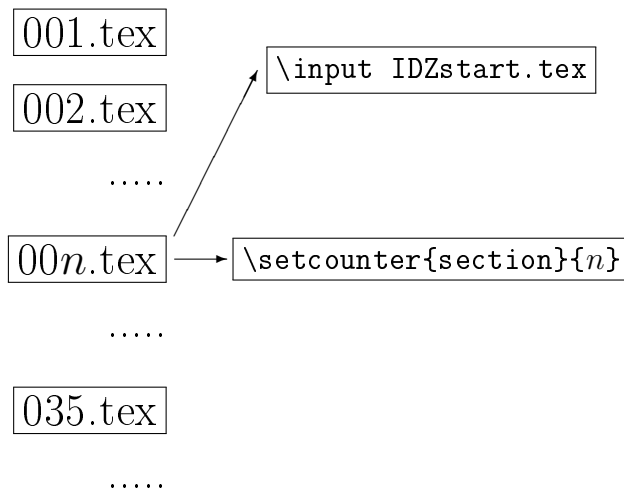
Рассмотрим устройство файла 00*n*.pdf.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



Рассмотрим устройство файла `00n.pdf`.

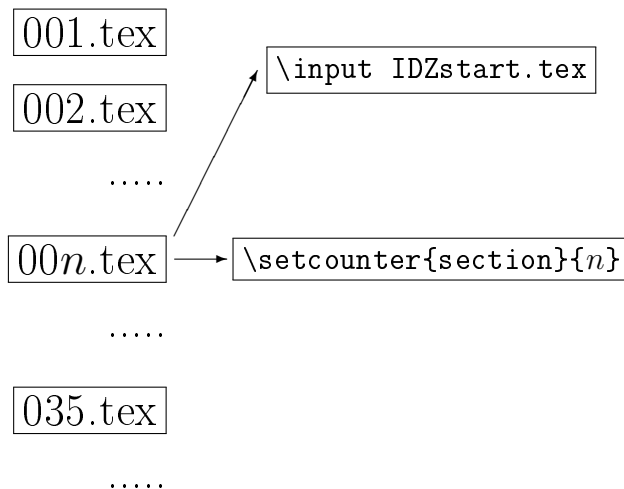
XIV.8.2. Структурная схема исходного кода для создания электронного теста



Рассмотрим устройство файла `00n.pdf`.

Здесь n совпадает с номером из имени файла, т.е.

XIV.8.2. Структурная схема исходного кода для создания электронного теста

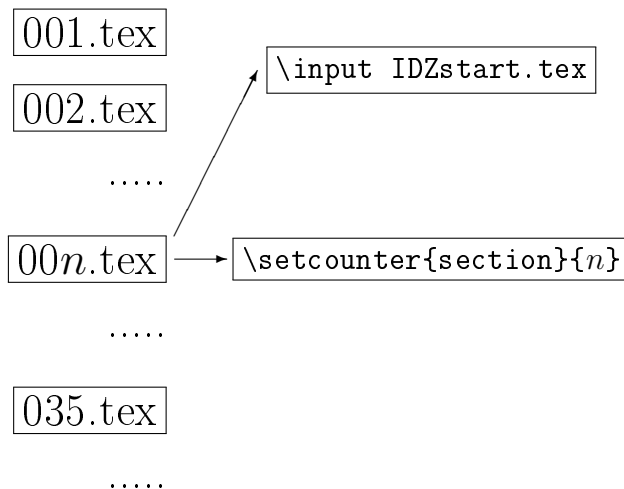


Рассмотрим устройство файла $00n.pdf$.

Здесь n совпадает с номером из имени файла, т.е.

в $001.pdf$ записано `\setcounter{section}{1}`,

XIV.8.2. Структурная схема исходного кода для создания электронного теста



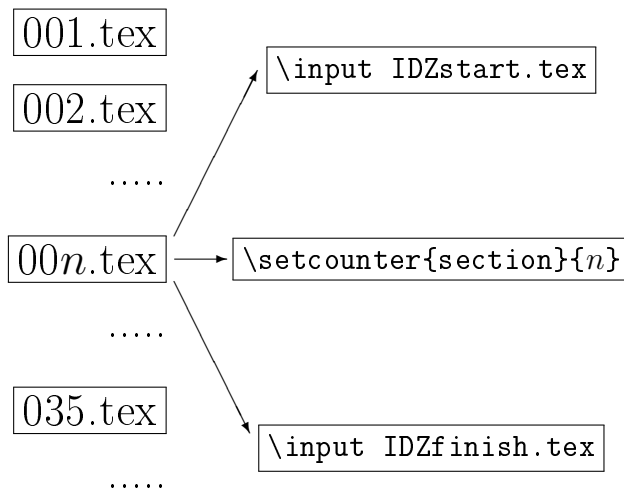
Рассмотрим устройство файла $00n.pdf$.

Здесь n совпадает с номером из имени файла, т.е.

в $001.pdf$ записано `\setcounter{section}{1}`,

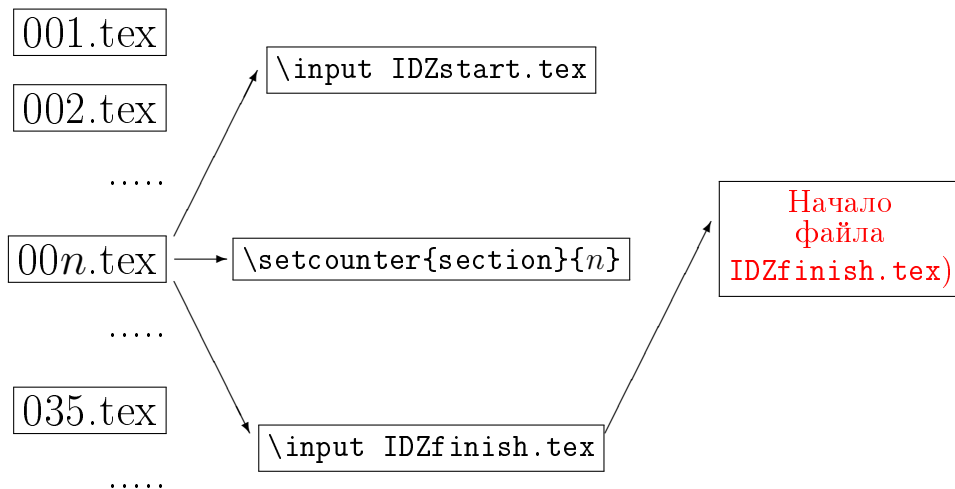
в $002.pdf$ записано `\setcounter{section}{2}`...

XIV.8.2. Структурная схема исходного кода для создания электронного теста



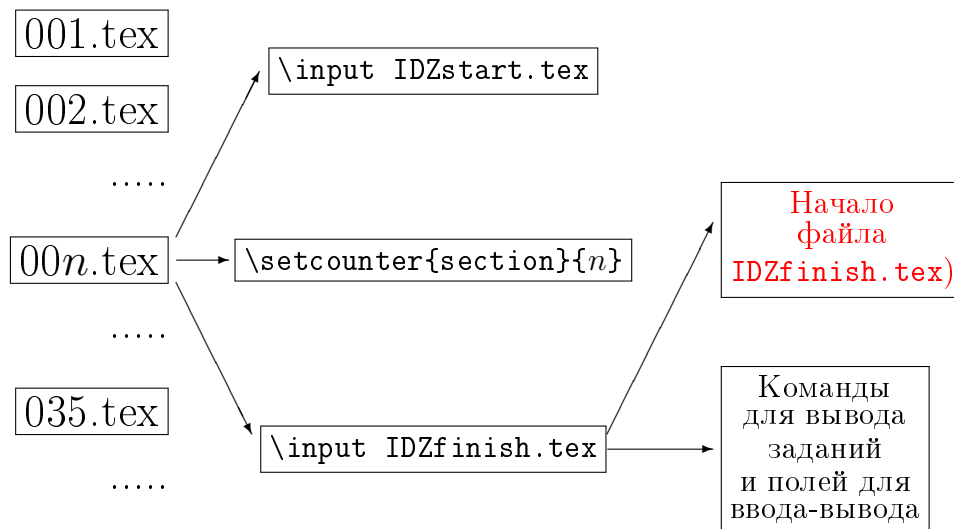
В файле `IDZfinish.tex` содержатся команды и ссылки непосредственно для формирования тестов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



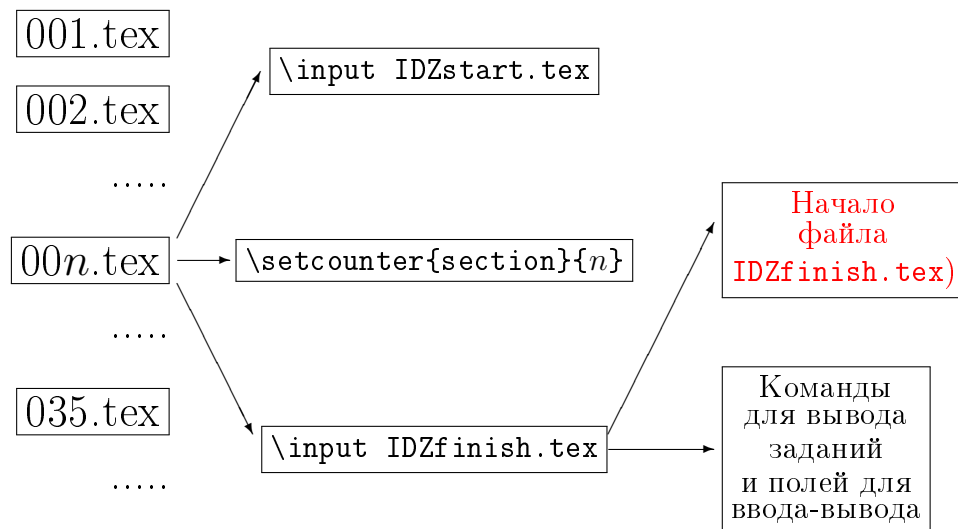
В файле `IDZfinish.tex` содержатся команды и ссылки непосредственно для формирования тестов.

XIV.8.2. Структурная схема исходного кода для создания электронного теста



В файле `IDZfinish.tex` содержатся команды и ссылки непосредственно для формирования тестов.

XIV.9. Структура файла IDZfinish.tex



В файле IDZfinish.tex содержатся команды и ссылки непосредственно для формирования тестов.

Этот файл содержит ряд ключевых команд, в которые автор текста может и должен вносить необходимые изменения.

XIV.9.1. IDZfinish: начало файла

Обычно первых строках находятся команды для загрузки баз заданий, содержащихся в файлах `STest . . . .`

XIV.9.1. IDZfinish: начало файла

`\input STest...%`

Обычно первых строках находятся команды для загрузки баз заданий, содержащихся в файлах `STest....`

XIV.9.1. IDZfinish: начало файла

`\input` STest...%

Разумеется, можно загружать несколько файлов с базами заданий.

XIV.9.1. IDZfinish: начало файла

`\input STestElemMathA.tex%`

`\input STestFormulasAa.tex%`

Разумеется, можно загружать несколько файлов с базами заданий.

XIV.9.1. IDZfinish: начало файла

<code>\input STestElemMathA.tex%</code>	Если команда <code>\PrintCorr</code>
<code>\input STestFormulasAa.tex%</code>	определена как 0, то на экран
<code>\def\PrintCorr{0}%</code>	не выводится кнопка сравне-
	ния.

Кнопку сравнения удобно использовать на этапе отладки генератора заданий при подготовке очередного генератора из файла `STest...`

XIV.9.1. IDZfinish: начало файла

<code>\input STestElemMathA.tex%</code>	Если команда <code>\PrintCorr</code>
<code>\input STestFormulasAa.tex%</code>	определена как 1, то кнопка
<code>\def\PrintCorr{1}%</code>	сравнения будет предоставле-
	на тестируемому.

Кнопку сравнения удобно использовать на этапе отладки генератора заданий при подготовке очередного генератора из файла `STest...`

XIV.9.1. IDZfinish: начало файла

```
\input STestElemMathA.tex%  
\input STestFormulasAa.tex%  
\def\PrintCorr{1}%  
  \def\PrintNameOfTest{}
```

В случае, когда команда `\PrintNameOfTest` определена, при компиляции будет выведена вспомогательная информация о названии команды, определенная в файле `STest...` и номере данного генератора, т.е. о значении параметра **#4** в команде `\STest...`, определенной в файле `STest...`

XIV.9.1. IDZfinish: начало файла

```
\input STestElemMathA.tex%  
\input STestFormulasAa.tex%  
\def\PrintCorr{1}%  
%\def\PrintNameOfTest{}%
```

В варианте, выдаваемом обучаемым, введение команды `\PrintNameOfTest` следует блокировать, например, с помощью символа `%`, превращающем часть до конца строки в комментарий, который игнорируется компилятором.

XIV.9.1. IDZfinish: начало файла

```
\input STestElemMathA.tex%      В команде \LearnYear опреде-
\input STestFormulasAa.tex%     ляется учебный год.
\def\PrintCorr{1}%
%\def\PrintNameOfTest{}%
\def\LearnYear{...}
```

XIV.9.1. IDZfinish: начало файла

```
\input STestElemMathA.tex%
```

```
\input STestFormulasAa.tex%
```

```
\def\PrintCorr{1}%
```

```
%\def\PrintNameOfTest{}%
```

```
\def\LearnYear{2015-2016}
```

Например, для 2015-6 учебного года команду `\LearnYear` надо определить указанным образом.

XIV.9.1. IDZfinish: начало файла

`\input STestElemMathA.tex%`

`\input STestFormulasAa.tex%`

`\def\PrintCorr{1}%`

`%\def\PrintNameOfTest{}%`

`\def\LearnYear{2015-2016}`

`\input GroupTeacher.inp`

Информация о преподавателе
и группе или классе содержит-
ся в файле `GroupTeacher.inp`

XIV.9.1. IDZfinish: начало файла

<code>\input</code>	<code>STestElemMathA.tex%</code>	Тема контрольной работы за-
<code>\input</code>	<code>STestFormulasAa.tex%</code>	даётся с помощью команды
<code>\def</code>	<code>\PrintCorr{1}%</code>	<code>\TestTheme</code>
<code>%\def</code>	<code>\PrintNameOfTest{}%</code>	
<code>\def</code>	<code>\LearnYear{2015-2016}</code>	
<code>\input</code>	<code>GroupTeacher.inp</code>	
<code>\newcommand</code>	<code>{\TestTheme}[3]{...}</code>	

XIV.9.1. IDZfinish: начало файла

<code>\input STestElemMathA.tex%</code>	Например, так определяется
<code>\input STestFormulasAa.tex%</code>	тема «Элементарная матема-
<code>\def\PrintCorr{1}%</code>	тика».
<code>%\def\PrintNameOfTest{}%</code>	
<code>\def\LearnYear{2015-2016}</code>	
<code>\input GroupTeacher.inp</code>	
<code>\newcommand{\TestTheme}[3]{Элементарная математика}</code>	

XIV.9.1. IDZfinish: начало файла

```
\input STestElemMathA.tex%  
\input STestFormulasAa.tex%  
\def\PrintCorr{1}%  
%\def\PrintNameOfTest{}%  
\def\LearnYear{2015-2016}  
\input GroupTeacher.inp  
\newcommand{\TestTheme}[3]{Элементарная математика}
```

После этих команд начинается блок, предназначенный для выбора конкретных заданий, описанный далее.

XIV.9.2. IDZfinish.tex: печать теста

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки  
%данного теста, например, TestLinSpaceAa  
}%
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки  
%данного теста, например, TestLinSpaceAa  
}{%Текст перед первым пунктом задания
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки  
%данного теста, например, TestLinSpaceAa  
}%Текст перед первым пунктом задания
```

Например, «Пусть то-то и то-то, и через... обозначено...»

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки  
                %данного теста, например, TestLinSpaceAa  
}{%Текст перед первым пунктом задания  
    Например, «Пусть то-то и то-то, и через... обозначено...»  
}{%Серия макрокоманд \STest... с текстами заданий
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки
               %данного теста, например, TestLinSpaceAa
}{%Текст перед первым пунктом задания
  Например, «Пусть то-то и то-то, и через... обозначено...»
}{%Серия макрокоманд \STest... с текстами заданий
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки
               %данного теста, например, TestLinSpaceAa
}{%Текст перед первым пунктом задания
  Например, «Пусть то-то и то-то, и через... обозначено...»
}{%Серия макрокоманд \STest... с текстами заданий
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{99}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки
               %данного теста, например, TestLinSpaceAa
}{%Текст перед первым пунктом задания
  Например, «Пусть то-то и то-то, и через... обозначено...»
}{%Серия макрокоманд \STest... с текстами заданий
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{99}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

В названии команды (желательно, чтобы она совпадала с именем файла) присутствует сокращение от темы теста.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки
               %данного теста, например, TestLinSpaceAa
}{%Текст перед первым пунктом задания
  Например, «Пусть то-то и то-то, и через... обозначено...»
}{%Серия макрокоманд \STest... с текстами заданий
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
                  {\arabic{eqquestionnoi}}{99}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

Выбор конкретного задания из файла `STest...` определяется значением четвёртого параметра команды `\STest...`

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки
                %данного теста, например, TestLinSpaceAa
}{%Текст перед первым пунктом задания
    Например, «Пусть то-то и то-то, и через... обозначено...»
}{%Серия макрокоманд \STest... с текстами заданий
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
    {\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%
    {\arabic{eqquestionnoi}}{99}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}
}{%Текст после вывода всех баллов

}{5}{6}{7}{8}{9}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.2. IDZfinish.tex: печать теста

```
\PrintETestA{TestLinSpaceAa%Фрамент уникального имени метки  
%данного теста, например, TestLinSpaceAa  
}{%Текст перед первым пунктом задания
```

Например, «Пусть то-то и то-то, и через... обозначено...»

```
{%Серия макрокоманд \STest... с текстами заданий  
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%  
{\arabic{eqquestionnoi}}{33}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}  
\STestComplexNumb{\arabic{section}}{\arabic{subsection}}%  
{\arabic{eqquestionnoi}}{99}{\GPA}{\GPB}{\GPC}{\GPD}{\GPE}  
}{%Текст после выведения всех баллов
```

Например, это может быть инструкция или пояснения.

```
{5}{6}{7}{8}{9}
```

Рассмотрим использование макрокоманды `\STest...`, определенной в **файле `STest...`**, см. также **раздел с примером**.

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input` *Файл_с_макрокомандой_для_группы.tex%*

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл с макрокомандой для группы формируется из списка группы в редакторе TeXCreator (версия не ниже 4.01) формируется в соответствии **с рекомендациями раздела XIV.10.2.**

Этот файл размещается в папке, в которой размещается база файлов TeX.

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл с макрокомандой для группы формируется из списка группы в редакторе TeXCreator (версия не ниже 4.01) формируется в соответствии **с рекомендациями раздела XIV.10.2.**

Этот файл размещается в папке, в которой размещается база файлов TeX.

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл с макрокомандой для группы формируется из списка группы в редакторе TeXCreator (версия не ниже 4.01) формируется в соответствии **с рекомендациями раздела XIV.10.2.**

Например, для MikTeX2.9 под Windows эти файлы должны находиться в папке

$$\left. \begin{array}{l} \text{ProgramFiles} \\ \text{ProgramFiles(x86)} \end{array} \right\} / \text{MikTeX 2.9/tex/TeXSample/IDZspisok/}$$

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл с макрокомандой для группы формируется из списка группы в редакторе TeXCreator (версия не ниже 4.01) формируется в соответствии **с рекомендациями раздела XIV.10.2.**

Для TeXLive под Linux эти файлы должны находиться в папке `usr/share/texmf/TeXSample/IDZspisok/`

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл с макрокомандой для группы формируется из списка группы в редакторе TeXCreator (версия не ниже 4.01) формируется в соответствии **с рекомендациями раздела XIV.10.2.**

Для данного случая при 64-битной Windows:

Program Files/MikTeX 2.9/tex/TeXSample/IDZspisok/Abc15s1.tex

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

```
\input Abc15s1.tex%
```

```
\input Файл_данных_преподавателя
```

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл данных преподавателя содержит информацию о нём.

(кодировка UTF-8)

```
\def\TeacherEMail{???@???.ru}%
```

```
\def\PhotoTeacher{Файл_с_фото.jpg}%
```

```
\def\TeacherFam{Фамилия_преподавателя}%
```

```
\def\TeacherName{Имя_преподавателя}%
```

```
\def\TeacherFather{Отчество_преподавателя}%
```

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

```
\input Abc15s1.tex%
```

```
\input Файл_данных_преподавателя
```

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл данных преподавателя содержит информацию о нём.

Например, для MikTeX2.9 под Windows эти файлы должны находиться в папке

$$\left. \begin{array}{l} \text{ProgramFiles} \\ \text{ProgramFiles(x86)} \end{array} \right\} / \text{MikTeX 2.9/tex/TeXSample/IDZteacher/}$$

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

```
\input Abc15s1.tex%
```

```
\input Файл_данных_преподавателя
```

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл данных преподавателя содержит информацию о нём.

Для TeXLive под Linux эти файлы должны находиться в папке `usr/share/texmf/TeXSample/IDZteacher/`

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

```
\input Abc15s1.tex%
```

```
\input Файл_данных_преподавателя
```

Для нормальной работы IDZfinish.tex в той папке, где генерируются задания по конкретной теме для конкретной группы или класса, должен быть файл GroupTeacher.tex.

Файл данных преподавателя содержит информацию о нём.

Для данного случая при 64-битной Windows:

Program Files/MikTeX 2.9/tex/TeXSample/IDZteacher/MelnikovYuB.tex

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

```
\input Abc15s1.tex%
```

```
\input Файл_данных_преподавателя
```

Для размещения файлов в этих папках необходимо обладать правами администратора (справедливо для Linux и Windows Vista и выше).

```
Program Files/MikTeX 2.9/tex/TeXSample/IDZspisok/Abc15s1.tex
```

```
Program Files/MikTeX 2.9/tex/TeXSample/IDZteacher/MelnikovYuB.tex
```


XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

`\input Файл_данных_преподавателя`

Для размещения файлов в этих папках необходимо обладать правами администратора (справедливо для Linux и Windows Vista и выше).

Program Files/MikTeX 2.9/tex/TeXSample/IDZspisok/Abc15s1.tex

Program Files/MikTeX 2.9/tex/TeXSample/IDZteacher/MelnikovYuB.tex

После размещения этих файлов следует проинформировать о них издательскую систему LaTeX

под Windows следует выполнить `mo_admin.exe`, т.е. «кликнуть» левой кнопке мыши Пуск, все программы, MikTeX 2.9, Settings (Admin);

XIV.9.3. IDZfinish.tex: файл GroupTeacher.tex

`\input Abc15s1.tex%`

`\input Файл_данных_преподавателя`

Для размещения файлов в этих папках необходимо обладать правами администратора (справедливо для Linux и Windows Vista и выше.

Program Files/MikTeX 2.9/tex/TeXSample/IDZspisok/Abc15s1.tex

Program Files/MikTeX 2.9/tex/TeXSample/IDZteacher/MelnikovYuB.tex

После размещения этих файлов следует проинформировать о них издательскую систему LaTeX

под Windows следует выполнить `mo_admin.exe`, т.е. «кликнуть» левой кнопке мыши Пуск, все программы, MikTeX 2.9, Settings (Admin); под Linux выполнить в терминале (вызывается Ctrl+Alt+T команду `sudo texhash`.

XIV.9.4. Пример фрагмента из \STest...

```
\newcommand{\STestPolynomAa}[9]{%
```

```
  \ifnum#4=1
```

```
    \GenerateNonZeroParamsZ{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}{#9}%
```

```
    ...
```

```
  \fi
```

```
  \ifnum#4=2
```

```
    \GenerateNonZeroParamsZ{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}{#9}%
```

```
    ...
```

```
  \fi
```

```
  \ifnum#4=3
```

```
    \GenerateNonZeroParamsZ{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}{#9}%
```

```
    ...
```

```
  \fi}
```

Рассмотрим пример выражения, обозначенного троеточием.

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

Рассмотрим пример теста по теме «выделение полного квадрата в квадратном трёхчлене».

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...
```

```
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[1t]{...  
  
}
```

```
\mbox{Заполните соответствующие поля:}\newline  
\mbox{$\backslash\begin{mathGrp}\PTs*{1}
```

```
\end{mathGrp}  
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\text{CountVvx}} = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...
```

```
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2}
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \alpha)^2 + \beta - \alpha^2.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{$\backslash\begin{mathGrp}\PTs*{1}`

`x^{2}` `{\the\CountVvx}{x}`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}`

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...
```

}

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \ComPlusN{\the\CountVvx}{x}
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...

}

\mbox{Заполните соответствующие поля:}\newline
\mbox{$\backslash\begin{mathGrp}\backslash\text{PTs}*{1}
x^{\{2\} \backslash\setcounter{countSign}{1}\backslash\text{ComPlusN}\backslash\text{the}\backslash\text{CountVvx}\{x\}

\end{mathGrp}
\backslash\text{CorrAnsButtonGrp}\backslash\text{the}\backslash\text{CountVax},\backslash\text{the}\backslash\text{CountVay}}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = (x + \alpha)^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVvx}}.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{$\begin{mathGrp}\PTs*{1}`

`x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = (x + \alpha)^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVvx}}.$$

```
\makebox(0,0)[1t]{...
```

}

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}  
      \the\CountVvy      =
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = (x + \alpha)^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVvx}}.$$

```
\makebox(0,0)[lt]{...
```

}

```
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{}=
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = (x + \alpha)^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVax}}.$$

```
\makebox(0,0)[1t]{...
```

}

```
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{}=
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}{$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \underbrace{\alpha}_{\backslash\text{CountVax}})^2 + \beta - \alpha^2.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{\$ \begin{mathGrp} \PTs*{1}`

`x^{2} \setcounter{countSign}{1} \ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+ \the\CountVax \right)^{2}`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax, \the\CountVay} \$}`

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \underbrace{\alpha}_{\backslash\text{CountVax}})^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVvy}}.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{$\begin{mathGrp}\PTs*{1}`

`x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+\coeff{\the\CountVax}\right)^{\{2\}`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{\$ \begin{mathGrp} \PTs*{1}`

`x^{\{2\} \setcounter{countSign}{1} \ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+\coeff{\the\CountVax}\right)^{\{2\}`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + \underbrace{2\alpha x + \beta}_{\backslash\text{CountVvy}} = (x + \alpha)^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

`\makebox(0,0)[lt]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{\$ \begin{mathGrp} \PTs*{1}`

`x^{2} \setcounter{countSign}{1} \ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+\coeff{\the\CountVax}\right)^{2}`

`+ \the\CountVay`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + \underbrace{2\alpha x + \beta}_{\backslash\text{CountVvx}} = (x + \underbrace{\alpha}_{\backslash\text{CountVax}})^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

`\makebox(0,0)[1t]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{\$ \begin{mathGrp} \PTs*{1}`

`x^{2} \setcounter{countSign}{1} \ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+\coeff{\the\CountVax}\right)^{2}`

`+\coeff{\the\CountVay}.`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

$\underbrace{\hspace{1.5cm}}_{\backslash\text{CountVvy}}$
 $\underbrace{\hspace{1.5cm}}_{\backslash\text{CountVvx}}$
 $\underbrace{\hspace{1.5cm}}_{\backslash\text{CountVax}}$
 $\underbrace{\hspace{1.5cm}}_{\backslash\text{CountVay}}$

`\makebox(0,0)[lt]{...`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{\$\begin{mathGrp}\PTs*{1}`

`x^{\{2\}} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{}=`

`\left(x+\coeff{\the\CountVax}\right)^{\{2\}}`

`+\coeff{\the\CountVay}.`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}\$}`

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

`\makebox(0,0)[lt]{...\CountVvx=\CountVax`

}

`\mbox{Заполните соответствующие поля:}\newline`

`\mbox{$\begin{mathGrp}\PTs*{1}`

`x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}`

`\ComPlusN{\the\CountVvy}{\}=`

`\left(x+\coeff{\the\CountVax}\right)^{\{2\}`

`+\coeff{\the\CountVay}.`

`\end{mathGrp}`

`\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}`

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...\CountVvx=\CountVax \multiply\CountVvx 2
}
```

```
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...\CountVvx=\CountVax \global\multiply\CountVvx 2
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{2}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$x^2 + 2\alpha x + \beta = (x + \alpha)^2 + \beta - \alpha^2.$$

```
\makebox(0,0)[lt]{...\CountVvx=\CountVax \global\multiply\CountVvx 2
```

}

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{2}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```


XIV.9.4. Пример фрагмента из \STest...

$$x^2 + \underbrace{2\alpha x}_{\backslash\text{CountVvx}} + \underbrace{\beta}_{\backslash\text{CountVvy}} = (x + \underbrace{\alpha}_{\backslash\text{CountVax}})^2 + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
```

}

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{\}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{2}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{2}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
}
```

```
\mbox{Заполните соответствующие поля:}\newline
```

```
\mbox{$\begin{mathGrp}\PTs*{1}
```

```
x^{2} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
```

```
\ComPlusN{\the\CountVvy}{}=
```

```
\left(x+\coeff{\the\CountVax}\right)^{2}
```

```
+\coeff{\the\CountVay}.
```

```
\end{mathGrp}
```

```
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
\advance\CountVay -\CountVvy
}\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
\advance\CountVay -\CountVvy \CountVay=-\CountVay}
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\underbrace{x^2 + 2\alpha x + \beta}_{\backslash\text{CountVvx}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
\advance\CountVay -\CountVvy \global\CountVay=-\CountVay}
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
\advance\CountVay -\CountVvy \global\CountVay=-\CountVay}
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```


XIV.9.4. Пример фрагмента из \STest...

$$\overbrace{x^2 + 2\alpha x + \beta}^{\backslash\text{CountVvy}} = \underbrace{(x + \alpha)^2}_{\backslash\text{CountVax}} + \underbrace{\beta - \alpha^2}_{\backslash\text{CountVay}}.$$

```
\makebox(0,0)[lt]{...\backslash\CountVvx=\CountVax \global\multiply\CountVvx 2
\CountVay=\CountVax \multiply\CountVay\CountVay
\advance\CountVay -\CountVvy \global\CountVay=-\CountVay}
\mbox{Заполните соответствующие поля:}\newline
\mbox{$\begin{mathGrp}\PTs*{1}
x^{\{2\} \setcounter{countSign}{1}\ComPlusN{\the\CountVvx}{x}
\ComPlusN{\the\CountVvy}{\}=
\left(x+\coeff{\the\CountVax}\right)^{\{2\}
+\coeff{\the\CountVay}.
\end{mathGrp}}
\CorrAnsButtonGrp{\the\CountVax,\the\CountVay}$}
```

XIV.9.4. Пример фрагмента из \STest...

Рассмотрим пример?

XIV.10. Одноразовые мероприятия для генерирования тестов

Следует сгенерировать файл с данными преподавателя и два типа файлов, в которых используется список группы.

Создание **файла с данными преподавателя** осуществляется только один раз.

XIV.10. Одноразовые мероприятия для генерирования тестов

Список группы используется при создании двух файлов:

- 1) `Имя_группы.tex` — макрокоманда `tex`, которая находится в `tex/texsample/IDZSpisok`, отвечающая за печать имени конкретного обучаемого в каждом электронном ИДЗ;

XIV.10. Одноразовые мероприятия для генерирования тестов

Список группы используется при создании двух файлов:

- 1) `Имя_группы.tex` — макрокоманда `tex`, которая находится в `tex/texsample/IDZSpisok`, отвечающая за печать имени конкретного обучаемого в каждом электронном ИДЗ;
- 2) `IDZgenerate.bat` (файл в кодировке 866) — пакетный файл для генерирования конкретного домашнего задания для учащихся конкретного класса или студентов конкретной группы.

XIV.10. Одноразовые мероприятия для генерирования тестов

Список группы используется при создании двух файлов:

- 1) **Имя_группы.tex** — макрокоманда `tex`, которая находится в `tex/texsample/IDZSpisok`, отвечающая за печать имени конкретного обучаемого в каждом электронном ИДЗ;
- 2) **IDZgenerate.bat** (файл в кодировке 866) — пакетный файл для генерирования конкретного домашнего задания для учащихся конкретного класса или студентов конкретной группы.

Файл **Имя_группы.tex** создается максимум один раз в начале учебного года.

XIV.10. Одноразовые мероприятия для генерирования тестов

Список группы используется при создании двух файлов:

- 1) **Имя_группы.tex** — макрокоманда `tex`, которая находится в `tex/texsample/IDZSpisok`, отвечающая за печать имени конкретного обучаемого в каждом электронном ИДЗ;
- 2) **IDZgenerate.bat** (файл в кодировке 866) — пакетный файл для генерирования конкретного домашнего задания для учащихся конкретного класса или студентов конкретной группы.

Файл **IDZgenerate.bat** создается в каждой папке `ФИО_Преподавателя/Название_группы/IDZ...`, в которой генерируется конкретное домашнее задание для учащихся конкретного класса или студентов конкретной группы.

XIV.10.1. Создание файла с данными преподавателя

1) Создать файл `Имя_преподавателя_транслитом.tex` (папка пока не важна) и открыть файл для редактирования (например, в `Far` эти оба действия можно сделать с помощью `Shift+F4`).

2) Выбрать кодировку `UTF-8` (в `Far` при подключенном плагине `fartrans11` выбор кодировки осуществляется сочетанием клавиш `Shift+F8`).

3) Записать в этот файл следующее:

```
\def\TeacherEMail{...email...@...}  
\def\PhotoTeacher{...Name_of_photo...jpg}%  
\def\TeacherFam{Фамилия_преподавателя}%  
\def\TeacherName{Имя_преподавателя}%  
\def\TeacherFather{Отчество_преподавателя}%.
```


XIV.10.1. Создание файла с данными преподавателя

4) Сохранить файл (в Far это F2) и закрыть его (в Far это Esc).

5) Переписать созданный файл вместе с фотографией (сохраненной в файле ...Name_of_photo...jpg) в папку Program Files/MikTeX2.9/tex/texsample/IDZteacher.6) Сообщить T_EX о новой информации в базе: под Windows XP,7,10 зайти в меню Пуск/Все программы/MikTeX2.9/, запустить Settings(Admin), в открывшемся окне во вкладке General нажать сначала на кнопку Refresh FNDB, по окончании работы программы нажать на кнопку Update Formats; под Windows 8 для запуска Settings(Admin) запустить программу mo_admin из Program Files/MikTeX2.9/miktex/bin/x64.

Под Linux файлы с фотографией и информацией о преподавателе надо разместить в usr/share/texmf/texsample/IDZteacher. При этом для того, чтобы сообщить T_EX о новой информации в базе достаточно в окне терминала запустить `sudo texhash`.

XIV.10.2. Подготовка файла *.tex со списком группы для генерации заданий

Если в окне программы TeXSample во вкладке PreTeX отсутствует вкладка MakeSpisokA.pretex, надо создать во вкладке PreTeX новую вкладку, скопировать в неё (в программе Far команды F4, Ctrl+C) содержимое файла MakeSpisokA.pretex, сохранить результат под именем MakeSpisokA.pretex.

Потом необходимо сделать следующее:

XIV.10.2. Подготовка файла *.tex со списком группы для генерации заданий

- 1) Открыть Far с правами администратора (навести указатель мыши на «иконку» программы Far, нажать правую кнопку мыши и выбрать соответствующий пункт выпадающего меню).
- 2) Скопировать файл IDZ-Spisok.tex с переименованием, соответственно, Имя_группы s1.tex, Имя_группы s2.tex и т.п.: в Far использовать F5.
- 3) В файле AllTests/Дополнительные/Для списка группы/SpisokGroups.tex выбрать строку типа \GroupParams{...}{...}{...}{...}{...}, которой еще не приписана группа, приписать название соответствующей группы, скопировать эту строчку в буфер. Открыть или создать (например, в Far комбинацией клавиш Shift+F4) файл GroupParams.txt, вставить Ctrl+V строку \GroupParams{...}{...}{...}{...}{...}. Проследите, чтобы в документе была только одна строка (поставьте курсор в конец строки и несколько раз нажмите Del).
- 4) Открыть с помощью TeXCreator Имя_группы sN.tex.
- 5) Скопировать в буфер обмена список группы или класса.
- 6) Вставить в этот файл список группы из буфера обмена, затерев весь текст, который был в файле (Ctrl+A, Ctrl+V). Сохранить файл, не закрывая. При необходимости, отредактировать (например, убрать номера, если они были).
- 7) Перейти во вкладку PreTeX (внизу экрана), открыть вкладку MakeSpisokA.ptetex, во второй строке указать число студентов в группе. Обработать Имя_группы sN.tex с помощью MakeSpisokA.ptetex. Сохранить полученный файл (Ctrl+S).
- 8) Скопировать полученный файл в tex/TeXSample/IDZSpisok/... с помощью F5.
- 9) Трижды крикнуть «Ура!» и трижды повернуться на правой ножке против часовой стрелки.

XIV.11. Структура папок с материалами о группе для создания генератора ИДЗ для них

Файлы и папки



Ну тут то все понятно

2015-6 (учебный год)

→ Иванов Пётр Сидорович

(Ф.И.О. преподавателя)

→ ЁКЛМН-15 (название группы)

→ ЁПРСТ-15 (название группы)

→ ...

→ ОЙЁЙ-15 (название группы)

→ YoKLMN15-Limits.zip

→ YoKLMN15-Diff.zip

→ ...

→ YoPRST15-....zip

→ Сидоров Иван Петрович

...

XIV.11. Структура папок с материалами о группе для создания генератора ИДЗ для них

Файлы и папки



Ну тут то все понятно

2015-6 (учебный год)

→ Иванов Пётр Сидорович

(Ф.И.О. преподавателя)

→ ЁКЛМН-15 (название группы)

→ IDZ-Limits (ИДЗ по пределам)

→ IDZ-Diff (ИДЗ по производной)

→ ...

→ GroupName.txt

→ GroupTeacher.inp

→ IDZgenerate.tex

→ ЁПРСТ-15 (название группы)

...

XIV.11. Структура папок с материалами о группе для создания генератора ИДЗ для них



Ну тут то все понятно

GroupTeacher.inp — файл с \input название файла со списком группы, и \input название файла с данными преподавателя;

GroupName.txt (имя группы транслитом)

IDZgenerate.tex

IDZName.txt (уже находится в папке).

XIV.12. Содержимое папки с материалами для генерации конкретного индивидуального домашнего задания

- 001.tex–099.tex — файлы для генерации задания для конкретного студента (учащегося).
- IDZstart.tex, IDZfinish.tex;
- GroupName.txt — файл с названием группы, лучше транслитом (латинскими буквами);
- GroupTeacher.inp — файл с \input название файла со списком группы, и \input название файла с данными преподавателя;
- IDZName.txt — файл с названием темы ИДЗ, лучше транслитом (латинскими буквами);
- файлы электронных учебников, на которые имеются гиперссылки в индивидуальных домашних заданиях.

Целесообразно файл **GroupName.tex** скопировать в папки, предназначенные для генерации индивидуальных заданий. Например, сделать этот файл в папке **IDZ-Set**, и потом скопировать его в папки **IDZ-Matrix**, **IDZ-Lim** и др.

XIV.13. Подготовка файлов для генерации заданий

Если в окне программы `TeXSample` во вкладке `PreTeX` отсутствует вкладка `MakeIDZgenerate.pretex`, надо создать во вкладке `PreTeX` новую вкладку, скопировать в неё (в программе `Far` команды `F4`, `Ctrl+C`) содержимое файла `MakeIDZgenerate.pretex`, сохранить результат под именем `MakeIDZgenerate.pretex`.

XIV.13.1. Подготовка файлов для генерации заданий: начальная подготовка для группы

- 1) Открыть Far.
- 2) В левом окне перейти в папке `/.../AllTests`, в правом окне в нужном месте создать папку с названием из ваших Фамилия Имя Отчество (например, Иванов Петр Сидорович).
- 3) В правом окне в вашей папке создать папки для каждой из ваших групп (в Far команда F7).
- 4) В папку каждой группы скопировать папки с тестами из `/.../AllTests`, которые будете выдавать студентам (в Far команда F5).
- 5) Скопировать файл `IDZgenerate.tex` в папку для генерации заданий (или создать с помощью Shift+F4, выбрать кодировку 866, в Far при подключенном плагине `fartrans11` выбор кодировки осуществляется сочетанием клавиш Shift+F8). Открыть эту копию в TeXCreator.
- 6) Скопировать в буфер обмена список группы или класса.

XIV.13.1. Подготовка файлов для генерации заданий: начальная подготовка для группы

7) Вставить содержимое буфера обмена в открытый файл `IDZgenerate.tex`, удалить кратные пробелы (в `TeXCreator` с помощью `Ctrl+F`), сохранить файл. Закрыть `IDZgenerate.tex` и его копию разместить во всех папках с материалами для генерирования заданий для данной группы.

8) Создать файл `GroupName.txt` (в `Far` это `Shift+F4`), в него записать имя группы (транслит), напр., `EMA15`.

Обратите внимание! В файле `GroupName.txt` должна быть только одна строка, нажимать `Enter` при редактировании нельзя. Можно поставить курсор в конце названия группы и несколько раз нажать `del`.

9) Скопировать `GroupName.txt` в каждую из папок с ИДЗ для этой группы.

10) Итак, в каждую из папок с ИДЗ для этой группы необходимо скопировать файлы:

`GroupName.txt`, `GroupTeacher.inp`, `IDZgenerate.tex`.

XIV.13.2. Подготовка файлов для генерации заданий: для каждого ИДЗ для группы

1) Необходимо выбрать, какие именно задания будут выданы обучаемым в данном ИДЗ. Для этого следует перейти в папку, с материалами для генерации заданий, открыть в TeXCreator файлы `000.tex` и `IDZfinish.tex`. Сделать файл `000.tex` главным (красный «минус» заменить на «зеленый плюс»). Проследите, чтобы в файле `IDZfinish.tex` символ `%` (символ комментария) не предшествовал команде `\def\PrintNameOfTest{}`.

2) Скомпилировать `000.tex` и просмотреть результат в Adobe Reader. Выбрать нужные тесты, выписать номера нужных генераторов.

3) В файле `IDZfinish.tex` забить `%` все строки, относящиеся к `\PrintETestA{...` для тестов, которые не нужны.

XIV.13.2. Подготовка файлов для генерации заданий: для каждого ИДЗ для группы

- 4) В папке с материалами для генерации заданий открыть файл `IDZgenerate.tex`, перейти во вкладку `PreTeX` (слева-снизу окна программы `TeXSample`), открыть вкладку `MakeIDZgenerate.pretex`, во второй строке указать число студентов в группе.
- 5) Обработать `IDZgenerate.tex` с помощью `MakeIDZgenerate.pretex`, измененный файл `IDZgenerate.tex` сохранить как `IDZgenerate.bat`.
- 6) Если случайно забыли при сохранении переименовать `IDZgenerate.tex` в `IDZgenerate.bat`, можно сделать это в `Far` с помощью `F6`.
- 7) Отредактировать файл `IDZfinish.tex`, во-первых, поставив `%` перед `\def\PrintNameOfTest{}` и, во-вторых, запретив печать ответа: `\def\PrintCorr{0}`.
- 8) Данную папку с заданиями можно использовать в качестве эталонной для формирования заданий для других групп потока, с естественными изменениями файлов `IDZgenerate.tex`, `GroupTeacher.inp`, `GroupName.txt`.
- 9) Трижды крикнуть «Ура!» и трижды повернуться на правой ножке против часовой стрелки.

XIV.13.3. Генерирование заданий

1) Запустите `IDZgenerate.bat` из `Far`.

2) Для ускорения компиляции вывод протокола компиляции в `IDZgenerate.bat` подавляется, поэтому при сбое компиляции компьютер «как бы зависает». В этом случае (при «как бы зависании» более чем на 3 минуты) закройте `Far`. Откройте его снова, зайдите в папку с генерируемыми заданиями и наберите в командной строке `pdflatex 001.tex`. Обнаружится, в чем причина сбоя в компиляции. Устраните причину сбоя сами или обратитесь за консультацией.

3) Сформированные файлы вместе с учебником запакуйте в архив. Для этого выделите все файлы `pdf`: «большой» (или «серый») «плюс», наберите в появившемся окне `*.pdf`. Для создания архива наберите `Shift+F1`, в качестве имени архива укажите имя группы и тему теста.

4) Удалите все сгенерированные файлы `pdf`, не удаляя файлы учебника.

5) Выложите архив студентам на портал.

Спасибо

За

Внимание!

e-mail: melnikov@k66.ru, melnikov@r66.ru

сайты: <http://melnikov.k66.ru>, <http://melnikov.web.ur.ru>

Перейдем: к **примерам по Т_EX**нологии PrimPattern,

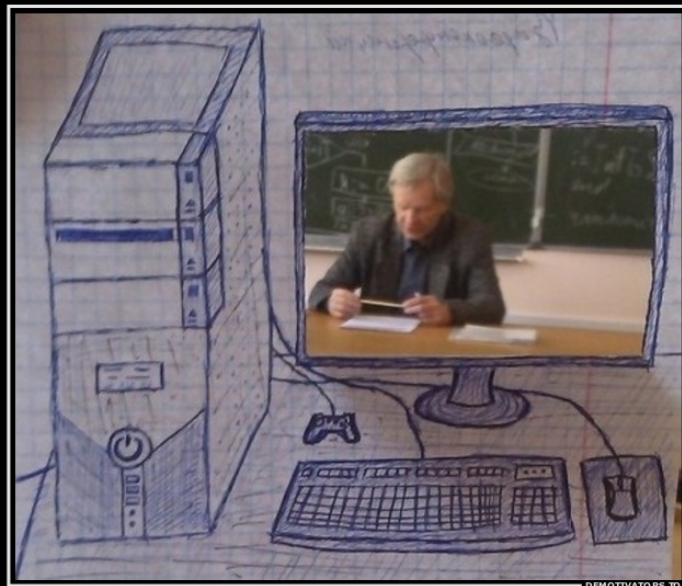
к **справочнику по Т_EX** 00Sprav

к **методическим вопросам?**





Мы на верном пути



Дистанционное обучение