

Верстка математических текстов в \LaTeX

Д.П. Кириенко, dk@179.ru

Курс занятий для математических классов средней школы №179 МИОО, 2010 г.

1 Основные принципы

Для работы с \LaTeX необходимо создать в любом редакторе обычный текстовый файл, содержащий набор специальных команд (предположим что он называется `file.tex`). Затем при помощи команды `latex file.tex` он компилируется в файл `file.dvi` в формате DVI (DeVice Independent). Этот файл можно просмотреть при помощи команды `xdvi file.dvi` (`yap file.dvi` если вы пользуетесь дистрибутивом MiKTeX под Windows), преобразовать в postscript файл при помощи утилиты `dvips` или в PDF-файл при помощи `dvipdf`:

Получение `file.dvi` из `file.tex`: `latex file.tex`

Получение `file.ps` из `file.dvi`: `dvips file.dvi -o file.ps`

Получение `file.pdf` из `file.dvi`: `dvipdf file.dvi`

Просмотр `file.dvi` в Linux: `xdvi file.dvi`

Просмотр `file.dvi` в Windows, дистрибутив MiKTeX: `yap file.dvi`

В настоящее время наиболее распространенным форматом для передачи документов с сохранением сложного форматирования и оформления, является формат PDF. Получить PDF-файл из документа \LaTeX можно одной командой: `pdflatex file.tex`.

Для удобства работы с \LaTeX мы будем использовать редактор TeXworks, который позволяет редактировать файлы, делать верстку и просматривать результат верстки. Этот редактор работает под Linux, Windows и другими системами.

Если в исходном \LaTeX -файле содержатся ошибки, то компилятор выдаст сообщение об ошибке и предложит выбрать дальнейшее действие. Обычная реакция — нажать клавишу Enter для продолжения работы компилятора.

2 Структура \LaTeX -файла

\LaTeX файл состоит из *преамбулы* документа, *основного текста*, *включенных и выключенных формул*. В тексте могут встречаться следующие специальные знаки:

% означает начало комментария; { и } означают начало и конец блока; \ означает специальную команду \LaTeX ; \$ используется для выделения формул; ~ означает неразрывный пробел; &, #, _, ^ прочие специальные символы. Пример простейшего документа \LaTeX с комментариями:

```
% Преамбула документа
% Следующая команда определяет класс документа: статья (article)
\documentclass{article}
% Подключаем пакет inputenc и устанавливаем кодировку текста (utf8)
\usepackage[utf8]{inputenc}
% Подключаем пакет babel и устанавливаем язык документа {russian}
\usepackage[russian]{babel}
% Эта команда означает начало основного текста
\begin{document}
Если  $a$  и  $b$  --- катеты прямоугольного треугольника, а  $c$  --- его гипотенуза,
то имеет место теорема Пифагора:

$$a^2+b^2=c^2$$


```

```
% Эта команда заканчивает документ, весь текст после нее игнорируется
\end{document}
```

Документ начинается с команд, устанавливающих класс документа и подключающих дополнительные пакеты локализации. Затем идет основной текст, содержащий в себе включенные и выключенные формулы. Включенная формула пишется внутри строки и содержимое формулы окружается знаками `$`: `$a$`. Отдельные буквы-переменные надо обязательно оформлять в виде формулы, чтобы они отображались правильным (наклонным) шрифтом. Выключенные формулы пишутся на отдельной строке и выделяются двумя знаками `$$`.

Набор обычного текста

Текст состоит из абзацев. Абзацы разделяются одной или несколькими пустыми строками. Внутри абзаца слова можно разделять одним или несколькими пробелами или разрывом строки. Строку в исходном тексте можно разрывать в произвольном месте, все равно `LaTeX` при верстке сделает разбивку текста по строкам самостоятельно. Также в тексте могут встречаться символы `~`, обозначающие неразрывный пробел (ставится перед тире и в других случаях). Для задания короткого тире нужно написать два знака минус подряд: `--`, для задания длинного тире — три знака минус.

В тексте и формулах часто встречаются специальные команды. Они начинаются с символа `\` и состоят из последовательности латинских букв произвольной длины или одного символа, не являющегося буквой. Примеры команд: `\LaTeX`, `\documentclass`, `\,`, `\"`. У некоторых команд может быть один или несколько аргументов, записываемых после команды. Обязательные аргументы заключаются в фигурные скобки. Необязательные аргументы заключаются в квадратные скобки и идут перед обязательными. Примеры: `\frac{5}{13}`, `\sqrt[3]{x}`.

3 Набор формул

В режиме верстки формул (как включенных, так и выключенных) игнорируются все пробелы. `LaTeX` расставит пробелы автоматически, при этом между различными математическими символами будут различные по ширине пробелы в соответствии с типографскими правилами. Далее приведены наиболее распространенные средства для изображения формул, но возможности `LaTeX` позволяют создать любую мыслимую формулу. Для дальнейшего изучения `LaTeX` рекомендуются книги Евгения Балдина “`LaTeX`, GNU/Linux и русский стиль” и Сергея Львовского “Набор и верстка в системе `LaTeX`”, которые можно скачать в интернете.

3.1 Переменные и индексы

Любые буквы в формуле трактуются как переменные. Запись `abc` трактуется как произведение переменных a , b и c . Степень переменной набирается при помощи символа `^`: `a^2` (a^2). Если степень содержит более одного символа, то ее надо заключить в группу из фигурных скобок: `x^{179}` (x^{179}). Нижние индексы набираются при помощи символа `_` по аналогичным правилам, у переменной может быть как верхний, так и нижний индекс: `$x_1^2+\dots+x_n^2$` ($x_1^2 + \dots + x_n^2$).

3.2 Дроби

Дроби можно набирать с помощью косой дробной черты: `$1/x$` ($1/x$), а можно с помощью команды `\frac`, у которой два параметра, записываемых после нее в фигурных скобках: числитель и знаменатель дроби. Например: `$\frac{1}{2}-\frac{1}{3}=\frac{1}{6}$` ($\frac{1}{2} - \frac{1}{3} = \frac{1}{6}$).

3.3 Операции и отношения

Математические знаки $+$, $-$, $=$, $<$, $>$, $:$ записываются соответствующими символами. Также полезно запомнить следующие команды, которые можно использовать только в формулах:

Операции `\pm` (\pm), `\mp` (\mp), `\cdot` (\cdot), `\times` (\times).

Отношения `\approx` (\approx), `\equiv` (\equiv), `\le` (\leq), `\ge` (\geq), `\neq` (\neq).

Множества `\in` (\in), `\notin` (\notin), `\subset` (\subset), `\subseteq` (\subseteq), `\cup` (\cup), `\cap` (\cap), `\setminus` (\setminus).

Кванторы `\forall` (\forall), `\exists` (\exists).

Любой символ можно перечеркнуть, если его предварить командой `\not`: $A \not\subset B$ ($A \not\subset B$).

3.4 Греческий алфавит

альфа	α	<code>\alpha</code>	бета	β	<code>\beta</code>	гамма	γ	<code>\gamma</code>	дельта	δ	<code>\delta</code>
эпсилон	ϵ	<code>\epsilon</code>	дзета	ζ	<code>\zeta</code>	эта	η	<code>\eta</code>	тэта	θ	<code>\theta</code>
иота	ι	<code>\iota</code>	каппа	κ	<code>\kappa</code>	лямбда	λ	<code>\lambda</code>	мю	μ	<code>\mu</code>
ню	ν	<code>\nu</code>	кси	ξ	<code>\xi</code>	омикрон	\omicron	<code>\omicron</code>	пи	π	<code>\pi</code>
ро	ρ	<code>\rho</code>	сигма	σ	<code>\sigma</code>	тау	τ	<code>\tau</code>	ипсилон	υ	<code>\upsilon</code>
фи	ϕ	<code>\phi</code>	хи	χ	<code>\chi</code>	пси	ψ	<code>\psi</code>	омега	ω	<code>\omega</code>

Кроме этих команд существуют также альтернативные варианты начертания некоторых букв: `\varepsilon` (ε), `\vartheta` (ϑ), `\varpi` (ϖ), `\varrho` (ϱ), `\varsigma` (ς), `\varphi` (φ). Из них в русских текстах принято использовать буквы ε и φ .

Для обозначения заглавных греческих букв есть следующие команды: `\Gamma` (Γ), `\Delta` (Δ), `\Theta` (Θ), `\Lambda` (Λ), `\Xi` (Ξ), `\Pi` (Π), `\Sigma` (Σ), `\Upsilon` (Υ), `\Phi` (Φ), `\Psi` (Ψ), `\Omega` (Ω). Остальные греческие буквы эквивалентны латинским буквам. Не следует использовать буквы Σ и Π для обозначения сумм и произведений, для этого есть специальные команды.

3.5 Тригонометрические и прочие функции

Формулу вроде $\sin x$ необходимо записывать с использованием специальной команды `\sin` x . Команда `\sin` в формуле пишет название функции прямым шрифтом и правильно расставляет интервалы. Если не использовать команду, а написать просто $\$ \sin x \$$ получится нечитаемая формула: $\sin x$. Аналогично определяются и прочие функции: `\cos`, `\tg`, `\ctg`, `\arcsin`, `\arccos`, `\arctg`, `\arcctg`, `\log`, `\ln`, `\exp`, `\min`, `\max`, `\sup`, `\inf`. Также есть команды для определения суммы `\sum` (\sum), произведения `\prod` (\prod) и интеграла `\int` (\int).

Эти функции можно снабжать верхними и нижними индексами. Несколько примеров:

$$\begin{array}{ll} \sin^2 x + \cos^2 x = 1 & \$\sin^2 x + \cos^2 x = 1\$ \\ \log_a a^b = b & \$\log_a a^b = b\$ \\ \lim_{n \rightarrow \infty} \frac{n+1}{n-1} = 1 & \$\lim_{n \to \infty} \frac{n+1}{n-1} = 1\$ \\ \sum_{i=0}^{\infty} q^i = \frac{1}{1-q} & \$\sum_{i=0}^{\infty} q^i = \frac{1}{1-q}\$ \\ \int_1^a \frac{1}{x} = \ln a & \$\int_1^a \frac{1}{x} = \ln a\$ \end{array}$$

Здесь мы использовали команду `\to` для рисования стрелочки и `\infty` для символа бесконечности.

3.6 Скобки

Круглые и квадратные скобки набираются при помощи соответствующих символов. Также при помощи символа `|` набираются вертикальные линии, например, обозначающие модуль ($\$ |x| \$$ даст $|x|$). Для набора фигурных скобок используются команды `\{` и `\}`.

Часто возникает необходимость увеличить размер скобок, как например в формуле

$$\left(1 + \frac{1}{n}\right)^n = \left(1 + \frac{1}{n}\right)^n$$

Сравните скобки в левой и правой частях равенства. Справа стоят обычные скобки, а левая часть получена при помощи кода `\left(1 + \frac{1}{n}\right)^n`, то есть надо поставить перед открывающей скобкой команду `\left`, а перед закрывающей скобкой команду `\right`, и \LaTeX автоматически подберет необходимые размеры скобок. Команды `\left` и `\right` — парные, они не могут встречаться поодиночке. Скобки, следующие после них могут быть разного вида. Если хочется опустить правую или левую скобку, то вместо нее необходимо поставить точку (`\right.`).

3.7 Корни и прочее

Квадратные корни задаются при помощи функции `\sqrt` у которой один обязательный параметр — подкоренное выражение, которое необходимо записать в фигурных скобках. Пример: `\sqrt{4}` ($\sqrt{4}$). Если требуется указать степень корня, то ее необходимо указать как необязательный параметр в квадратных скобках перед подкоренным выражением, например, `\sqrt[3]{x^3}=x` ($\sqrt[3]{x^3} = x$).

Штрихи, обозначающие, например, производную функции обозначаются апострофом: `f'` (f').

Факториал обозначается восклицательным знаком: `5!=120` ($5! = 120$).

3.8 Упражнения

Напишите следующий текст. В начале текста напишите свое имя, фамилию, дату выполнения работы:

Числа Фибоначчи задаются рекуррентным соотношением $F_n = F_{n-1} + F_{n-2}$.

Для чисел Фибоначчи верно тождество Кассини: $F_{n+1}F_{n-1} - F_n^2 = (-1)^n$ при $n > 0$.

Число $\varphi = \frac{1+\sqrt{5}}{2}$ носит название “золотого сечения”.

Известная формула приведения: $\sin \alpha = \cos(\pi/2 - \alpha)$.

Если α, β, γ — углы треугольника, то $\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma + 2 \cos \alpha \cos \beta \cos \gamma = 1$.

Известное соотношение в треугольнике: $m_a^2 = (2b^2 + 2c^2 - a^2)/4$.

Другое известное соотношение: $l_a = \frac{2bc \cos \frac{\alpha}{2}}{b+c}$.

Еще одно соотношение: $l_a = \sqrt{\frac{4p(p-a)bc}{(b+c)^2}}$.

Неравенства между средними: $\frac{n}{\frac{1}{a_1} + \dots + \frac{1}{a_n}} \leq \sqrt[n]{a_1 \dots a_n} \leq \frac{a_1 + \dots + a_n}{n} \leq \sqrt{\frac{a_1^2 + \dots + a_n^2}{n}}$

4 Смена шрифтов в тексте

Не увлекайтесь частой сменой шрифтов в тексте! Чем меньше различных видов шрифтов вы используете, тем легче его читать и тем красивее он выглядит!

4.1 Размер

Для изменения размера шрифта используются следующие команды:

<code>\tiny</code>	Меньше не бывает
<code>\scriptsize</code>	Очень маленький (как в верхних и нижних индексах)
<code>\footnotesize</code>	Маленький (как в сносках)
<code>\small</code>	Мелкий
<code>\normalsize</code>	Нормальный
<code>\large</code>	Немного увеличенный
<code>\Large</code>	Большой
<code>\LARGE</code>	Очень большой
<code>\huge</code>	Громадный
<code>\Huge</code>	Больше некуда

Эти команды переключают размер шрифта и действуют до конца группы. Группа — это фрагмент текста, заключенный в фигурные скобки или какое-нибудь окружение, поэтому если хочется какой-либо текст набрать увеличенным шрифтом, то это необходимо оформить в виде:

```
{\large Этот текст будет большего размера}.
```

4.2 Начертание

Каждый доступный шрифт имеет несколько атрибутов.

Первый атрибут — семейство (`family`) шрифта, или *гарнитура*. Наиболее распространенные семейства: `rmfamily` (шрифт с засечками), `sfamily` (шрифт без засечек), `ttfamily` (моноширинный шрифт в стиле пишущей машинки).

Второй атрибут — насыщенность (`series`), определяет толщину шрифта. Насыщенность бывает `mdseries` (обычная) или `bfseries` (**полужирная**).

Третий атрибут — начертание (`shape`). Бывает `upshape` (прямое начертание), `itshape` (*курсив*), `slshape` (*наклонный шрифт*) и `scshape` (КАПИТЕЛЬ).

Для изменения этих параметров есть команды двух видов. Команды первого вида работают аналогично командам для смены размера шрифта и их действие распространяется до конца группы. Команды второго вида используются с одним аргументом — текстом, который следует набрать этим шрифтом. Эти команды приведены в следующей таблице:

Без аргументов	С аргументом	На печати выйдет
<code>{\rmfamily Шрифт}</code>	<code>\textrm{Шрифт}</code>	Шрифт
<code>{\sffamily Шрифт}</code>	<code>\textsf{Шрифт}</code>	Шрифт
<code>{\ttfamily Шрифт}</code>	<code>\texttt{Шрифт}</code>	Шрифт
<code>{\mdseries Шрифт}</code>	<code>\textmd{Шрифт}</code>	Шрифт
<code>{\bfseries Шрифт}</code>	<code>\textbf{Шрифт}</code>	Шрифт
<code>{\upshape Шрифт}</code>	<code>\textup{Шрифт}</code>	Шрифт
<code>{\itshape Шрифт}</code>	<code>\textit{Шрифт}</code>	<i>Шрифт</i>
<code>{\slshape Шрифт}</code>	<code>\textsl{Шрифт}</code>	<i>Шрифт</i>
<code>{\scshape Шрифт}</code>	<code>\textsc{Шрифт}</code>	ШРИФТ

Различные атрибуты шрифта можно комбинировать. Например, применение `\large \sffamily \itshape` даст *большой курсивный шрифт без засечек*.

Если вы несколько раз сменили шрифт, то вернуть “одним махом” шрифт к стандартному можно при помощи команды `\normalfont`. Аналогичная ей команда с одним аргументом — `\textnormal`.

4.3 Задание

1. На странице размера А4 наберите несколько панграмм: шрифтами разного размера, разной гарнитуры, разной насыщенности и разного начертания.

Панграмма — фраза, содержащая все буквы алфавита. Панграммы используются для получения представления об используемом шрифте.

Примеры панграмм:

Экс-граф? Плюш изъят. Бъём чуждый цен хвоц! (ровно 33 буквы)

Эй, жлоб! Где туз? Прячь юных съёмщиц в шкаф. (ровно 33 буквы)

— Любя, съешь пиццы, — вздохнёт мэр, — кайф жгуч. (ровно 33 буквы)

В чащах юга жил был цитрус... — да, но фальшивый экземпляр! (нет буквы ё)

Южно-эфиопский грач увёл мышь за хобот на съезд ящериц.

Аэрофотосъёмка ландшафта уже выявила земли богачей и процветающих крестьян.

Съешь ещё этих мягких французских булок, да выпей же чаю.

Флегматичная эта верблюдица жуёт у подъезда засыхающий горький шиповник.

Широкая электрификация южных губерний даст мощный толчок подъёму сельского хозяйства.

2. Комбинируя 3 возможные гарнитуры, 2 насыщенности и 4 начертания можно получить 24 варианта шрифта. Однако, не все эти комбинации допустимы. Определите, какие комбинации “гарнитура – насыщенность – шрифт” являются допустимыми, а какие — нет. Результат оформите в печатном виде с примерами всех допустимых комбинаций.

5 Многострочные выключенные формулы

Включенные формулы автоматически переносятся на новую строку. Выключенные формулы всегда располагаются на одной строке, но часто возникает необходимость сделать формулу на несколько строк. Если для этого разбить формулу на две, то вертикальный интервал между формулами будет слишком велик. Лучше всего использовать дополнительные средства, содержащиеся в пакете `amsmath`. Для его подключения необходимо в преамбуле документа указать команду `\usepackage{amsmath}`.

Самый простой способ набора формулы в несколько строк — использование окружения `multline`. Рассмотрим пример:

$$\begin{array}{l} 1 + 2 + 3 + 4 + \dots \\ + 50 + 51 + 52 + \dots \\ + 99 + 100 = 5050 \end{array} \quad (1)$$

```
\begin{multline}
1+2+3+4+\ldots\
+50+51+52+\ldots\
+99+100=5050
\end{multline}
```

Фрагмент текста, начинающийся с команды `\begin{multline}` и заканчивающийся командой `\end{multline}` называется окружением `multline`. Это окружение **не должно быть** заключено в знаки `$$` — оно само по себе является формулой. Внутри этого окружения встречается команда `\ldots`, генерирующая многоточия и команда `\`, означающая разрыв строки. Обратите внимание на появившийся номер (1) рядом с формулой — чтобы его не было, надо использовать окружение `multline*`.

В предыдущем примере одна формула была разбита на несколько строк. Если подряд идет несколько выключенных формул, то для них можно использовать окружение `gather` (или `gather*` для отключения нумерации). Пример:

$$\begin{array}{l} \sin(x + y) = \sin x \cos y + \cos x \sin y \\ \cos(x + y) = \cos x \cos y - \sin x \sin y \end{array}$$

```
\begin{gather*}
\sin(x+y)=\sin x \cos y + \cos x \sin y\
\cos(x+y)=\cos x \cos y - \sin x \sin y
\end{gather*}
```

Наконец, рассмотрим верстку системы уравнений. В этом случае нам поможет окружение `aligned`, которое, в отличие от предыдущих окружений, необходимо использовать внутри формулы, т.е. внутри `$$`.

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

```
$$\left\{
\begin{aligned}
x+2y&=3,\
x-y&=1.
\end{aligned}
\right. $$
```

Формула начинается с команды `\left\{`, генерирующую большую фигурную скобку и заканчивающуюся парной ей командой `\right.`, не выводящую на экран ничего. Внутри этих команд находится окружение `aligned`, создающую многострочную формулу из двух уравнений. В этих формулах также встречается знак `&`, разделяющий столбцы в формуле: выражения в первом столбце выровнены вправо, а во втором столбце — влево.

В одной выключенной формуле может быть несколько окружений `aligned`, это позволяет записывать в одной строке несколько систем уравнений, например, разделяя их знаком равносильности `\Leftrightarrow` (`\Leftrightarrow`).

Автоматическая нумерация формул

В любом сложном математическом тексте часто возникает необходимость сослаться на ранее выписанную формулу. Для этого необходимая формула помечается собственным номером, который записывается в круглых скобках справа от формулы, после этого в тексте можно сослаться на данный номер. Как правило, формулы нумеруются последовательно: (1), (2), (3), ..., что создает проблемы при добавлении новых пронумерованных формул между уже вставленными: при этом сдвигается вся нумерация и необходимо исправлять все ссылки на все формулы, следующие после добавленной.

Но \TeX умеет автоматически нумеровать формулы и автоматически исправлять их номера всюду в тексте. Для этого любая выключенная формула может быть “помечена” специальной меткой, чтобы впоследствии можно было на нее сослаться по данной метке, а не по ее номеру. Номера же для помещенных формул \TeX расставит автоматически. Для того, чтобы пометить формулу она должна быть оформлена в виде окружения `equation`:

$$\sin^2 x + \cos^2 x = 1 \quad (2)$$

```
\begin{equation}
\sin^2 x + \cos^2 x = 1
\end{equation}
```

Окружение `equation` не нужно заключать внутри других формул, то есть внутри знаков `$$` — оно само по себе является формулой. Если между двумя окружениями `equation` вставить еще одно такое окружение, то номера, присвоенные формулам, изменятся.

Для удобства ссылки на формулу ее необходимо пометить при помощи команды `\label` с обязательным параметром — меткой, присвоенной данной формуле. Например, ранее мы могли бы пометить формулу при помощи метки `trig`:

```
\begin{equation}
\label{trig}
\sin^2 x + \cos^2 x = 1
\end{equation}
```

На печати помеченная формула ничем не отличается от формулы без метки, но в \TeX -документе можно сослаться на номер помеченной формулы при помощи команды `\ref` с параметром — меткой формулы. Например, можно писать так:

Аналогом основного тригонометрического тождества (2) для гиперболических функций является тождество $\operatorname{ch}^2 x - \operatorname{sh}^2 x = 1$.

Аналогом основного тригонометрического тождества (`\ref{trig}`) для гиперболических функций является тождество `\operatorname{ch}^2 x - \operatorname{sh}^2 x = 1`.

Для того, чтобы \TeX заменил ссылки на формулы на их номера, необходима двукратная перекомпиляция документа. При первой компиляции \TeX собирает информацию обо всех метках в документе и сохраняет ее в файле с расширением `.aux`, а при повторной компиляции данные о номерах формул считываются из `aux`-файла и вставляются вместо ссылок.

Также можно при помощи команды `\pageref` получить номер страницы, на которой встречается данная формула:

Основное тригонометрическое тождество (2) сформулировано на странице 8.

Основное тригонометрическое тождество (`\ref{trig}`) сформулировано на странице `\pageref{trig}`.

Абзацы

Абзацные отступы, вертикальные интервалы, разрывы строк и страниц

Чтобы подавить в одном абзаце отступ (красную строку) необходимо в его начале дать команду `\noindent`. Это полезно в случае, когда выключенная формула разрывает предложение, тогда следующий абзац не должен начинаться с красной строки. Если хочется изменить величину абзацного отступа во всем тексте, то необходимо дать в преамбуле документа команду вроде `\parindent=1cm`, устанавливающую величину отступа в 1 сантиметр. Можно использовать и другие единицы длины, например, миллиметры (mm). Иногда бывает полезно сделать абзацный отступ равный 0, а между абзацами вставить небольшой вертикальный промежуток. Для этого нужно вставить в преамбулу документа две команды:

```
\parindent=0mm    % Устанавливаем отступ в 0mm
\parskip=1mm      % Устанавливаем вертикальный интервал в 1mm
```

Для того, чтобы перед одним конкретным абзацем сделать вертикальный промежуток используется одна из команд `\smallskip`, `\medskip`, `\largeskip`, создающие маленький, стандартный и большой промежуток соответственно. Можно создать промежуток и заданного размера при помощи команды `\vspace`, например, `\vspace{1cm}` создаст вертикальный промежуток в 1 см.

Для того, чтобы начать новую строку, не разрывая абзаца, используется команда `\\`. Этой команде можно задать необязательный аргумент, задающий величину вертикального промежутка в месте разрыва: `\\[2mm]`.

Для разрыва страницы используется команда `\newpage`.

5.1 Различные окружения

Для создания абзацев с нестандартным выравниванием используются различные окружения:

`center` — абзацы с выравниванием по центру.
`flushleft` — абзацы с выравниванием влево.
`flushright` — абзацы с выравниванием вправо.
`quote` — цитаты.
`verbatim` — буквальное воспроизведение (без форматирования).
`itemize` — нумерованный список.
`enumerate` — нумерованный список.

Окружения начинаются с команды `\begin{имя_окружения}` и заканчиваются командой `\end{имя_окружения}`. Пример:

Этот абзац выровнен по центру.	<code>\begin{center}</code> Этот абзац\\выровнен по центру. <code>\end{center}</code>
Этот абзац выровнен по правому краю.	<code>\begin{flushright}</code> Этот абзац\\выровнен по правому краю. <code>\end{flushright}</code>

Окружение `quote` используется при цитат из других текстов. Окружение `verbatim` используется когда нужно вывести текст, не форматирова его средствами \LaTeX , при этом текст выводится моноширинным шрифтом (семейства `ttfamily`). Например, в этом листке окружение `verbatim` используется для примеров верстки в \LaTeX , также его следует использовать при воспроизведении исходных текстов программ.

В списках (окружения `itemize` и `enumerate` должны быть команды `\item` для задания нового элемента списка. Списки могут быть вложены друг в друга.

Нумерованный список из трех элементов:

1. One
2. Two
3. Three

Нумерованный список из трех элементов:

```
\begin{enumerate}
\item One
\item Two
\item Three
\end{enumerate}
```

6 Оформление документа “в целом”

6.1 Заголовок статьи

Любая математическая статья должна начинаться с указания названия статьи, автора и даты работы. Делается это при помощи команд `\title`, `\author` и `\date`, принимающими один аргумент, который, как всегда, пишется в фигурных скобках. После этих команд нужно дать команду `\maketitle`, которая сверстает заголовок статьи. Эти команды нужно давать сразу же после команды `\begin{document}`. Пример:

```
\title{0 простых четных числах}
\author{Д.П. Кириенко}
\date{Москва, 2006 г.}
\maketitle
```

В результате получится примерно то же самое, что и на первой странице этих листочков. В аргументах каждой из перечисленных команд может быть любой текст. Если не указать команду `\date`, то вместо нее будет автоматически напечатана текущая дата. Если хочется, чтобы вообще не отображалась дата, то необходимо дать команду с пустым аргументом: `\date{}`.

6.2 Аннотация

После названия статьи, как правило, идет ее аннотация (по-английски `abstract`) с кратким содержанием статьи. Для аннотации используется окружение `abstract`:

Аннотация

В этой работе рассматриваются уникальные математические объекты: четные простые числа.

```
\begin{abstract}
В этой работе рассматриваются уникальные
математические объекты: четные простые числа.
\end{abstract}
```

6.3 Структура документа

Вся статья разбивается на разделы (`section`), те в свою очередь разбиваются на подразделы (`subsection`), которые дробятся на еще более мелкие единицы (`subsubsection`). Для создания новых разделов и подразделов используются одноименные команды с единственным аргументом: заголовок раздела или подраздела. Пример:

```
\section{Существование простых четных чисел}
Здесь мы рассмотрим вопросы существования простых четных чисел.
```

```
\subsection{Пример простого четного числа}
Легко видеть, что 2 является простым четным числом.
```

```
\subsection{Невозможность существования других простых четных чисел}
Очевидно, что все четные числа, отличные от 2 имеют нетривиальный делитель, поэтому не могут быть простыми. Таким образом, не существует других простых четных чисел.
```

В результате получится следующий текст:

7 Существование простых четных чисел

Здесь мы рассмотрим вопросы существования простых четных чисел.

7.1 Пример простого четного числа

Легко видеть, что 2 является простым четным числом.

7.2 Невозможность существования других простых четных чисел

Очевидно, что все четные числа, отличные от 2 имеют нетривиальный делитель, поэтому не могут быть простыми. Таким образом, не существует других простых четных чисел.

Разделы нумеруются автоматически. Если хочется, чтобы некоторый текст был оформлен как раздел, но не пронумерован, необходимо для него использовать команду `\section*` (и аналогичные для подразделов).

Существуют еще более мелкие единицы деления текста: `\paragraph`, `\subparagraph`, `\subsubparagraph`. Они не нумеруются.

Создание оглавления

Оглавление создается одной командой `\tableofcontents`. На печати получится следующее:

Содержание

1	Основные принципы	1
2	Структура \LaTeX-файла	1
3	Набор формул	2
3.1	Переменные и индексы	2
3.2	Дроби	2
3.3	Операции и отношения	2
3.4	Греческий алфавит	3
3.5	Тригонометрические и прочие функции	3
3.6	Скобки	3
3.7	Корни и прочее	4
3.8	Упражнения	4
4	Смена шрифтов в тексте	5
4.1	Размер	5
4.2	Начертание	5
4.3	Задание	6
5	Многострочные выключенные формулы	7
5.1	Различные окружения	9
6	Оформление документа “в целом”	10
6.1	Заголовок статьи	10
6.2	Аннотация	10
6.3	Структура документа	10
7	Существование простых четных чисел	11
7.1	Пример простого четного числа	11
7.2	Невозможность существования других простых четных чисел	11
7.3	Стиль оформления страницы	12
7.4	Классовые опции	12
7.5	Сноски и примечания	12

В оглавление включаются все разделы, подразделы и подподразделы, имеющие номера (то есть без звездочек).

дочки в команде). Работает создание оглавления следующим образом: при компиляции \LaTeX -файла в специальный файл с именем заканчивающимся на `.aux` записывается информация о номерах, названиях разделов и номерах страниц, на которые они попадают. При повторной компиляции \TeX читает содержимое этого файла и вставляет информацию из него в оглавление. Таким образом, после первой компиляции оглавление еще отсутствует в документе и необходимо перекомпилировать документ во второй раз. Иногда при этом происходит изменение номеров страниц (поскольку в документ вставляется большой фрагмент текста) и тогда требуется еще одна компиляция.

7.3 Стиль оформления страницы

По умолчанию, внизу каждой страницы ставится ее номер. Это можно изменить, если задать свой стиль оформления страницы при помощи команды `\pagestyle`. У нее единственный аргумент, который может быть один из 4 слов:

`\pagestyle{empty}` — номера страниц отсутствуют;
`\pagestyle{plain}` — номер страницы внизу (по умолчанию используется этот стиль);
`\pagestyle{headings}` — присутствуют номера страниц и колонтитулы с названиями разделов;
`\pagestyle{myheadings}` — позволяет пользователю задавать свой стиль оформления колонтитулов.

Эти команды следует давать в преамбуле документа, поскольку действуют они на весь документ. Если хочется изменить стиль оформления одной страницы, то нужно использовать команду `\thispagestyle` также с единственным аргументом, аналогичным рассмотренным выше.

7.4 Классовые опции

Здесь мы рассмотрим самую первую команду любого документа `\documentclass`. У нее единственный обязательный параметр — класс создаваемого документа. Кроме рассмотренного значения `article` (статья) существуют еще `letter` (письмо), `report` (отчет или большая статья), `book` (книга) и `proc` (тезисы статьи). У каждого из стилей есть свои особенности, например, книга печатается на двух сторонах бумаги и делится не только на разделы, но еще и на части (`part`) и главы (`chapter`).

Перед обязательным параметром можно задать один или несколько необязательных параметров (в квадратных скобках через запятую). Пример:

```
\documentclass[12pt,a4paper,twocolumn]{article}
```

Наиболее часто используются следующие параметры:

`10pt` — основной текст имеет размером 10 пунктов (задан по умолчанию);
`11pt` — основной текст имеет размер 11 пунктов;
`12pt` — основной текст имеет размер 12 пунктов;
`a4paper` — используется бумага формата А4 (по умолчанию считается, что используется бумага формата `letter`, близкого к А4, но не совпадающего с ним);
`landscape` — текст направлен вдоль широкого края бумаги;
`twoside` — поля страниц различаются на четных и нечетных страницах (в классе `book` так принято по умолчанию);
`twocolumn` — текст верстается в две колонки;
`titlepage` — название, автор и дата документа печатаются на отдельной странице;
`fleqn` — выключенные формулы размещаются слева, а не по центру строки.

7.5 Сноски и примечания

Сноски (примечания к тексту, располагаемые внизу страницы¹) делаются при помощи команды `\footnote` с параметром — текстом сноски. Например, приведенный выше пример сделан при помощи команды

```
(... внизу страницы\footnote{вот как эта})
```

Вместо этой команды в тексте автоматически выставляется номер сноски².

Другой распространенный вид комментариев — заметки на полях. Они делаются при помощи команды `\marginpar`, опять таки, с единственным аргументом. Например, следующая команда

¹вот как эта

²Поэтому заботиться о проставлении номеров не надо

`\marginpar{!!!}`

приведет к результату, который можно увидеть справа.

!!!