

*Ярославский Владимир Валерьевич,
Ярославская Екатерина Юрьевна*

TeX ДЛЯ НАЧИНАЮЩИХ

Система TeX была разработана в 1978 году профессором Стенфордского университета, известным математиком и программистом Дональдом Кнутом (Donald E. Knuth). Он является автором популярного издания «Искусство программирования для ЭВМ», выпущенного в трех томах. Считается, что профессионал в области информатики должен быть знаком с этими книгами.

В процессе подготовки своих книг к печати Д.Кнуту надоело постоянно исправлять неточности при форматировании математических формул и текста. В результате он создал систему TeX для подготовки к печати математических текстов высокого полиграфического качества. С тех пор серьезные научные издательства и математические общества используют TeX. Надо заметить, что под словом TeX следует понимать как компьютерную систему (программу), так и сам язык управления печатью. В первую очередь TeX нужно знать тем, кто собирается стать математиком. Все крупные математические конференции принимают статьи только в TeX'e. Основное предназначение TeX'a – издание прекрасных книг и особенно книг, содержащих много

математики. Давайте разберемся, как изображается и читается название TeX. Изображается так, как показано на рисунке 1, буква E смещена вниз. Для удобства букву E просто пишут маленькой. Несмотря на то, что название состоит из латинских букв, они подразумевают собой греческие буквы τ (тау), ϵ (эпсилон), χ (хи), входящие в слово *tesnh* — «искусство». Произносится название как *tex* (так же, как, например, начало слова *техника*).

В чем преимущества и отличия этой системы от ныне популярного редактора Microsoft Word? TeX – это не редактор и потому не обладает свойством WYSIWYG (What You See Is What You Get – то, что видишь, то и получишь). Для TeX'a текст – это программа, которую нужно откомпилировать и на ее основе показать изображение на экране или напечатать на принтере. TeX реализован для всех операционных систем, в то время как Word присутствует только в Windows. Имеется возможность управлять форматированием текста, с легкостью набирать сложные математические формулы, самим формировать любые математические обозначения. Переносы слов, выравнивание по правой границе, автоматическую нумерацию формул – это все берет на себя TeX. Если вам нужно использовать в тексте множество экзотических шрифтов, создать рекламные буклеты или поздравительные открытки, добавить цветные картинки, красочные рамки — для этих целей используйте Word или StarOffice. Но если



Рисунок 1.

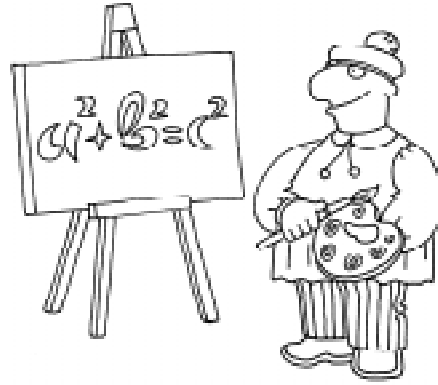
вам нужна красивая математика, выполненная по всем типографским правилам, TeX будет идеальным помощником. TeX'овские документы без проблем переносятся между платформами и одинаково точно воспроизводятся независимо от платформы. В частности, готовые TeX'овские документы легко конвертируются в распространенный формат PDF. Подготовив, например, какую-нибудь статью в TeX'e, вы можете создать ее PDF-версию и разослать ее своим друзьям и коллегам. Независимо от того, на какой операционной системе они работают (Windows, Unix/Linux или др.), этот PDF-файл они смогут прочитать свободно распространяемой программой Acrobat Reader.

Существуют макропакеты, такие как AmS-TeX и LaTeX, являющиеся наборами макрокоманд, состоящих из элементарных команд языка TeX. Команды этих пакетов реализуют сложные процедуры верстки, облегчая и упрощая пользователю работу по подготовке научно-технических документов. В дальнейшем будем рассматривать макропакет LaTeX, а именно LaTeX2e (хотя следует заметить, что многое будет относиться к самому TeX'у). Наиболее известную реализацию этих пакетов, MiKTeX, можно загрузить с сайта <http://www.miktex.org>.

Рассмотрим на рисунке 2 общую схему создания документов.

Сначала нужно набрать документ в любом простом текстовом редакторе (например, Notepad или встроенный редактор в Far'e). TeX'овские документы обычно имеют расширение .tex. После компиляции создается файл с расширением .dvi (DeVice Independent — независимый от устройства), которой можно просмотреть на экране или распечатать на принтере. Если нужно внести исправления, то редактируется исходный .tex-файл. Он заново компилируется, результаты опять просматриваются или распечатываются.

LaTeX'овский документ имеет следующую структуру:



...если вам нужна красивая математика...

```
\documentclass{<тип>}
\usepackage[russian]{babel}
<определения, настройки, опции>
\begin{document}
...
\end{document}
```

Тип документа может быть **book**, **article**, **report** и **letter**. Отличие между ними состоит в стиле нумерации, составлении оглавления, размере заголовков и др. Для большинства случаев подойдет тип **article**. Директива `\usepackage[russian]{babel}` подключает стилевой пакет русской локализации. Рассмотрим простой пример, который выводит текст «Здравствуй, мир!»:

```
\documentclass{article}
\usepackage[russian]{babel}
\begin{document}
Здравствуй, мир!
\end{document}
```

Вы, наверное, уже успели заметить, что команды начинаются с символа «\» (back slash). Текст документа начинается сразу после команды `\begin{document}` и продолжается до `\end{document}`. Как и в

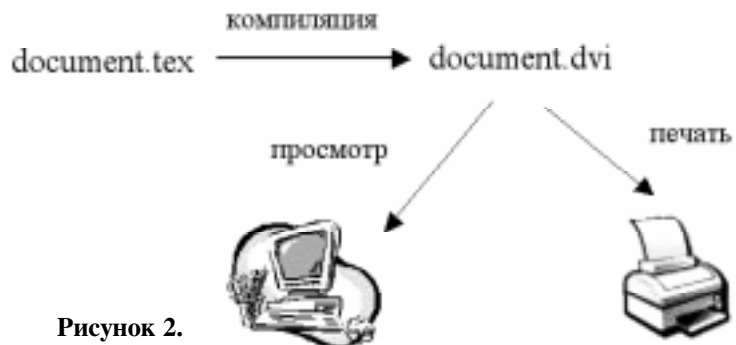


Рисунок 2.

языках программирования, можно определить свои команды, например:

```
\newcommand\hi[1]{Здравствуй, #1!}
```

Определение начинается с команды `\newcommand`, далее следует имя новой команды, в квадратных скобках число параметров (если они есть) и сама команда, в теле которой параметры обозначаются как `#` и номер параметра, команда `\\` означает переход на новую строку. Следующий пример выводит две строки: *Здравствуй, мир!* и *Здравствуй, вселенная!*

```
\documentclass{article}
\usepackage[russian]{babel}
\newcommand\hi[1]{Здравствуй, #1!\\}
\begin{document}
\hi{мир}
\hi{вселенная}
\end{document}
```

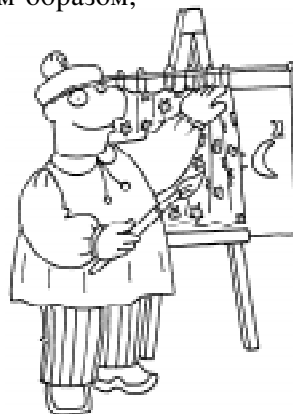
При подготовке больших статей может оказаться полезной команда `\input`, которая вставляет в текущий документ содержимое файла, переданного в качестве параметра, и TeX'овские комментарии `%`:

```
\documentclass{article}
\usepackage[russian]{babel}
\begin{document}
\input{chapter1.tex}
% \input{chapter2.tex}
% эта глава закомментирована
...
\input{chapterN.tex}
\end{document}
```

Комментарий действует от символа `%` и до конца строки. Таким образом, во время подготовки большой статьи или книги можно временно закомментировать ненужные главы, оставив только одну, и работать с ней. Если нужно закомментировать большой кусок текста, то более удобна будет следующая команда:

```
\newcommand\rem[1]{}
```

Очень удобная работа с текстом.



...можно временно закомментировать ненужные главы...

```
\rem{Этот весь большой кусок текста}
{закомментирован.}
```

Теперь познакомимся со шрифтами, какие они бывают и какого размера:

<code>\rm</code>	– прямой
<code>\tiny</code>	– крошечный
<code>\it</code>	– курсив
<code>\scriptsize</code>	– мелкий
<code>\bf</code>	– полужирный
<code>\footnotesize</code>	– подстрочный
<code>\tt</code>	– машинописный
<code>\small</code>	– маленький
<code>\sf</code>	– рубленый
<code>\normalsize</code>	– нормальный
<code>\sl</code>	– наклонный
<code>\large</code>	– большой
<code>\sc</code>	– капитель
<code>\Large</code>	– очень большой
<code>\LARGE</code>	– совсем большой
<code>\huge</code>	– огромный
<code>\Huge</code>	– громадный

Область действия этих команд задается фигурными скобками, например,

```
...вот такое {\LARGE большое} слово...
```

После компиляции вы получите:
...**вот такое БОЛЬШОЕ** слово...

В TeX'e существуют особые символы, которые нужно набирать специальным образом:

<code>\</code>	<code>\\backslash</code>	<code>#</code>	<code>\#</code>	<code>{ \{</code>
<code>~</code>	<code>\char126</code>	<code>\$</code>	<code>\\$</code>	<code>{ \}</code>
<code>^</code>	<code>\char94</code>	<code>%</code>	<code>\%</code>	
<code>_</code>	<code>_</code>	<code>&</code>	<code>\&</code>	

При работе с текстом также требуются команды:

<code>\newpage</code>	– начать новую страницу,
<code>\\</code>	– начать новую строку,
<code>\indent</code>	– вставить отступ, используемый в начале абзацев,
<code>\noindent</code>	– подавить (не вставлять) такой отступ,
<code>\<пробел></code>	– один пробел,
<code>\,</code>	– небольшой пробел (используется в формулах),

В TeX'e абзацы разделяются между собой пустой строкой. Выравнивание по правой границе система

берет на себя. Следует отметить, что последовательность из нескольких пробелов TeX воспринимает как один пробел.

Важно уметь различать дефисы и тире. Для набора короткого дефиса (как в слове *какой-нибудь*) используется один знак минуса. Для длинного дефиса, который появляется в диапазонах чисел, номерах телефонов, набирайте два знака минуса: **323--282--055** (*323-282-055*). Для тире (например, *TeX — великолепный язык*) нужно уже три знака минуса: **TeX --- великолепный язык**. Сам же знак минуса набирается в математическом режиме **\$- \$**.

При форматировании абзацев TeX переносит слова. Иногда требуется запретить перенос какого-нибудь слова. Для этой цели существует команда **\hbox**, которая сообщает TeX'у, что слово является неделимым. TeX не может знать всех переносов всех слов, и потому в некоторых случаях следует ему помочь, указав точки переноса: **пе\ -ре\ -строй\ -ка**.

Следует упомянуть о дополнительных пробелах, которые можно добавлять в текст: вертикальные **\vspace{<размер>}** и горизонтальные **\hspace{<размер>}**. Для вертикальных отступов есть стандартные команды: **\bigskip**, **\medskip** и **\smallskip**. Размер можно задавать в миллиметрах (**mm**), сантиметрах (**cm**), дюймах (**in**), пунктах (**pt**). Соотношения между ними такие: 1 cm = 10 mm, 1 in = 2.54 cm, 1 in = 72 pt.

Для формирования таблиц используется следующая схема:

```
\begin{tabular}{<описание>}
<ячейки>
\end{tabular}
```

где *<описание>* состоит из последовательности символов |, c, r, l. Одной букве (c, r, l) соответствует одна колонка. Буква c означает, что данные в этой колонке будут

№	Фамилия И.О.	телефон
1.	Иванов П.К.	353-2210
2.	Петров Н.И.	471-5128
3.	Ветров К.С.	728-7915

Рисунок 3.

выравнены по центру, буква r – выравнены по правой границе, буква l – по левой. Символ | проводит вертикальную черту между колонками. В теле таблицы команда **** переводит на новую строку, **\hline** проводит линию, символ **&** разграничивает данные между ячейками. Листинг 1 нарисует таблицу (см. рисунок 3).

Расскажем о самом важном в системе TeX – о наборе математики. Режим математических формул начинается (и заканчивается) символом **\$** или двумя символами **\$\$**. В первом случае математическая формула располагается в текущей строке. Во втором случае математическая формула размещается на отдельной строке. На рисунке 4 показано, как выглядит следующий пример:

Квадратным уравнением называется уравнение вида $ax^2+bx+c=0$, где $a \neq 0$. Например, $x^2+7x+1=0$

Квадратным уравнением называется уравнение вида $ax^2 + bx + c = 0$, где $a \neq 0$. Например,

$$x^2 + 7x + 1 = 0$$

Рисунок 4.



После компиляции вы получите: ... вот такое большое слово...

Листинг 1.

```
\begin{tabular}{|r||c|l|}\hline
\bf\No & \bf фамилия И.О. & \bf телефон \\ \hline
1. & Иванов П.К. & 353-2210 \\ \hline
2. & Петров Н.И. & 471-5128 \\ \hline
3. & Ветров К.С. & 728-7915 \\ \hline
\end{tabular}
```

$\alpha, \beta, \gamma, \delta, \Delta, \pi, \chi, \omega,$
 $\Omega, \xi, \Xi, \lambda, \zeta, \rho, \theta, \psi.$

Рисунок 5.

В системе TeX можно набирать различные специальные символы, например, греческие буквы. Наберем некоторые из них: альфа (команда TeX – `\alpha`), бета (`\beta`), гамма (`\gamma`), дельта (`\delta`), Дельта (`\Delta`), пи (`\pi`), хи (`\chi`), омега (`\omega`), Омега (`\Omega`), кси (`\kappa`), Кси (`\Ksi`), лямбда (`\lambda`), дзета (`\zeta`), ро (`\rho`), тета (`\theta`), пси (`\psi`). На рисунке 5 показан результат компиляции.



Режим математических формул начинается (и заканчивается) символом \$...

В формулах часто используются верхние и нижние индексы (см. рисунок 6):
 $B^k A_i$
 $x_3^2 = x_3^2$
 c_n^i
\$B^k\$, \$A_i\$

Рисунок 6.

Рассмотрим примеры с использованием нижних и верхних индексов вместе (порядок расстановки верхних и нижних индексов неважен): `$x^2_3 = x_3^2$` или чуть сложнее `$c^{i^2}_n$`.

Рисунок 7 иллюстрирует, как набирают некоторые математические значки. О том, как набирают другие символы и более сложные формулы, любознательный читатель сможет найти в указанной ниже литературе.

\dot{X}	<code>\dot{X}</code>	\pm	<code>\pm</code>
\ddot{X}	<code>\ddot{X}</code>	\oplus	<code>\oplus</code>
x''	<code>x''</code>	\int	<code>\int</code>
∞	<code>\infty</code>	\sum	<code>\sum</code>
\rightarrow	<code>\rightarrow</code>	\cap	<code>\cap</code>
\emptyset	<code>\emptyset</code>	\cup	<code>\cup</code>
∇	<code>\nabla</code>	\forall	<code>\forall</code>
\exists	<code>\exists</code>	\wedge	<code>\wedge</code>
\forall	<code>\forall</code>	$ $	<code>\setminus</code>
\neq	<code>\neq</code>	\div	<code>\div</code>
\in	<code>\in</code>	\Rightarrow	<code>\Rightarrow</code>
\ni	<code>\ni</code>	\times	<code>\times</code>

Рисунок 7.

Литература

1. Дональд Е.Кнут. Все про TeX. Протвино, 1993. 592 с.
2. Стивак М. Восхитительный TeX: руководство по комфортному изготовлению научных публикаций в пакете AmS-TeX. М.: Мир, 1993. 285 с.
3. Тельников К.О., Чеботаев П.З. LaTeX. Издательская система для всех. Новосибирск: Сибирский хронограф, 1994. 294 с.
4. Львовский С.М. Набор и верстка в пакете LaTeX. М.: Космосинформ, 1995. 374 с.
5. Гуссенс М., Миттельбах Ф., Самарин А. Путеводитель по пакету LaTeX и его расширению LaTeX2e. М.: Мир, 1999. 606 с.

Ярославский Владимир Валерьевич,
 кандидат физико-математических наук, программист,

Ярославская Екатерина Юрьевна,
 преподаватель кафедры высшей математики Санкт-Петербургского государственного морского технического университета (СПбГМТУ).

