

数学软件教程

2017年10月20日

LaTeX 科技文档排版

吕荐瑞

暨南大学数学系



1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

语言简介

LaTeX 是一种流行的科技文档排版语言。相对于 Word 这种办公排版软件，它有如下优点：

- ▶ 排版出的数学公式美观
- ▶ 自动编号能力强大
- ▶ 稳定性好，不容易崩溃

排版流程

与 C 语言类似，利用 LaTeX 排版文档需要一个编译的步骤。基本的流程如下：

1. 在 `somename.tex` 文件中编写文档
2. 用 `xelatex` 等程序编译上述文档
3. 得到 `somename.pdf` 文档用于阅读或打印

软件准备

为方便你使用 LaTeX, [这里](#)提供了相关软件下载.

1. 下载 miktex-2.9-2017.7z 压缩包
2. 将该文件解压到某个路径不含空格的目录
3. 双击 texworks.cmd 打开 TeXworks 编辑器

首次打开 TeXWorks 时可能较慢, 可以耐心多等待几秒.

打开模板

在打开 TeXworks 编辑器之后，我们先从模板文件建立第一个 LaTeX 文档：

1. 点击 File -> New from Template... 菜单项
2. 在窗口中点击打开 Chinese documents 目录
3. 选中 article-demo-cn.tex 再点击 Open 按钮
4. 点击 File -> Save 将该文件保存到某个目录

以后可在 File -> Open Recent 菜单打开 tex 文件.

开始编写

现在，我们开始试着编写我们的 LaTeX 文档：

1. 试着在编辑器中修改几个文字
2. 点击工具栏的绿色三角图标将会编译此文档
 - ▶ 如果无错误，右边将会显示生成的 PDF 文档
 - ▶ 如果有错误，下面的输出窗口将会显示错误信息
3. 编译时绿色图标将会变成红色，点击将停止编译
4. 根据错误信息修改文档，然后重新编译

注意：在绿色三角按钮的右边的下拉选择框中可以选择编译程序。对中文文档，务必将它选择到 XeLaTeX 这个程序。

1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

中文文档

最简单的中文文档如下（其中 % 后面的内容是注释）。

```
\documentclass{article}
```

```
\usepackage{ctex} %中文  
\usepackage{amsmath} %数学
```

```
\begin{document}
```

```
正文内容 .  $3^2+4^2=5^2$  .
```

```
\end{document}
```

导言区

正文区

文本段落

正文段落之间用空行隔开，比如：

第一个段落 .

第二个段落 .

第三个段落 .

而段落内部的换行用 `\newline` 命令.

数学公式

依据其出现位置，公式可分为行内公式和行间公式。

- ▶ 行内公式放在一对 $\$$ 中间
- ▶ 行间公式放在一对 $$$$ 中间

下面的例子同时出现这两种公式

如果 p 是素数， $\gcd(a, p) = 1$ ，则有
 $$$ a^{p-1} \equiv 1 \pmod{p} $$$

所得的结果如下

如果 p 是素数， $\gcd(a, p) = 1$ ，则有

$$a^{p-1} \equiv 1 \pmod{p}$$

上标下标

公式中的上标和下标分别用 \wedge 和 $_$ 表示。比如

$$\$(x_1+x_2)\^2 = x_1\^2 + 2x_1x_2 + x_2\^2\$$$

所得的结果如下

$$(x_1 + x_2)^2 = x_1^2 + 2x_1x_2 + x_2^2$$

如果上下标中包含多个符号，需要将它们放在一对花括号中。比如前面例子中的 $a^{\{p-1\}}$ 。

命令参数

用 `\` 加上一个或多个英文字母组成一个 LaTeX 命令。比如之前例子中见到的 `\gcd`, `\equiv` 和 `\pmod`。

有的命令需要包含一个参数, 比如 `\pmod`。命令的参数要放在一对花括号中, 紧接在命令之后。比如之前我们写的 `\pmod{p}`, 它给出了 $(\text{mod } p)$ 。

有的命令包含多个参数, 比如分式命令 `\frac`。比如我们可以从 `\frac{a+b}{c}` 得到 $\frac{a+b}{c}$ 。

有的命令包含的第一个参数是可选的, 这个可选参数要放在一对方括号中。比如 `\sqrt{x}` 得到 \sqrt{x} ; 而 `\sqrt[3]{x}` 得到 $\sqrt[3]{x}$ 。

1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

标题目录

要生成文章的标题栏，可以用下面几个命令：

```
\title{文章标题}  
\author{作者姓名}  
\date{写作日期}  
\maketitle
```

其中 `\date` 命令的参数为空时将不显示日期；而 `\date` 命令省略时将使用当前日期。

文章的目录可以用 `\tableofcontents` 命令生成。

定理环境

在 LaTeX 中很容易编写自动编号的定理。首先在导言区中写上

```
\newtheorem{theorem}{定理}  
\newtheorem{corollary}{推论}
```

然后在正文区中写上左侧的代码就得到右侧的结果：

```
\begin{theorem}  
定理内容 .  
\end{theorem}          定理 1  定理内容 .  
  
\begin{corollary}  
推论内容 .  
\end{corollary}       推论 1  推论内容 .
```

用 `\begin{名称}` 和 `\end{名称}` 包含的内容称为一个**环境**。

制作表格

利用 `tabular` 环境,我们可以制作下面的表格.其中 `|l|ccc|` 中的 `l` 表示第一列左对齐,后面的三个 `c` 表示后面三列居中对齐.

```
\begin{tabular}{|l|ccc|}
\hline
数学家 & 费马 & 欧拉 & 费马 \\
\hline
出生年份 & 1601 & 1707 & 1777 \\
\hline
\end{tabular}
```

数学家	费马	欧拉	费马
出生年份	1601	1707	1777

1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

数学阵列

在数学公式中有一个与 `tabular` 用法类似的 `array` 环境。比如我们可以制作下面的数学阵列。

```


$$\begin{array}{|l|rrrrr|}
\hline
n & 1 & 2 & 3 & 4 & 5 \\
\hline
n^2 & 1 & 4 & 9 & 16 & 25 \\
\hline
\end{array}$$


```

n	1	2	3	4	5
n^2	1	4	9	16	25

数学阵列

利用 `array` 环境, 很容易写出矩阵和行列式等多行多列的公式.

```
$$\left(\begin{array}{ccc}
```

```
1 & 2 & 3 \\
```

```
4 & 5 & 6 \\
```

```
7 & 8 & 9
```

```
\end{array}\right)$$
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

其中的 `\left` 和 `\right` 用于表示一对**定界符**; 定界符 (比如这里的圆括号) 会依据内容的高度自动调整大小.

数学阵列

注意，如果定界符其中一边空缺，则必须用 `.` 表示，如下例：

```
$|x|=\left\{\begin{array}{l}x, & \text{if } x \ge 0; \\-x, & \text{if } x < 0.\end{array}\right.$
```

$$|x| = \begin{cases} x, & \text{if } x \geq 0; \\ -x, & \text{if } x < 0. \end{cases}$$

多行公式

利用 `align*` 环境，很容易写出多行的对齐公式。比如

```
\begin{align*}
(x+y)^2 &= (x+y)(x+y) \\
&= x^2 + xy + yx + y^2 \\
&= x^2 + 2xy + y^2
\end{align*}
```

$$\begin{aligned}
(x+y)^2 &= (x+y)(x+y) \\
&= x^2 + xy + yx + y^2 \\
&= x^2 + 2xy + y^2
\end{aligned}$$

其中 `&` 指明各行对齐的位置。将 `align*` 改为 `align` 将得到带编号的公式。

1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

错误显示

在 TeXworks 中编译文档时，下面的输出窗口将会显示相关信息。输出窗口有这两个页面：

- ▶ “Console output” 页面显示详细的编译信息
- ▶ “Errors, warnings, badboxes” 页面只显示错误和警告
 - ▶ 编译错误用红色标示：这些问题必须改正后重新编译
 - ▶ 编译警告用黄色标示：这些问题初学者暂时可以忽略

如果文档没有错误，编译成功后输出窗口将会自动隐藏。

命令未定义

最常见的错误是命令名称输入错误。比如将 `\author{作者}` 错写为 `\authoros{作者}`，编译后将会看到如下错误信息：

```
! Undefined control sequence.  
l.12 \authoros
```

其中第一行说明错误原因是“命令未定义”，第二行说明错误出现在第 12 行的 `\authoros` 这里。

缺少 \$ 号

第二常见的错误是忘记将公式放在一对 \$ 里面. 比如将 $2^3=8$ 错写为 $2^3=8$, 编译后将会看到如下错误信息:

```
! Missing $ inserted.
```

```
<inserted text>
```

```
      $
```

```
1.19 2^
```

```
      3=8
```

其中第一行说明错误原因是“缺少 \$ 号”, 后面几行说明错误出现在第 19 行的 $2^3=8$ 这里.

花括号不配对

第三常见的错误是花括号无法配对。比如将 $\sqrt{2}$ 错写为 $\sqrt{2}$ ，编译后将会看到如下错误信息：

```
! Missing } inserted.  
<inserted text>  
      }  
l.17  $\sqrt{2}$ 
```

其中第一行说明错误原因是“缺少 } 号”，后面几行说明错误出现在第 17 行这里。

1 开始使用

2 快速入门

3 撰写文本

4 撰写公式

5 错误处理

6 命令速查

特殊字符

在 LaTeX 中，有几个特殊字符不能直接输入得到，而需要用下面对应的命令：

- ▶ `\{` 给出 {
- ▶ `\}` 给出 }
- ▶ `\$` 给出 \$
- ▶ `_` 给出 _
- ▶ `\#` 给出 #
- ▶ `\&` 给出 &
- ▶ `\%` 给出 %
- ▶ `\^{}` 给出 ^

最后，要得到反斜杠符号 `\`，在文本中可以用 `\textbackslash`，在公式中可以用 `\backslash` 命令。

希腊字母

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

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

函数命令

<code>\sin</code>	sin	<code>\arcsin</code>	arcsin	<code>\det</code>	det	<code>\sup</code>	sup
<code>\cos</code>	cos	<code>\arccos</code>	arccos	<code>\dim</code>	dim	<code>\inf</code>	inf
<code>\tan</code>	tan	<code>\log</code>	log	<code>\hom</code>	hom	<code>\sinh</code>	sinh
<code>\cot</code>	cot	<code>\ln</code>	ln	<code>\gcd</code>	gcd	<code>\cosh</code>	cosh
<code>\sec</code>	sec	<code>\lg</code>	lg	<code>\max</code>	max	<code>\tanh</code>	tanh
<code>\csc</code>	csc	<code>\exp</code>	exp	<code>\min</code>	min	<code>\coth</code>	coth

数学符号

- ▶ `\ge` 给出 \geq
- ▶ `\le` 给出 \leq
- ▶ `\neq` 给出 \neq
- ▶ `\cdot` 给出 \cdot
- ▶ `\cdots` 给出 \dots
- ▶ `\times` 给出 \times
- ▶ `\div` 给出 \div
- ▶ `\infty` 给出 ∞
- ▶ `\pm` 给出 \pm
- ▶ `\partial` 给出 ∂
- ▶ `\mid` 给出 $|$
- ▶ `\int` 给出 \int
- ▶ `\iint` 给出 \iint
- ▶ `\sim` 给出 \sim
- ▶ `\approx` 给出 \approx
- ▶ `\cong` 给出 \cong
- ▶ `\equiv` 给出 \equiv
- ▶ `\leftarrow` 给出 \leftarrow
- ▶ `\rightarrow` 给出 \rightarrow
- ▶ `\Leftarrow` 给出 \Leftarrow
- ▶ `\Rightarrow` 给出 \Rightarrow

大型算符

公式中常见的大型算符有这几个：

- ▶ `\lim` 给出 \lim
- ▶ `\sum` 给出 \sum
- ▶ `\prod` 给出 \prod

这些大型算符在行内公式出现时，上下标将显示在右边。

如果要想上下标显示在上下方，可以在命令后加上 `\limits` 命令。即上述命令要分别写成 `\lim\limits`、`\sum\limits` 和 `\prod\limits`。比如

```
$\lim\limits_{x\to 0}\sin x=1$
```

给出 $\lim_{x \rightarrow 0} \sin x = 1$ 。

参考资料

纸质书籍 刘海洋,《LaTeX 入门》,第 1 版,2013 年.

电子书籍 黄新刚,《LaTeX 笔记》,第 2 版,2013 年.