

LaTeX 文档

Rouliy

January 2022

目录

1	入门	4
1.1	关于 vs 编辑器的一些事	4
1.2	基本语法介绍	4
1.3	物理与逻辑结构	4
1.4	文字	5
1.4.1	字符输入	5
1.4.2	字体样式	6
1.4.3	换行, 换页和断字	6
1.4.4	长度	6
1.5	对齐和间距	6
1.5.1	段落和对齐	6
1.5.2	缩进和段间距	7
1.5.3	行间距	7
1.6	特殊段落	7
1.6.1	摘录	7
1.6.2	原文打印: 源代码和命令	7
1.6.3	脚注	8
1.6.4	边注	8
1.6.5	注释	8
1.7	列表	8
1.7.1	基本列表	8
1.7.2	定制列表	9
1.8	盒子	9
1.8.1	初级盒子	9
1.8.2	中级盒子	9
1.8.3	高级盒子	9
1.8.4	交叉引用	10
2	字体	10
2.1	字符集和编码的故事	10
2.2	字体格式	11
2.2.1	点阵和矢量字体	11

2.2.2	常见的字体格式	11
2.2.3	常见字体	11
2.3	字体的应用	11
2.3.1	早期技术	11
2.3.2	XeTeX	11
3	数学模式	11
3.1	基本元素	12
3.1.1	希腊字母	12
3.1.2	上下标和根号	12
3.1.3	分数	12
3.1.4	运算符	12
3.1.5	注音和标注	13
3.1.6	分隔符	13
3.1.7	省略号	14
3.1.8	空白间距	14
3.1.9	矩阵	14
3.2	多行公式	14
3.2.1	长公式	14
3.2.2	公式组	15
3.2.3	分支公式-分段函数的宝典	15
3.3	定理和证明	15
3.4	数学字体	16
4	图形	16
4.1	图形概述	16
4.1.1	图形格式	16
4.1.2	Driver	16
4.1.3	图形优化	16
4.1.4	图形转换和处理	17
4.2	插入图形	18
4.2.1	范围框	18
4.2.2	基本命令	18
4.2.3	图形操作	18
4.2.4	文件名和路径	19
4.2.5	figure 环境	19
4.2.6	插入多幅图形	19
4.3	矢量绘图	22
4.3.1	绘图工具概述	24
5	Metapost	24
5.1	准备工作	24
5.2	基本图形对象	25
5.2.1	点和线	25

5.2.2	预定义形状	25
5.2.3	曲线	25
5.3	图形控制	25
5.3.1	线型, 箭头, 彩色和填充	25
5.4	图形变换	26
5.5	标注	26
5.6	编程	26
5.6.1	数据类型和变量	26
6	PSTricks	26
6.1	Begin	27
6.2	基本图像对象	27
6.2.1	点和直线	27
6.2.2	圆, 椭圆, 圆弧, 扇形	27
6.2.3	曲线	28
6.2.4	网格和坐标轴	28
7	PGF	28
7.1	准备工作	28
7.2	基本图形对象	29
7.2.1	直线和矩形	29
7.2.2	圆, 椭圆, 弧形	29
7.2.3	曲线	29
7.2.4	网格	29
7.3	图形控制	29
7.3.1	箭头	29
7.3.2	线宽和线性	29
7.3.3	颜色和填充	29
7.3.4	渐变和阴影	29
7.3.5	样式	29
7.4	图形变换	29
7.5	示意图	29
7.5.1	节点	29
7.5.2	流程图	29
7.5.3	树	29
8	表格	29
8.1	简单表格	29
8.2	表格宽度控制	30
8.3	跨行跨列	30
8.4	数字表格	31
8.5	长表格	31
8.6	宽表格	32
8.7	彩色表格	32

9	文档的结构	34
9.1	标题	34
9.2	目录	34
9.3	参考文献	35
9.3.1	thebibliography	35
9.3.2	BibTeX	35
9.3.3	Natbib	36
9.4	索引	37
9.5	超链接	37
9.6	结构名	39
10	布局	39
10.1	历史	39
10.1.1	尺寸详解	39
10.2	页面样式	41
10.3	分栏	42
10.4	分页	43
11	符号	43
11.1	TeX 标志符号	43
12	简历的制作	44

1 入门

1.1 关于 vs 编辑器的一些事

注意在 VS 编辑器下编辑时，调出控制台 `ctrl+shift+Y`
 全屏模式：`ctrl+K` 接着按 `Z`；而后按两次 `esc` 可退出
`alt+z` 自动换行
`\LaTeX` 空格和 `\TeX` 空格生成

1.2 基本语法介绍

注意用 `xelatex` 程序编译源文件生成 PDF 文件。借助 LaTeX 工具
`\LaTeX` 源文件的语句分为三种：命令 (command)，数据和注释 (comment)。命令以 `\` 起始，多为一行；而环境包含一对起始声明和结尾声明，用于多行内容的场合。注释为百分号 `%`，在编译时忽略。

1.3 物理与逻辑结构

`\LaTeX` 文档的结构分为物理结构和逻辑结构。其中物理结构：源文件的组织形式，包括序言 (preamble) 和正文两部分。(现发行的包缺省时引擎一般为 `pdfTeX`)。逻辑结构：最终输出文档的结构，包含标题，目录，章节等。

10pt,11pt,12pt	正文字号, 第一个为缺省
letterpaper,a4paper	纸张尺寸, 如 [a4paper]
notitlepage,titlepage	标题后另起新页
onecolumn,twocolumn	栏数
draft	草稿模式, 某些行排得过满, draft 模式可以在他们右边标上粗黑线提醒用户

```

1 \documentclass[option]{ class } %文档类声明
2 \usepackage[option]{ package } %引入宏包

```

序言: 完成一些设置, 定义文件呢类型 documentclass, 引入宏包 usepackage, 定义命令, 环境等; 实际内容放在正文部分。L^AT_EX 中常见的文档类型 (documentclass): article、report、book, 其基本选项如下见表所示。注: 表格在线生成工具网址 <https://www.tablesgenerator.com/>。

逻辑结构: 文档的开头通常有标题, 作者, 摘要等信息。之后为章节等层次结构, 内容分散于层次结构间。具体用法如下:

```

1 \ title {LaTeX Notes}
2 \date{\today}
3 \maketitle %注意\maketitle命令放在最后
4 \begin{abstract}
5 \end{abstract}%摘要, article和report可以有, book里没

```

L^AT_EX 提供 7 种层次结构, 高级层次可以包含若干低级层次, article 中没有 chapter, 而 report 和 book 中支持所有层次

```

1 \part {..} %level -1
2 \chapter {..} %0
3 \section {..} %1
4 \subsection {..} %2
5 \subsubsection {} %3
6 \paragraph {} %4
7 \subparagraph {} %5
8
9 \setcounter{tocdepth}{2} %设置目录深度
10 \tableofcontents %列出目录
11
12 \chapter* {} %不想让某些层次的标题出现在目录中, 可以加上星号

```

1.4 文字

1.4.1 字符输入

普通字符, 控制符, 特殊符号, 预定义字符串, 注音符等。

特殊的控制符需在字符前加“\”表示为: #, \$, ^, _, \。L^AT_EX 中有短划线 (hyphen)、中划线 (en-dash) 和长划线 (em-dash)。短划线又称连字符, 用来连接单词; 中划线用来连接数字, 可以通过重复两次短划线得到; 长划线类似于中文的破折号, 重复三次短划线。为了便于比较, 这里也给出数学模式的减号。

```

1      computer-aided
2      1840--2010
3      to be---or not to be
4      $1-1=0$

```

实现的结果：computer-aided 1840-2010 to be—or not to be 1 – 1 = 0

1.4.2 字体样式

拉丁的文字体：衬线字体 (roman, serif), 无衬线字体 (sansserif), "sans" 来源于法语，表示“没有”。还有等宽字体 (monospace, typewriter), 衬线字体有修饰 [类似于中文的宋体，楷体，仿宋等]，无衬线字体则类似于黑体。强调的文字，画下划线的文字。

roman 字体, 衬线字体有修饰

sans serif, 无衬线字体, 类似黑体

SMALLCAPS, 小型大写字母, 大写字母的形状同, 但尺寸接近小写字母

粗体

半粗体

斜体, 修饰精细, 对原字体进行了重新设计, 版本问题

伪斜体, 比斜体要宽, 版本问题

等宽字体

1.4.3 换行, 换页和断字

通常用 \LaTeX 来自动换行, 用 `\\` 或 `\newline` 来强制换行, `\newpage` 来强制换页。

\LaTeX 会自动断字 (hyphenate), 使得每个字分布均匀, 有时也需要显示指明断字位置。

1.4.4 长度

精确排版时, 人们需要控制排版对象的尺寸和位置。其中 point 是传统印刷采用的单位, 而 big point 是 Adobe 推出的 PostScript 时的新单位, 注意 em 为相对单位, 比如当前字体为 11pt 时, lem 就是 11pt; 同时 ex 和 mu 也是相对单位。

对文章排版对象的尺寸和位置定义了一系列宏变量, 以便在排版重用

```

1      \setlength{变量名} %设置变量的值
2      \addtolength{变量名} %增加变量、
3      \newlength{变量名} %定义变量

```

1.5 对齐和间距

1.5.1 段落和对齐

\LaTeX 中段落为两端对齐 (fully justified), 段落分别居右, 中, 左 (`\raggedright`, `\centering`, `\raggedleft`) 功能相同。

```

1      \begin{ flushleft }
2      居左\\常用的对齐方式
3      \begin{center}

```

```

4          居中
5          \begin{ flushright }
6          居右
7          \end{ flushright }
8          \end{center}
9          \end{ flushleft }

```

1.5.2 缩进和段间距

正文的第一段落缺省不缩进首行，可以用宏包 `indentfirst`，其中段落首行缩进的距离可以用 `\parindent` 变量控制。

1.5.3 行间距

段落中相邻基线间的距离，缺省为单倍行距，`\linespread1.3` 为一倍半行距，1.6 为双倍行距。此命令改变正文的行距，目录，脚注，图表标题等。

局部文字的行间距用 `setpace` 宏包来实现

```

1      \usepackage{setspace}
2      \ singlespacing %单倍行距
3      \onehalfspacing %一倍半行距
4      \doublespacing %双倍行距
5      \setstretch {1.25} %设置任意行距
6      \begin{doublespacing}
7          双倍行距
8      \end{doublespacing}

```

1.6 特殊段落

1.6.1 摘录

三种摘录环境：`quote`, `quotation`, `verse`。`quote` 两端都缩进。`quotation` 在 `quote` 的基础上增加了首行缩进，`verse` 比 `quote` 多了第二行起的缩进。

`quote`, 引文两端缩进

在 `quote` 的基础上，`quotation` 增加了首行缩进

`verse`, 引文两端缩进

不过是第二行增加了缩进

1.6.2 原文打印: 源代码和命令

文档中的命令和源代码通常使用等宽字体，即原文打印。正文中插入少量的可以使用 `\verb` 命令；大段的文字可以使用 `verbatim` 环境比较方便。以及 `fancyvrb` 和 `listings` 宏包

```

verbatim不带星号的版本
printf("LaTeX");

```

```
verbatim不带星号的版本，能够显示空格
printf("\LaTeX*");
```

1.6.3 脚注

可以使用 `\footnote`¹，对脚注重新定义，脚注是个计数器（counter），有 5 中显示格式：

```
1 阿拉伯数字 \arabic{counter} 1,2,3
2 小写英文字母 \alph{counter} a,b,c
3 大写英文字母 \Alph{counter} A,B,C
4 小写罗马数字 \roman{counter} i,ii, iii
5 大写罗马数字 \Roman{counter} I,II,III
```

重定义 `\thecounter` 的方法改变显示的格式

```
1 \renewcommand{\thefootnote}{\roman{footnote}}
2 %注意此命令只改变之后的脚注序号
3 %以后出现的计数器，都可以用\thecounter的方法重新定义
```

1.6.4 边注

使用 `\marginpar` 命令。单面排版时，边注缺省排在页面右边空白处；此命令使用了浮动体来生成边注，不能在其他浮动体或脚注内嵌套。而 `marginnote` 宏包不涉及浮动。

边注的内容，
使用了
`marginnote`
宏包，没有用
浮动体

1.6.5 注释

对于大段文字的注释，使用 `verbatim` 包的 `comment` 环境

1.7 列表

1.7.1 基本列表

L^AT_EX 三种基本列表环境：无序，有序，描述列表。

- 无序列表
- 实例

1. 有序列表
2. 演示列表

列表 列表可单独使用，可相互嵌套

C++ 编程语言

Java 编程语言

HTML 编程语言

¹这是个脚注

其他列表：上述列表的行间距较大，为节省空间可以用 `paralist` 宏包，它提供了一系列压缩列表和行间列表环境。

压缩列表：+`compact`

- 无序的压缩列表
 - 实例
 - HTML
1. 有序的压缩列表
 2. HTMLo

行间列表：+ 前缀 `inpara-`

- 行间列表 • 显示在一行中
1. 行间有序
 2. 列表

1.7.2 定制列表

改变列表的符号和编号

1

```
\renewcommand{command}{def}
```

改变符号后的表格

- 无序列表
 - HTML
1. 有序列表
 2. HTML

1.8 盒子

1.8.1 初级盒子

相似与 HTML 和 CSS 中的模型。最简单的盒子命令为：`\mbox` 和 `\fbox`。前者把对象组合，后者则价格外框。010 6278 5001 010 6278 5001

1.8.2 中级盒子

稍微复杂的盒子为，`\makebox` 和 `\framebox` 命令提供宽度和对齐方式选项。居中（默认），居左，右，两端对齐 `[c,l,r,s]`。

宽度，对齐方式，内容

两	端	对	齐
---	---	---	---

1.8.3 高级盒子

大一些的对象，比如整个段落可用 `\parbox` 命令或 `minpage` 环境，两者类似。

语法格式：`[外部对齐](t,c,b 居顶，中，底对齐)[高度][内部对齐] 宽度内容`

高级盒子试验模型，与周围对象的纵向关系为外部对齐。

	盒子的内容, fbox 带边框
1	\fbox{
2	\parbox[c][36pt][t]{170pt}{
3	盒子的内容
4	}
5	}

1.8.4 交叉引用

(cross reference), 对前面的最近一个对象的编号或页码引用。

1	一个标签 \label{marker}
2	第\pageref{marker}页\ref{marker}节
3	其中, marker为标签名

标签 第10页1.8.4节, 注意标签第一次编译后会提示警告信息, 第二次才有正确的结果。

2 字体

字体的概念可以划分为三个层次:

1. 编码层, 字符(包括字母, 数字, 控制码等)的索引和编码, 即字符集(character)和字符编码(character encoding)
2. 格式层, 字型(glyph)的定义描述方法, 字体的文件储存格式
3. 显示层, 字体的外在表现形式, 如字体的样式或具体的字体

2.1 字符集和编码的故事

微软将 IBM 的代码页称为 OEM 代码页, 自己定义的称为 ANSI 代码页, 比如 936 (GBK 简体中文), 950 (Big5 繁体中文)。1981 年, 中国大陆推出第一个自己的字符集标准 GB2312, 94×94 的表, 包括 7445 个字符。GB2312 通常采用双字节 EUC-CN 编码, 所以后者又称为 GB2312 编码。GB2312 在 1993 年被拓展为 GBK, 包含 21886 个字符。GBK 不是正式的标准, 2000 年发布的 GB18030 包含 70244 个字符, 采用四字节的编码。

1990 年 ISO 推出了通用字符集(universal character set, UCS), 即 ISO10646。UCS 有两种编码: 双字节的 UCS-2 和四字节的 UCS-4。除了 ISO 外, 统一码联盟(The Unicode Consortium), 它于 1991 年推出了 Unicode 1.0, 其后双方合并, Unicode 2.0 版开始采用与 ISO 10646-1 相同的编码。

Unicode 主要有三种编码: UTF-8, UTF-16, UTF-32。UTF-8 使用 1-4 个 8 位编码, UTF-16 使用 1-2 个 16 位编码, 和 ASCII 不兼容。UTF-32 用 1 个 32 位编码。IETF 要求所有的网络协议都支持 UTF-8, ICM 也建议所有的电子邮件软件都支持 UTF-8。

2.2 字体格式

2.2.1 点阵和矢量字体

电脑字体的数据格式可分为 3 大类：点阵（bitmap）字体，轮廓（outline）字体 [矢量字体] 和笔画（stroke-based）字体。

点阵字体通过点阵描述字体；矢量字体通过一组直线段和曲线描述字形，轮廓字体的缺陷在于其所采用的贝塞尔曲线在光栅设备（如显示器和打印机）不能渲染，故需要额外的补偿处理；笔画字体其实也是轮廓字体，不过其描述的不是完整字形，而是笔画，多用于东亚文字。

2.2.2 常见的字体格式

轮廓字体格式：Type1, TrueType, OpenType。Type1 采用三次的贝塞尔曲线，TrueType 采用二次贝塞尔曲线，处理比三次曲线快，但需要更多的点描述。OpenType 被认为是 Type1 和 TrueType 的超集，既可以使用二次曲线也可以使用三次曲线。

2.2.3 常见字体

TEX 的缺省字体是 Knuth 用 METAFONT 生成的 Computer Modern; XETEX 的缺省字体是 1997 年 AMS 发布的 Latin Modern，基于 Computer Modern，但是扩展了字符集，其封装格式有 Type1 和 OpenType。

2.3 字体的应用

LaTeX, pdflatex, xelatex 编译程序，dvips 和 dvipdfmx 驱动，DVI 浏览器等分别采用不同的字体技术路线。

2.3.1 早期技术

latex 和 DVI

用 latex 生成 DVI 时只需要 TFM 文件，DVI 并不包含字形信息，而只包含对字体的引用。DVI 浏览器显示 DVI 时一般使用 PK，找不到则调用 METAFONT 在后台生成。

dvips

缺省时，dvips 也会查找.pk，没有则自动生成。

2.3.2 XeTeX

XETEX 用一个 XML 文件记录系统字体路径，MikTeX 用的是 localfonts.conf，TeXlive 用的是 fonts.conf。设置字体时需要字体的引用名，和字体的文件名是不同的概念。

XETEX 提供的字体命令比较原始，繁琐，而 fontspec 宏包提供了较好的封装。

3 数学模式

在序言部分加载 amsmath 宏包，提供数学的排版功能。数学模式有两种形式：行间（inline）模式（在正文中插入数学内容）和独立（display）模式（独立排列，可以有编号或没有编号）。

- 1 行间公式用`$$...$$`
- 2 无编号的独立公式用`\[...]`
- 3 建议不用`$$...$$`，会和AMS-Latex有冲突

$$E = mc^2$$

$$E = mc^2 \quad (1)$$

段内的文本你，此处为 `math:` $\sum_{i=1}^n$

公式中的文字用 `textrm` 命令，段外的文本，此处为 `displaymath:` $\sum_{i=1}^n$

3.1 基本元素

3.1.1 希腊字母

希腊字母需要用命令输入，大写的希腊字母的命令首字母也是大写。 $\alpha \Gamma \Delta$

$$\epsilon \zeta \Theta$$

3.1.2 上下标和根号

$$\theta = \sigma + \rho \quad (2)$$

$$x_{ij}^2 \quad \sqrt{x} \quad \sqrt[3]{x}$$

3.1.3 分数

用`\frac` 命令表示，可根据环境自动调整字号，如行间公式中中小一点，独立公式中大一点。也可以设置分数字号，如`\dfrac` 把分数的字号设置为独立公式中的大小，`\tfrac` 则将其设置为行间公式的大小。

$$\frac{1}{2}$$

$$\frac{1}{2} \frac{1}{2}$$

3.1.4 运算符

小运算符可直接输入如 `+` `-` `*` `/` `=` 等可以直接输入。

$$\pm \times \div \cdot \cap \cup \geq \leq \neq \approx \equiv$$

和，积，极限，积分等大运算符用`\sum`，`\prod`，`\lim`，`\int` 等命令。

- 1 quad是空格
- 2 `\sum_{i=1}^n i`
- 3 `\prod_{i=1}^n`
- 4 `\lim_{x \rightarrow 0} x^2`

```

5 \int_a^b x^2 dx $
6 $\sum\limits_{i=1}^n i
7 \prod\limits_{i=1}^n j
8 \[\sum\limits_{i=1}^n i
9 \prod\limits_{i=1}^n \] $

```

说明: quad 是空格

$$\sum_{i=1}^n i \quad \prod_{i=1}^n \quad \lim_{x \rightarrow 0} x^2 \quad \int_a^b x^2 dx \quad \sum_{i=1}^n i \quad \prod_{i=1}^n j$$

$$\sum_{i=1}^n i \quad \prod_{i=1}^n \quad \lim_{x \rightarrow 0^-} x^2 \quad \int_a^b x^2 dx$$

多重积分表示:

```

1 \iint 二重积分
2 \iiint 三重积分

```

$$\int_{-\infty}^{+\infty} x^3 dx$$

$$\lim_{\lambda \rightarrow 0} \sum_{i=1}^n f(\xi_i, \eta_i) \Delta \sigma_i = \iint_{D_{xy}} f(x, y) d\sigma$$

$$\iiint_{\Omega} x^3 dx dy dz$$

箭头: \rightarrow \Rightarrow `\xleftarrow` 和 `\xrightarrow`。可扩展的箭头:

$$\overleftarrow{\frac{x+y+z}{x < y}}$$

3.1.5 注音和标注

数学注音符号 (accent), 列出一些长的标注符号。

$$\bar{x} \quad \vec{x} \quad \hat{x} \quad \check{x} \quad \dot{x} \quad \tilde{x} \quad \acute{x} \quad \grave{x} \quad \ddot{x}$$

速度, 加速度:

$$\dot{x} \quad \ddot{x} \quad \ddot{\ddot{x}}$$

3.1.6 分隔符

各种括号 $()$, $[]$, $\{\}$, 尖括号 $\langle \rangle$ 。

\overline{xxx} \overleftarrow{xxx} \underline{xxx} \overrightarrow{xxx} \overbrace{xxx} \widehat{xxx} \widetilde{xxx} `amsmath` 宏包推荐公式中用: $|$ $|$ $\|$ $\|$ 代替 \LaTeX 中的 $|$ 。

同时可以在分隔符前面加 `\big` `\Big` `\bigg` `\Bigg` 等调整其大小。

$$(\,,\left(\,,\left(\,,\left(\right.\right.\right.$$

3.1.7 省略号

省略号用`\dots` `\cdots` `\vdots` `\ddots` 等命令表示。

```
1 \dots 下标省略号
2 \cdots 中部的省略号
3 \vdots 竖向的省略号
4 \ddots 斜向的省略号
```

$$x_1, x_2, \dots, x_n \quad 1, 2, \dots, n \quad \vdots \quad \ddots$$

3.1.8 空白间距

```
\, 3/18em; ||
\: 4/18em; ||
\; 5/18em; ||
\quad 1em; | |
\qquad 2em; | |
\! -3/18em; ||
```

3.1.9 矩阵

数学模式下可用 `array` 环境生成矩阵。外部对齐方式 (t,c,b)，列对齐方式 (l,c,r)。`\\` 用来分隔行和 `&` 用来分隔列。

```
1 \begin{array}[外部对齐]{列对齐}
2 行列内容
3 \end{array}
```

$$\begin{array}{ccc} x_1 & x_2 & \dots \\ x_3 & x_4 & \dots \\ \vdots & \vdots & \ddots \end{array}$$

`amsmath` 的 `pmatrix`, `Bmatrix`, `vmatrix`, `Vmatrix` 等环境可以在矩阵两边家伙加上各种分隔符，但它们没有对齐参数。

`\smallmatrix` 命令可以生成行间矩阵。A little matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

3.2 多行公式

对应公式太长导致一行放不下，用 `amsmath` 提供一些多行公式。

3.2.1 长公式

无需对齐的长公式使用 `multline` 环境。需要对齐的长公式使用 `split` 环境，但其本身不能单独使用，必须包含在其他的数学环境中，因此也称为次环境，用`\\`和`&`来分行和设置对齐的位置。

$$x = a + b + c +$$

$$d + e + f + g \quad (3)$$

$$x = a + b + c + \\ d + e + f + g$$

3.2.2 公式组

不需对齐的公式组：`gather` 环境。需对齐的公式组用 `align` 环境。

$$a = b + c + d \tag{4}$$

$$x = y + z \tag{5}$$

$$a = b + c + d \tag{6}$$

$$x = y + z \tag{7}$$

`multline`, `gather`, `align` 等环境都带有 `*` 的版本，不生成公式编号。

3.2.3 分支公式-分段函数的宝典

通常用 `case` 次环境生成分支公式。

$$y = \begin{cases} -x, & x \leq 0 \\ x, & x > 0 \end{cases}$$

3.3 定理和证明

`\newtheorem` 命令可用来定义定理之类的环境，其语法如下：

环境名 [编号延续] 显示名 [编号层次]

`\newtheorem` 环境名 `envname` [caption] within 以下为定制四个环境：定义，定理，引理和推论，在一个 section 内统一编号，引理和推论会延续定理的编号。

```
1 \newtheorem{definition}{定义}[section]
2 \newtheorem{theorem}{定理}[section]
3 \newtheorem{lemma}{引理}[theorem]
4 \newtheorem{corollary}{推论}[theorem]
```

定义 3.1. 引入复数概念： $i^2 = -1$

定理 3.1. 两点之间线段最短

引理 3.2. 从 A 点到 B 点走直线最快

推论 3.3. 从某一点到另一点的最短路径

`amsthm` 宏包提供的 `proof` 环境可用来输出证明，并且在证明的结尾加上 QED 符号。演示如下所示：

证明正定矩阵可逆. $A^*B=C$

□

3.4 数学字体

和文本模式类似，数学模式下也可选用不同的字体样式。`\mathbb` 和 `\mathfrak` 需要 `amsfonts` 宏包，`\mathscr` 需要 `mathrsfs`，汉字不会显示出来，仅限于在数学模式下选用。

$ABCXYZ$ $ABCXYZ$ $ABCXYZ$ $ABCXYZ$ $ABCXYZ$ $ABCXYZ$

$ABCXYZ$ $ABCXYZ$ \mathfrak{ABCXYZ} \mathcal{ABCXYZ}

4 图形

当年 Knuth 开发 Tex 时，GIF,JPEG,PNG,EPS 等格式还未问世，但通过 `\special` 命令，让后面的驱动自行决定怎样处理图形。

4.1 图形概述

4.1.1 图形格式

L^AT_EX 支持点阵格式 JPEG 和 PNG，也支持矢量格式 EPS 和 PDF；对于示意图，首选矢量格式；包含大量自然色彩的图像（如照片）首选 JPEG；人工点阵图像应该选择 PNG。

人们考虑把 PostScript 作为文档中嵌入图形的标准格式，担心嵌入文档的 PostScript 会搞破坏，就产生了 Encapsulated PostScript(EPS)。同样，担心嵌入 HTML 的 ActiveX,Java Applet,JavaScript 中混入恶意代码，故会对它们产生限制，因此 EPS 就成了 L^AT_EX 的标准图格式。

4.1.2 Driver

pdflatex

pdflatex 支持 JPEG, PNG 和 PDF，但不支持 EPS。L^AT_EX 有两个宏包 `epstopdf` 和 `pst-pdf` 可以实时地 (on the fly) 把 EPS 转换为 PDF。然而前者有安全漏洞，后者用法繁琐，用户最好还是用其他软件事先把 EPS 转为 PDF。pdflatex 包含编译和驱动两种功能

dvipdfm(x)

dvipdfm 支持 JPEG,PNG 和 PDF，不支持 EPS，但可以实时调用 Ghostscript 把 EPS 转为 PDF，同时增加了 BMP 的支持。

xdvipdfmx

XeLatex 的缺省驱动 `xdvipdfmx` 支持 BMP,JPEG,PNG,EPS 和 PDF。`xdvipdfmx` 比 `dvips`，`pdflatex` 和 `dvipdfmx` 都好。

4.1.3 图形优化

矢量图形的一个优点是可以无限缩放的，而输出质量不变，尺寸对矢量图形的意义不大。描述矢量图形所需的数据较少，其文件体积也一般较小。

图形尺寸

点阵图像的像素是一种相对尺寸，实际尺寸 = 像素/分辨率 (resolution)，最常用的分辨单位是像素/英寸 (pixel per inch,PPI)，点/英寸 (dots per inch,DPI) 混用。人们倾向于认为 PPI 是图形的分辨单位，DPI 是硬件设备的输出分辨率。

当图形分辨率和输出分辨率不一致时, 会有一个重新采样 (resampling) 的过程: 从高分辨率到低分辨率叫下采样 (downsampling), 反之叫上采样。高分辨率图形遇到低分辨率设备会形成浪费, 低分辨率图形遇到高分辨率设备, 看插值的效果。

点阵图形的分辨率选择: 屏幕上阅读 (72 PPI), 考虑到放大 150 PPI, 高质量的打印 300PPI 足够。考虑在 L^AT_EX 文档中插入图形, 通栏的话宽度是 4.8-5.4in(缺省宽度取决于字体的大小), 仅用于屏幕阅读的话, 宽度 400px 足够; 放大阅读或输出打印需要 800px 或 1600px 足够。

点阵图形尺寸相关的基本操作有: 裁剪 (crop), 改尺寸 (resize), 改分辨率。

1. 裁剪时像素会变少, 分辨率不变, 缺省输出尺寸也就变小。
2. 改变尺寸时像素改变, 分辨率不变, 缺省输出尺寸改变。
3. 单纯改变图形文件分辨率时, 像素不变, 缺省尺寸相应改变

综上, 点阵图形的信息量取决于像素, 图形的分辨率为“建议”缺省输出尺寸, 不影响图形质量。

色彩深度

色彩深度 (color depth) 是每个像素所用颜色的位数。1 位表示 2 色; 2 位表示 4 色, 最早用于 CGA 显卡; 16 位表示 65536 色, 又称为高彩; 24 位表示真彩; 30-48 位表示为深彩, 一般照片用 24 位足够, 常用是将图形的色深从高改低。

4.1.4 图形转换和处理

命令行界面的图像处理软件 ImageMagick, 图形界面用 Paint.NET。

ImageMagick

ImageMagick 包含多个命令程序, 最常见的是 convert。比如下面的命令把 BMP 转换为 PNG:

```
1 convert fig.bmp fig.png
```

因为 Windows 中有个区分格式的同名程序, 故在 Windows 下使用 ImageMagick 则需要全路径, 或在 PATH 环境变量中把 ImageMagick 的路径放到 system32 前面。

```
1 convert fig.bmp -crop 300x200+10+10 fig.jpg
2 %裁剪并转换格式, 截取从(10,10)开始300x200的像素
3 convert fig.jpg -crop 300x200+10+10 -resize 30x20 fig1.jpg
4 %裁剪并缩放
5 convert fig.jpg -resize !300x200 fig1.jpg
6 %强制缩放不考虑长宽比
7 convert fig.jpg -density 300 fig1.jpg
8 %分辨率改为300PPI
9 convert fig.jpg -resample 300 fig1.jpg
10 %分辨率改为300PPI, 像素增加, 缺省输出尺寸维持不变
```

ImageMagick 功能强大, 参数选项很多, 但缩小图像做缩略图时不是很清晰

转换为 EPS

ImageMagick 转换 EPS 的方法如下。如果是 BMP 文件, 最好先压缩 JPEG 或 PNG, 再转为 EPS, 这样生成的 EPS 会比较小。

```
1 convert fig.png eps3:fig.eps
```

另一种方法是用虚拟打印机生成 EPS,它的优点是可以把几乎所有文件“打印”成 EPS。推荐 Bullzip PDF Printer,它可以把各种文件打印成 PS、EPS、PDF、BMP、JPEG、PCX、PNG、TIFF 等格式。

4.2 插入图形

4.2.1 范围框

latex 编译程序处理图形文件时需要范围框 (bounding box)。pdflatex 和 xelatex 出现的较晚。EPS 是一种嵌入图形格式,其范围框如下:

```
1 %!PS-Adobe-3.0 EPSF-3.0
2 %%BoundingBox: 5 5 105 105
```

前两个参数是图形左上角的坐标 (一般为原点),后两个参数是右下角的坐标,缺省的长度单位是 bp。

latex 在编译源文件时通过范围框为插图预留空间;它输出的 DVI 只记录图形尺寸和文件名,具体的图形处理由后面驱动负责,无范围框时,latex 则会报错。

两种方法为点阵图形提供范围框:1. 准备一个单独的范围框文件;2. 在插入图形时加范围框参数,用 latex 推荐用第二种方法。dvipdfm 附带的 ebb 程序可以检查 JPEG 和 PNG,生成范围框文件,bp 值 = 像素/分辨率 *72。

4.2.2 基本命令

面向 L^AT_EX2 的 graphics 和 graphicx 宏包,后者基于前者,语法更简单,功能更强大。插图命令的基本语法如下:

```
1 \usepackage[dvipdfm]{graphicx}
2 \includegraphics [bb=0 0 300 200]{fig.png}
```

引用宏包时可加驱动选项,使用 latex 缺省为 dvips, dvipdfm(x) 用 dvipdfm;pdflatex 和 xelatex 分别用 pdftex 和 xetex (可以不加驱动)。使用 latex 时,没有生成.bb 文件,需要或加范围框参数。

4.2.3 图形操作

\includegraphics 命令有一些参数选项用于缩放,裁剪等图形操作:

1.width=x,height=y; scale=s;

2.keeppaspectratio 保持图形比例,确保高,宽比例不会失调。

3.angle=a 逆时针旋转的角度,单位度

4.origin=h[l,c,r]v[t,c,b] 旋转中心,缺省在左下。水平方向可选左,中,右 (l,c,r),垂直方向选上,中,下 (t,c,b)。

5.totalheight=h 总高度,最高,最低两点间垂直距离。

6.viewport=x1,y1,x2,y2 可视区域左上角和右下角坐标,缺省单位 bp。

7.trim=l,b,r,t 左,下,右,上四边的裁剪值,缺省单位 bp。

8.clip 是否真正裁剪，配合 viewport 或 trim 使用。

9.page=n 选页，用于多页图形文件。

4.2.4 文件名和路径

```
1 includegraphics [width=60pt]{homer.pdf} 原图
2 includegraphics [width=80pt]{homer.pdf} 放大
3 includegraphics [width=80pt,height=100pt]{homer.pdf} 变形
4 includegraphics [width=80pt,height=100pt,keepaspectratio]{homer.pdf}
5 保持比例不变形
```

```
1 图形旋转
2 \includegraphics [angle=90]{homer.pdf}
3 \includegraphics [angle=90,origin=c]{homer.pdf}
```

```
1 \DeclareGraphicsExtension{.eps,.mps,.pdf,.png}
2 指定后缀列表让编译程序自行查找
3 \DeclareGraphicsRule{*}{eps}{*}{}
4 指出未知后缀的都是EPS
5 \graphicspath{{c:/file/iamge}} 绝对路径
6 \graphicspath{{./img/}} 相对路径
7 \graphicspath{{}} 多个路径
8 综上，前三个为设置的缺省搜索路径，注意文件名和路径名不能有空格
9 路径名用正斜杠/，路径名用/结尾。
```

4.2.5 figure 环境

插图通常需要占据大块的空间，因此在文字处理软件中经常要调整插图的位置。而 figure 环境则可以自动完成这样的任务，这种自动调整位置的环境称为浮动环境 (float)。

```
1 h(ere)t(top)b(bottom)p(page)
2 这里，页顶，页尾，浮动页
3 四个字母都写上表示放哪里都无所谓
4 \begin{figure}[htbp]
5     \centering
6     \includegraphics{1.jpg}
7     \caption{图像标题}
8     \label{fig:1.jpg}
9     \label一般放在标题之后
10 \end{figure}
```

4.2.6 插入多幅图形

并排摆放，共享标题

当我们需要两幅图形并排时，共享标题，可以在 figure 环境中使用两个\includegraphics 命令。

```

1      \begin{ figure }
2          \centering
3          \includegraphics { left .pdf}
4          \includegraphics { right .pdf}
5          \caption{并排排放，共享标题}
6      \end{ figure }

```

并排摆放，各有标题

各有标题，可以使用 minipage 环境，caption 命令会把环绕它的 minipage 环境变成 figure 环境。

```

1      \begin{ figure }[htbp]
2          \centering 使得minipage整体居中
3          \begin{ minipage }{60pt}
4              \centering 使内容整体居中
5                  \includegraphics { left .pdf}
6                  \caption{小标题1}
7          \end{ minipage }
8          \hspace{10pt}
9          \begin{ minipage }{60pt}
10             \centering 使内容整体居中
11                 \includegraphics { right .pdf}
12                 \caption{小标题2}
13             \end{ minipage }
14     \end{ figure }

```

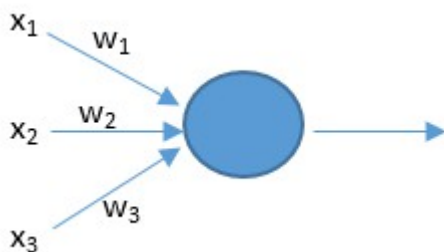


图 1: 小标题 1

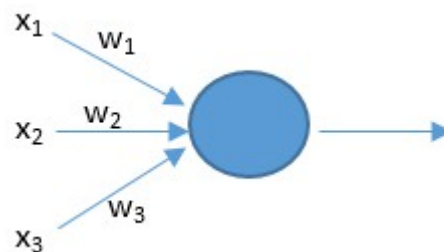


图 2: 小标题 2

共享标题，各有子标题

使用 subfig 宏包，其提供\subfloat 命令。

```

1      \begin{ figure }
2          \centering
3          \subfloat [左边子标题]{
4              \label { fig :a } %避免子标题折行，嵌套minipage
5              \begin{ minipage }[t]{60pt}

```

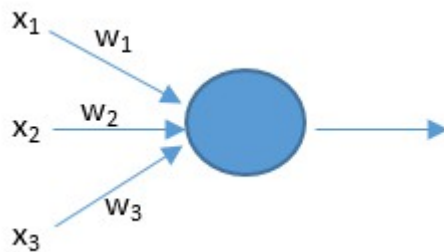


图 3: 插入的图片

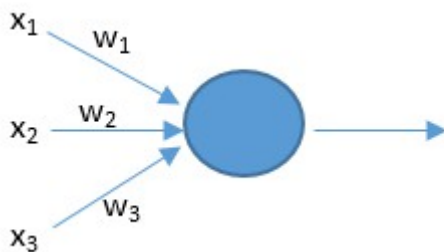
```

6          \centering
7          \includegraphics {2.jpg}
8      \end{minipage}
9  }
10 \hspace{10pt}%
11 \subfloat [右边子标题]{
12     \label { fig : b}
13     \begin{minipage}[t]{60pt}
14         \centering
15         \includegraphics {3.jpg}
16     \end{minipage}
17 }
18 \caption{共享的标题}
19 \label { fig : subfig } %标签在引用或交叉引用的时候使用
20 \end{figure}

```

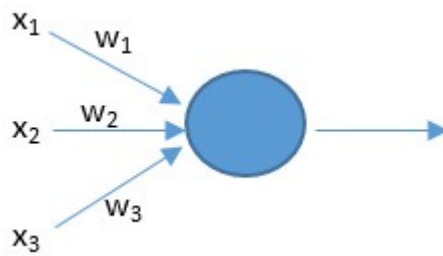


(a) 左边子标题

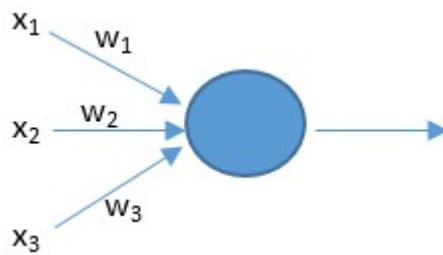


(b) 右边子标题

图 4: 共享的标题



(a) 左边子标题



(b) 右边子标题

图 5: 共享的标题

4.3 矢量绘图

色彩模型分类: 1. 用于照相, 摄影, 电视, 电脑的三原色模型; 2. 用于彩色印刷的 CMYK 四色模型。前者是加色模型, 后者是减色模型。

24 位真彩模型中, 三原色 (红, 绿, 蓝) 各用 8 位表示, 取值从 0 到 255。HTML 常用 16 进制表示。RGB 模型用的为笛卡尔坐标, 色彩变化不够连续。若将其坐标系改为圆柱坐标系, 就是 HSL 和 HSV 模型。

CMYK(蓝, 青, 洋红, 黄) 模型默认纸张原本为白色或浅色背景, 在上面印刷某种颜料就会减少颜色光的反射, 看上去是它的补色。印青色就会得到红色, 印洋红得到绿, 印黄得到蓝, 青、洋红、黄三种颜色都印上就会得到黑色。彩色印刷有分色和套色过程, 如果图形上有黑色直接拿黑颜料印刷会减少成本。CMYK 中的字母 K 就代表 key black。

预定义和自定义颜色

xcolor 宏包预定义颜色: 19 种基本颜色, 68 种 dvips 颜色, 151 种 svg 颜色, 317 种 Unix/X11 颜色。后 3 类颜色使用时需要加相应的预定义颜色集合选项:

```
1 \usepackage[dvipsnames]{xcolor}
2 \usepackage[svgnames]{xcolor}
3 \usepackage[x11names]{xcolor}
```

也可以使用 `\definecolor` 命令来自定义更多的颜色:

```
1 语法: \definecolor{名称}{模式}{参数}
2 \definecolor{d_red}{RGB}{255,0,0}
3 \definecolor{d_green}{HTML}{00FF00}
```

彩色文字

设置文字颜色可以使用 `\textcolor` 命令。

语法: `\textcolor 名称 [[模式] 代码文字`

颜色 红 粉红 灰色字体

彩色盒子

\colorbox 命令可以生成有彩色背景盒子，与\textcolor 类似。\fcolorbox 命令又给彩色盒子加上边框。

```
1 \colorbox{SkyBlue}{}
2 \fcolorbox{Silver}{SkyBlue}{}
3 \fcolorbox{RoyalBlue}{Lavender}{}

```



4.1 Base colors (always available)

black	darkgray	lime	pink	violet
blue	gray	magenta	purple	white
brown	green	olive	red	yellow
cyan	lightgray	orange	teal	

4.2 Colors via dvipsnames option

Apricot	Cyan	Mahogany	ProcessBlue	SpringGreen
Aquamarine	Dandelion	Maroon	Purple	Tan
Bittersweet	DarkOrchid	Melon	RawSienna	TealBlue
Black	Emerald	MidnightBlue	Red	Thistle
Blue	ForestGreen	Mulberry	RedOrange	Turquoise
BlueGreen	Fuchsia	NavyBlue	RedViolet	Violet
BlueViolet	Goldenrod	OliveGreen	Rhodamine	VioletRed
BrickRed	Gray	Orange	RoyalBlue	White
Brown	Green	OrangeRed	RoyalPurple	WildStrawberry
BurntOrange	GreenYellow	Orchid	RubineRed	Yellow
CadetBlue	JungleGreen	Peach	Salmon	YellowGreen
CarnationPink	Lavender	Periwinkle	SeaGreen	YellowOrange
Cerulean	LimeGreen	PineGreen	Sepia	
CornflowerBlue	Magenta	Plum	SkyBlue	

4.3 Colors via svgnames option

AliceBlue	DarkKhaki	Green	LightSlateGrey
AntiqueWhite	DarkMagenta	GreenYellow	LightSteelBlue
Aqua	DarkOliveGreen	Grey	LightYellow
Aquamarine	DarkOrange	Honeydew	Lime
Azure	DarkOrchid	HotPink	LimeGreen
Beige	DarkRed	IndianRed	Linen
Bisque	DarkSalmon	Indigo	Magenta
Black	DarkSeaGreen	Ivory	Maroon
BlanchedAlmond	DarkSlateBlue	Khaki	MediumAquamarine
Blue	DarkSlateGray	Lavender	MediumBlue
BlueViolet	DarkSlateGrey	LavenderBlush	MediumOrchid
Brown	DarkTurquoise	LawnGreen	MediumPurple
BurlyWood	DarkViolet	LemonChiffon	MediumSeaGreen
CadetBlue	DeepPink	LightBlue	MediumSlateBlue
Chartreuse	DeepSkyBlue	LightCoral	MediumSpringGreen
Chocolate	DimGray	LightCyan	MediumTurquoise
Coral	DimGrey	LightGoldenrod	MediumVioletRed
CornflowerBlue	DodgerBlue	LightGoldenrodYellow	MidnightBlue
Cornsilk	FireBrick	LightGray	MintCream
Crimson	FloralWhite	LightGreen	MistyRose
Cyan	ForestGreen	LightGrey	Moccasin
DarkBlue	Fuchsia	LightPink	NavajoWhite
DarkCyan	Gainsboro	LightSalmon	Navy
DarkGoldenrod	GhostWhite	LightSeaGreen	NavyBlue
DarkGray	Gold	LightSkyBlue	OldLace
DarkGreen	Goldenrod	LightSlateBlue	Olive
DarkGrey	Gray	LightSlateGray	OliveDrab

图 6: 颜色表

4.3.1 绘图工具概述

与 \LaTeX 配套使用的矢量绘图工具主要有 3 种: META-POST, PSTricks, PGF。概述如下:

- 工作方式: METAPOST 离线绘图, 生成的 MPS(一种特殊的 EPS); PSTricks 和 PGF 都采用在线绘图的方式, 即在 \LaTeX 文档内直接使用绘图命令。
- 兼容性: METAPOST 生成的 MPS 需要先转换为 PDF 才能被 pdf \LaTeX 使用; PSTricks 生成的 EPS 和 pdf \LaTeX 不兼容; PGF 提供针对各种驱动接口, 兼容性最好。
- 功能: PSTricks 有 PostScript 作后盾, 功能最强; METAPOST 擅长处理数学内容; PGF 擅长流程图。

也可以考虑一些面向 \LaTeX 的绘图前端, 如 Dia 和 Ipe, 或一些更通用的软件, 比如 gnuplot 和 Inkscape。

5 Metapost

METAFONT 只支持黑白, 是用来设计字体的。METAPOST 输出的是 EPS, 而且支持彩色, 同时可以在图形上加文字标注, 甚至插入 TEX 源码。

5.1 准备工作

METAPOST 的缺省长度单位是 bp, 也可以使用其他单位。

```
1      u := 10pt %变量赋值符号为:=
2      beginfig (1); %图形起始
3      ...      %绘图命令
4      endfig;
5      beginfig (2);
6      ...
7      endfig;
8      end
```

变量赋值符号为:=, 且一个变量在同一源文件中只需定义一次。一个 METAPOST 源文件 (.mp)。代码中每行语句以分号结尾, 注释以百分号起始, 回头命令包含在一对起始和结尾声明之间, 文件结尾也有一个即为声明。

我们可以用命令程序 mpost 编译源文件, 生成 MPS(特殊的 EPS), 然后将 MPS 插入 \LaTeX 中使用。

METAPOST 提供一个变量来设置输出文件名, 将下面的代码加到源文件的头部, 编译输出的文件名就会是 “fig-01.mps, fig-02.mps”

```
1      outputtemplate "%j-%2c.mps"; %加在源文件的头部
2      outputtemplate "flowchart.mps" %加在每个图形前面
```

x \LaTeX 不能识别 MPS 格式, 故需要使用 .eps 后缀, 或使用以下命令:

```
1      \DeclareGraphicsRule{.mps}{eps}{.mps}{}{}
```


5.2 基本图形对象

5.2.1 点和线

METAPOST 的缺省直径为 0.5bp，可以用 `withpen` 选项为单个绘图命令设置画笔，也可以用 `pickup` 命令为之后所有的绘图命令设置画笔。用 `draw` 命令将几个点以直线连接起来，`drawdot` 命令在指定坐标画点。

```
filenametemplate "line.eps"; u := 10pt ;beginfig(1); draw (0,0)-(4u,0)-(2u,2u)-(0,0)
withpen pencircle scaled .8pt; pickup pencircle scaled .8pt; draw (5u,0)-(9u,0)-(7u,2u)-
cycle; pickup pencircle scaled 3pt; drawdot (10u,0); drawdot (14u,0); drawdot (12u,12u);
endfig;
```

几段直线或曲线可以构成一条路径 (path)，在路径的末尾加个 `cycle` 命令构成封闭路径 (closed path)。

5.2.2 预定义形状

`fullcircle` 命令是以原点为圆心画的一个单位圆，类似的还有 `halfcircle`, `quartercircle`, `unitsquare` 等。注意单位正方形的参考点在其左下，使用不同方向的的缩放系数 `xscaled` 和 `yscaled`。

5.2.3 曲线

把画直线时坐标点之间换成..，就得到一条曲线。METAPOST 的曲线用三次贝塞尔算法实现，在曲线上使用方向 (direction)，张力 (tension) 和曲率 (curl) 等控制。

```
1      filenametemplate "predefined.eps";
2      beginfig (3);
3      pickup pencircle scaled .8pt;
4      draw fullcircle scaled 2u;
5      draw halfcircle scaled 2u;
6      draw quartercircle scaled 2u shifted (3u,0);
7      draw fullcircle xscaled 4u yscaled 2u shifted (9u,0);
8      draw unitsquare scaled 2u shifted (12u,-u);
9      endfig;
```

5.3 图形控制

5.3.1 线型，箭头，彩色和填充

注意：METAPOST 不能使用 `xcolor` 宏包，只支持 `rgb` 和 `cmyk` 色彩模式，其自定义颜色的方法如下：

```
1      color c [];
2      c1 := .9red + .6green + .3blue;
3      c2 := (.9,.6,.3);
```

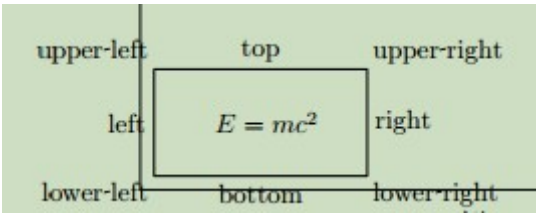
绘图命令一般通过 `withcolor` 选项来使用各种颜色。除了为每个绘图命令单独指定颜色，也可以使用全局命令 `\drawoption`，使得其后的绘图命令都使用某种颜色。

5.4 图形变换

对路径进行平移 (`shifted`)[参数: 移到的坐标点], 旋转 (`rotated`)[参数: 角度, 旋转中心为原点], 定点旋转 (`rotatedaround`)[参数: 旋转中心], 镜像 (`reflectedabout`)[参数: 两点确定的一条直线], 倾斜 (`slanted`)[倾斜比] 等变换操作。

5.5 标注

`label` 在指定位置加文字标注。命令的 8 个后缀, `top`, `bottom`, `lft`, `rt`, `ulft` 左上, `urt` 右上, `llft` 左下, `lrt` 右下。 `dotlabel` 命令加标注的同时画了个点。



1

`label . rt ("文字", (4u, 4u))`

5.6 编程

5.6.1 数据类型和变量

如前面所用的缩放系数 `u` 是 `numeric`, 点的坐标是 `pair`。除缺省的 `numeric` 外, 其他变量在使用时需要用数据类型来显示声明。相同类型的变量可以在一行语句中声明, 但带下标的变量不能放在同一行。

6 PSTricks

PSTricks 是基于 PostScript 的宏包, 可以直接在 \LaTeX 文档中插入绘图命令。以下为可以和 PSTricks 配合使用的辅助宏包:

表 7.1: PSTricks 辅助宏包			
<code>multido</code>	循环	<code>pst-3dplot</code>	三维绘图
<code>pst-node</code>	示意图	<code>pst-solides3d</code>	三维实体
<code>pst-tree</code>	树状图	<code>pst-circ</code>	电路
<code>pst-plot</code>	函数绘图	<code>pst-labo</code>	化学
<code>pst-func</code>	数学函数	<code>pst-geo</code>	地理
<code>pst-eucl</code>	几何函数	<code>pstricks-add</code>	杂项

6.1 Begin

缺省单位是 cm，绘图命令一般放在 `pspicture` 环境中，参数为矩形的左下角和右上角，从左下角的原点开始可以省略该点坐标。

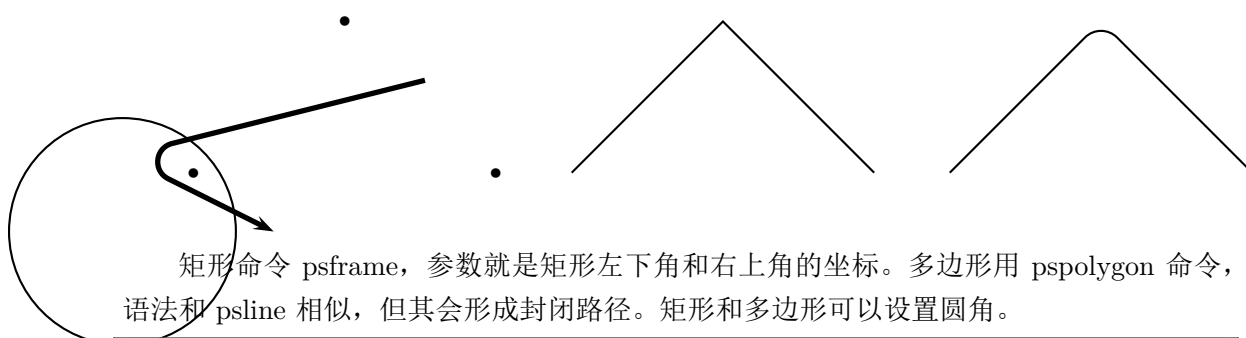
```
1 \paset{unit=10pt}
2 \begin{pspicture}(0,0)(4,2)
3 ...
4 \end{pspicture}
```

`pst-pdf` 宏包生成包含 PSTricks 图形的 EPS，然后根据需要转换为 PDF。每个 `pspicture` 环境中的内容会自成一页，方便插入文档。

6.2 基本图像对象

6.2.1 点和直线

```
1 \begin{pspicture}(-.2,-.2)(14,2.2)
2 \psdot(0,0) %\dot命令画一个点
3 \psdots(4,0)(2,2) %\dots命令画多个点
4 \psline(5,0)(7,2)(9,0)
5 \psline命令把多个点用直线段连接起来，线段间缺省为尖角。
6 \psline[linearc=.3](10,0)(12,2)(14,0)
7 \end{pspicture}
8 %
```



矩形命令 `psframe`，参数就是矩形左下角和右上角的坐标。多边形用 `pspolygon` 命令，语法和 `psline` 相似，但其会形成封闭路径。矩形和多边形可以设置圆角。

```
1 \begin{pspicture}(19,3)
2 \psframe(0,0)(4,3)
3 \psframe[framearc=.3](5,0)(9,3)
4 \pspolygon(10,0)(14,0)(12,3)
5 \pspolygon[linearc=.3](15,0)(19,0)(17,3)
6 \end{pspicture}
```

6.2.2 圆，椭圆，圆弧，扇形

圆形用 `\pecircle`，参数是圆心和半径。椭圆 `psellipse`，参数：中心，长径，短径。

6.2.3 曲线

`pscurve` 曲线命令，平滑曲线连接。`psecurve` 命令显示曲线的两个端点；`psscure` 命令则把曲线封闭起来，`showpoints` 参数用来指示是否显示曲线的构成点。贝塞尔曲线用 `psbezier` 命令，参数是曲线的控制点。抛物线用 `psparabola` 命令，参数：抛物线通过某点，抛物线的顶点。

6.2.4 网格和坐标轴

制图用到坐标和网络。`psgrid` 命令输出矩形网格，三个参数点。坐标标注在通过第一点的两条直线上，第二和第三点是矩形的两个对角顶点。当第一个参数省略时，坐标标注在通过第一个顶点的两条矩形边上。坐标轴用 `pst-plot` 宏包的 `\psaxes` 命令。

7 PGF

7.1 准备工作

一般不直接使用 PGF 底层命令，而是通过 TikZ 来调用。在引用 `tikz` 宏包之前，需设置 PGF 系统驱动。比如 `dvipdfmx` 的设置方法如下，使用 `pdflatex` 和 `xelatex` 时，知道驱动是谁。

```
1 \def\pgfsysdriver{\pgfsys-dvipdfmx.def}
2 \usepackage{tikz}
```

PGF 的缺省长度单位是 `1cm`，TikZ 提供 `\tikz` 命令和 `tikzpicture` 环境，具体的绘图指令可以放在 `\tikz` 后面，也可以放在 `tikzpicture` 环境，两者效果相同。

```
1 \tikz ....% 绘图命令
2 \begin{tikzpicture}
3     .... % 绘图命令
4 \end{tikzpicture}
```

为节省编译时间，可以用 `preview` 宏包生成独立图形文件，虽然这样做不是必须的。

```
1 \documentclass{article}
2 \usepackage[active,tightpage,xetex]{preview}
3 \usepackage{tikz}
4
5 \begin{document}
6     \begin{preview}
7         \begin{tikzpicture}
8             ...
9         \end{tikzpicture}
10    \end{preview}
11 \end{document}
```

7.2 基本图形对象

7.2.1 直线和矩形

PGF 绘图命令语法和 METAPOST 类似, `\draw` 称为一个命令, 它后面的`-`(用来画直线), `cycle`(用来封闭路径), `rectangle` 等操作, `[rounded corners]` 称为一个选项, 用来加圆角。

```
1 \draw (0,0)--(4,0)--(2,2)--(0,0) %起始点是(0,0);
2 \draw (5,0)--(9,0)--(7,2)--cycle %起始点(7,2);
3 \draw [rounded corners] (10,0)--(14,0)--(12,2)--cycle;
4 \draw (15,0) rectangle (19,2);
5 %起始点(15,0), 另一对角顶点(19,2);
6 \draw [rounded corners] (20,0) rectangle (24,2);
```

7.2.2 圆, 椭圆, 弧形

7.2.3 曲线

7.2.4 网格

7.3 图形控制

7.3.1 箭头

7.3.2 线宽和线性

7.3.3 颜色和填充

7.3.4 渐变和阴影

7.3.5 样式

7.4 图形变换

7.5 示意图

7.5.1 节点

PGF 中的节点 (node) 可以是简单的标签, 也可以各形状的边框, 还可以有各种复杂的属性。

7.5.2 流程图

7.5.3 树

8 表格

8.1 简单表格

`tabular` 环境提供最简单的表格功能, `\hline` 命令表示横线, `|` 表示竖线; `&` 表示分列。
`tabular` 环境语法: `[纵向对齐] { 横向对齐和分隔符 }`

操作系统	发行版	编辑器
Windows	MikTeX	TexMakerX
Unix/Linux	teTeX	Kile

表格也有类似图像的浮动环境 `table`，可把表格简化为三线表。三个横线一样粗细，也可以使用 `booktabs` 宏包，分别用 `\toprule`, `\midrule`, `\bottomrule` 等命令表示。

操作系统	发行版	编辑器
Windows	MikTeX	TexMakerX
Unix/Linux	teTeX	Kile

8.2 表格宽度控制

需要具体某列的宽度，将其参数从 `l,c,r` 改为 `p` 宽度，此时纵向对齐方式为居顶。使用宽度控制参数之后，表格内容缺省居左对齐。可使用列前置命令的语法：`> { 命令 } 列参数`。列前置命令仅对紧邻其后的一列有效。

操作系统	发行版	编辑器
Windows	MikTeX	TexMakerX
Unix/Linux	teTeX	Kile

控制整个表格的宽度，可以使用 `tabularx` 宏包的同名环境，其中 `X` 参数表示可以折行（自动换行）。语法：`{ 表格宽度 } { 横向对齐, 分隔符, 折行 }`，如果想把纵向对齐方式改为居中和居底，可以使用 `array` 宏包，提供两个对齐参数：`m` 宽度，`b` 宽度。

```

1 \begin{table}
2   \centering
3   \begin{tabularx}{350pt}{IXIX}
4     %350pt总宽度，四列对齐方式，
5     \toprule
6     \midrule
7     \bottomrule
8   \end{tabularx}
9 \end{table}

```

设置列宽为 p2cm 超过 宽度会自动 换行	可以折行的 第二列,无边 落木萧萧下, 无尽长江滚 滚来	不可折行的 第三列	可折行的第 四列,欲把西 湖比西子,人 面桃花相映 红
kk	lk	opo	oo
ff	hh	jjj	ghh

8.3 跨行跨列

表格单元需要横跨多列，使用 `\multicolumn` 命令。同时使用 `booktabs` 宏包的 `\cmidrule` 命令横跨几列横线。

语法: `\multicolumn { 横跨列数 } { 对齐方式 } { 内容 }`

语法: `\cmidrule {起始列-结束列}`

语法: `\multirow {竖跨行数} {跨度} {内容}` 跨行的表格可以使用 `multirow` 宏包的 `\multirow` 命令。

常用工具		
操作系统	发行版	编辑器
Windows	MikTex	TexMakerX
Unix	teTex	Kile

8.4 数字表格

表格包含大量数字时, 手工调整小数点和位数对齐很麻烦, 可使用 `warpcol` 宏包, 为 `tabular` 环境提供了列对齐参数 `P`, 语法: `P {-m.n}`, 其中 `m` 和 `n` 分别是小数点前后的位数, 数字前的负号可选。使用 `multicolumn` 可以保护表头, 防止被 `P` 参数误伤。

数学常数	物理常数
3.14159	2.99792
27.18281	-17.58819

8.5 长表格

表格太长跨页时, 可以用 `longtable` 宏包。具体工作如下:

1. 用 `longtable` 环境取代 `tabular` 环境;
2. 在表格开始部分定义每页首页出现的通用表头, 表头最后一行末尾不用 `\\` 换行, 而是加一个 `\endhead` 命令;
3. 定义首页表头 (和通用表头不同时), 最后一行的结尾为 `\endfirsthead` 命令结尾。
4. 以 `\endfoot` 命令结尾的通用表尾;
5. 以 `\endlastfoot` 命令结尾的末页结尾 (如果和通用表尾不同的话);
6. 表格的具体内容;

表 1: 长表格演示

作者	作品
白居易	浔阳江头夜送客, 枫叶荻花秋瑟瑟。 主人下马客在船, 举酒欲饮无管弦。 醉不成欢惨将别, 别时茫茫江浸月。 主人下马客在船, 举酒欲饮无管弦。 醉不成欢惨将别, 别时茫茫江浸月。 主人下马客在船, 举酒欲饮无管弦。 醉不成欢惨将别, 别时茫茫江浸月。 主人下马客在船, 举酒欲饮无管弦。 醉不成欢惨将别, 别时茫茫江浸月。

下一页 ...

作者	作品
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。
	主人下马客在船，举酒欲饮无管弦。
	醉不成欢惨将别，别时茫茫江浸月。

8.6 宽表格

表格太宽时用 rotating 宏包，用 sidewaysstable 环境替代 table 环境即可。

8.7 彩色表格

用 colortbl 宏包，提供 \columncolor,rowcolor,cellcolor 分别设置列，行，单元格的顏色。columncolor 需要放到前置命令里，rowcolor,cellcolor 分别放到行，单元格之前。colortbl 宏包可以使用 xcolor 宏包的色彩模型；两者同时，则前者不能直接加载，需要通过后者的选项 table 来加载。三个命令同时使用，顺序为：单元格，行，列。

表 2: 主流英文词典

Title	Abbr	Publisher	Year	Pages	Entries	Price
Oxford English Dict, 2nd Ed	OED	OxfordUniv	1989	21728	616500	995
Shorter Oxford English Dict,7th Ed	SOED	OxfordUniv	2007	3888	600000	175

操作系统	发行版	编辑器
Windows	MikTeX	TexMakerX
Linux	teTeX	Kile
通用	TeX Live	TeXworks

xcolor 宏包的 rowcolors 命令可以设置奇偶行的颜色，语法：{起始行} {奇数行颜色} {偶数行颜色}。hiderowcolors 命令用来暂停显示前面设置的奇偶行颜色，否则后续其他表格会继续显示颜色，另一个命令 showrowcolors 可以重新激活奇偶行颜色设置。

9 文档的结构

正文的层次结构，前目录，后索引。长文档通常分为多个文件，与 HTML 文件类似，PDF 也提供超链接功能，通常版权页为偶数页，目录首页为奇数页等。可以在主控文档中引用子文档，\include 命令会新起一页，不想用新页可以改用 \input 命令。

```

1 %master.tex
2 \begin{document}
3   \include{chapter1.tex}
4   \include{chapter2.tex}
5 \end{document}
```

拆分长文档，当文档很长时，编译一遍需要很长的时间，可以使用 syntonly 宏包，这样编译时只检查语法，而不是生成结果文件。

```

1 \usepackage{syntonly}
2 ...
3 \syntonly
```

9.1 标题

标准文档没有为作者所属单位定义的专门命令。只有 title,author,date 命令。设置完上述内容后，notitlepage 和 titlepage 用来控制标题是否单独占一页。report 和 book 文档类的标题缺省独占一页，article 文档类的标题缺省和正文共占一页。

9.2 目录

\tableofcontents 命令用来生成整个文档的章节目录，可以用 \setcounter 命令来指定目录的层次深度。不想显示某个章节目录的，可以用 * 来声明章节。同理，用 \listoffigures 和 \listoftables 命令来生成图目录和表目录。注意，当结构层次变化时，需要编译两次获得新的目录。

第一次编译生成一些中间文件，后缀分别为.toc(目录)，.lof(图目录)，.lot(表目录)；第二次编译则把中间文件和其他内容整合起来。利用宏包 caption 自定义类似于插图和表格的浮动环境。

自定义浮动环境-> 语法：\DeclareCaptionType[选项][环境][名称][目录名]

```

1 \DelcareCaptionType[ fileext =loe]{example}[例][例目录]
2 %指定的loe为例目录中间文件后缀，example为环境名
```

```

3 %例为浮动环境的标题前缀，例目录为目录的标题
4 \begin{example}[h]
5 ...
6 \end{example}

```

9.3 参考文献

9.3.1 thebibliography

LaTeX 最原始的方法是用 thebibliography 环境和 \bibitem 命令来定义参考文献条目。thebibliography 环境一般放在文档的末尾，定义好参考文献后，可以用 \cite 命令在正文中引用条目。

```

1 \begin{ thebibliography }
2 \bibitem{Rowling_1997}{9}
3 %参数9表示参考文献条目编号的宽度
4 Jock K. Rowling,
5 \emph{Harry Potter}.
6 Bloomsbury,London,
7 1997.
8 \end{ thebibliography }

```

```

1 %引用定义好的文献
2 \ cite {Rowling_1997}

```

9.3.2 BibTeX

thebibliography 需要用户自己调整显示格式。另一条路：用数据库文件.bib 记录参考文献目录，用样式文件.bst 设置显示格式。一般只需维护数据库即可。秉承 LaTeX 内容和格式分离的思想，同理在 SGML/DSSSL，HTML/CSS，XML/XSL 等技术思路相同。

BibTeX 将参考文献分为十几种类型，每种类型的参考文献有必选项和可选选项。具体如下：

manual 手册

- 必选项：title
- 可选项：author,organization,address,edition,month,year,note

编辑.bib 文件时可以用普通文本编辑器，也可以用专门的文献管理软件提高效率，如 JabRef。一些其他的文献管理软件或网络服务可以输出为.bib 格式，如 Endnote,Zotero 等。

```

1 @book{Rowling_1997,
2 author = "Joanne K.Rowling",
3 title = "Harry Potter and the Sorcerer ",
4 publisher = "Bloomsbury,London",
5 year = "1997"
6 }

```

```

7 %其中每行是一个数据项，第一个数据项是关键字，供引用时用；
8 其他的数据项以 名称 = 值的形式出现，值要写在双引号内；数据项间用逗号隔开。

```

有了数据后，L^AT_EX 发行版提供 4 种标准的样式，具体如下：

1. **plain**: 参考文献列表按作者姓氏排序，序号为阿拉伯数字。
2. **unsrt**: 参考文献列表按正文引用顺序排序，序号阿拉伯数字。
3. **aloha**: 参考文献列表按照作者姓氏排序，序号为作者姓氏加年份。
4. **abbrv**: 类似 plain 样式，作者名字，月份，期刊等用缩写。

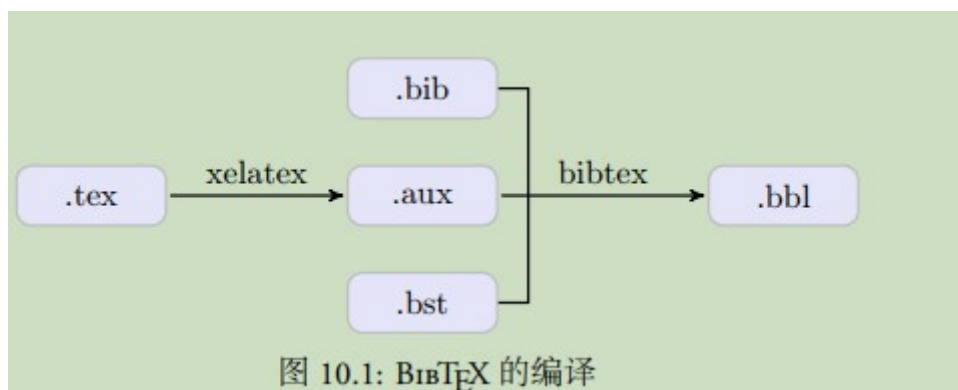
选定样式后，需要在文档中用 `\bibliographystyle` 命令设置样式，然后用 `\bibliography` 命令用来输出参考文献列表。前文中的交叉引用的文档需要编译两遍，含参考文献的文档则需要执行 4 次编译操作。

```

1 \bibliographystyle {plain}
2 \bibliography {myref}

```

1. 第一遍 xelatex 把参考文献条目的关键字写到中间文件.aux 中去。
2. bibtex 根据.aux,.bib,.bst 生成一个.bbl 文件，即参考文献列表。它的内容就是 thebibliography 环境和一些\bibitem 命令。
3. 第二遍 xelatex 把交叉引用写到.aux 中去。
4. 第三遍 xelatex 则在正文中正确地显示引用。



有多个子文档时，可以在每个子文档中用 `\bibliography` 命令设置不同的样式。建议使用统一的样式，用 xelatex 编译主控文档，用 bibtex 编译各个子文档。

9.3.3 Natbib

参考文献在正文中的引用通常有两种模式：作者-年份和数字。L^AT_EX 提供的 `\cite` 命令只支持数字模式，而 natbib 宏包则支持两种模式。

natbib 提供了 3 种列表样式：plainnat,abbrvnat,unsrtnat，它们的文献参考列表和相应的 L^AT_EX 标准样式 plain,abbrv,unsrt 效果相同，只是在引用时可以自由选择作者-年份或数字模式，三种列表样式都有自己的缺省样式，如果定制引用样式，可以使用 `\setcitestyle` 命令。

natbib 提供多种引用命令，最基本的是 `\citet` 和 `\citep`，一般不使用 L^AT_EX 本身提供的 `\cite` 命令，其在作者-年份模式下和 `\citet` 效果相同，在数字模式下和 `\citep` 相同。

引用模式	authoryear,number,super
括号	round,square,open=char,close=char
引用条目分隔符	分号，逗号，citesep=char
作者年份分隔符	aysep=char
共同作者年份分隔符	yysep=char
注解分隔符	notesep=tey

9.4 索引

makeidx 宏包提供索引命令，\makeindex 命令；注意索引关键字在全文中保持唯一性，一般在文章末尾打印索引。

```

1 \usepackage{makeidx}
2 \makeindex
3 ...
4 \begin{document}
5   \index{索引关键字}
6 \end{document}
```

当编译含索引的文档时，用户需要执行 3 次编译操作。

1. 第 1 遍 xelatex 把索引条目写到一个.idx 文件中。
2. makeindex 把.idx 排序写到一个.ind 文件中。
3. 第 2 遍 xelatex 在 \printindex 命令的地方引用.ind 的内容，生成正确的内容。



9.5 超链接

hyperref 宏包提供超链接功能，给文档的内部交叉引用和参考文献加上超链接，\hyperref 命令对已经定义的标签进行简单包装，\url 和 \href 也可以定义外部链接。

```

1 \label{sec:hyperlink}
2 ..
3 编号形式的链接：\ref{sec:hyperlink} \\
4 文字形式链接：\hyperref[sigma]{sec:hyperlink}{链接}
```

```

1 \url{http://www.dralpha.com/} \\
2 \href{https://baidu.com/}{百度}
```

article 期刊或杂志上的文章

- 必选项: author, title, journal, year
- 可选项: volume, number, pages, month, note

conference 同 inproceedings

book 正式出版的书籍

- 必选项: author/editor, title, publisher, year
- 可选项: volume/number, series, address, edition, month, note

booklet 非正式出版的小册子

- 必选项: title
- 可选项: author, howpublished, address, month, year, note

inbook 书的一部分, 比如章、节, 或某些页

- 必选项: author/editor, title, chapter/pages, publisher, year
- 可选项: volume/number, series, type, address, edition, month, note

incollection 书中比较独立的一部分

- 必选项: author, title, booktitle, publisher, year
- 可选项: editor, volume/number, series, type, chapter, pages, address, edition, month, note

inproceedings 会议论文

- 必选项: author, title, booktitle, year
- 可选项: editor, volume/number, series, pages, address, month, organization, publisher, note.

mastersthesis 硕士论文

- 必选项: author, title, school, year
- 可选项: type, address, month, note

misc 实在不好分类时只好用它

- 必选项: 无
- 可选项: author, title, howpublished, month, year, note

phdthesis 博士论文

- 必选项: author, title, school, year
- 可选项: type, address, month, note

proceedings 会议论文集

- 必选项: title, year
- 可选项: editor, volume/number, series, address, month, organization, publisher, note

techreport 技术报告

- 必选项: author, title, institution, year
- 可选项: type, number, address, month, note

unpublished 未出版文档

- 必选项: author, title, note
- 可选项: month, year

9.6 结构名

每个文档都有自己的名字，一般用来在标题或引用时显示。比如：主目录 (Contents), 图目录 (List of Figures), 表目录 (List of Tables); 章 (chapter), 节 (section), 小节 (subsection); 图 (figure), 表 (Table)。

```
1 \renewcommand{\contentsname}{目录}
2 \renewcommand{\listfigurename}{图目录}
```

10 布局

在 L^AT_EX 排版对象都是一个盒子，盒子间的相互嵌套到更大的盒子中，排版时页面是最大的盒子，纸张规格有两大标准：公制和美制。

10.1 历史

1786 年，德国科学家就发现 $\frac{1}{\sqrt{2}}$ 这个比例可用于分割纸张，在较长的方向一分为二，得到的两张纸也是同样比例。德国标准化学会 (DIN) 在 1922 年发布 476 纸张标准，从面积 1 m^2 的 A0(841mm×1189mm) 开始，每次减半长边。1961 年 ISO 将 A 和 B 系列采纳为推荐标准，1975 年变为 ISO216 标准，其中的 B 系列比 DIN476 略大，从 1000mm×1414mm 的 B0 开始，1985 年发布的 ISO269 加上 C 系列，尺寸是 A 系列和 B 系列纸张尺寸的几何平均。

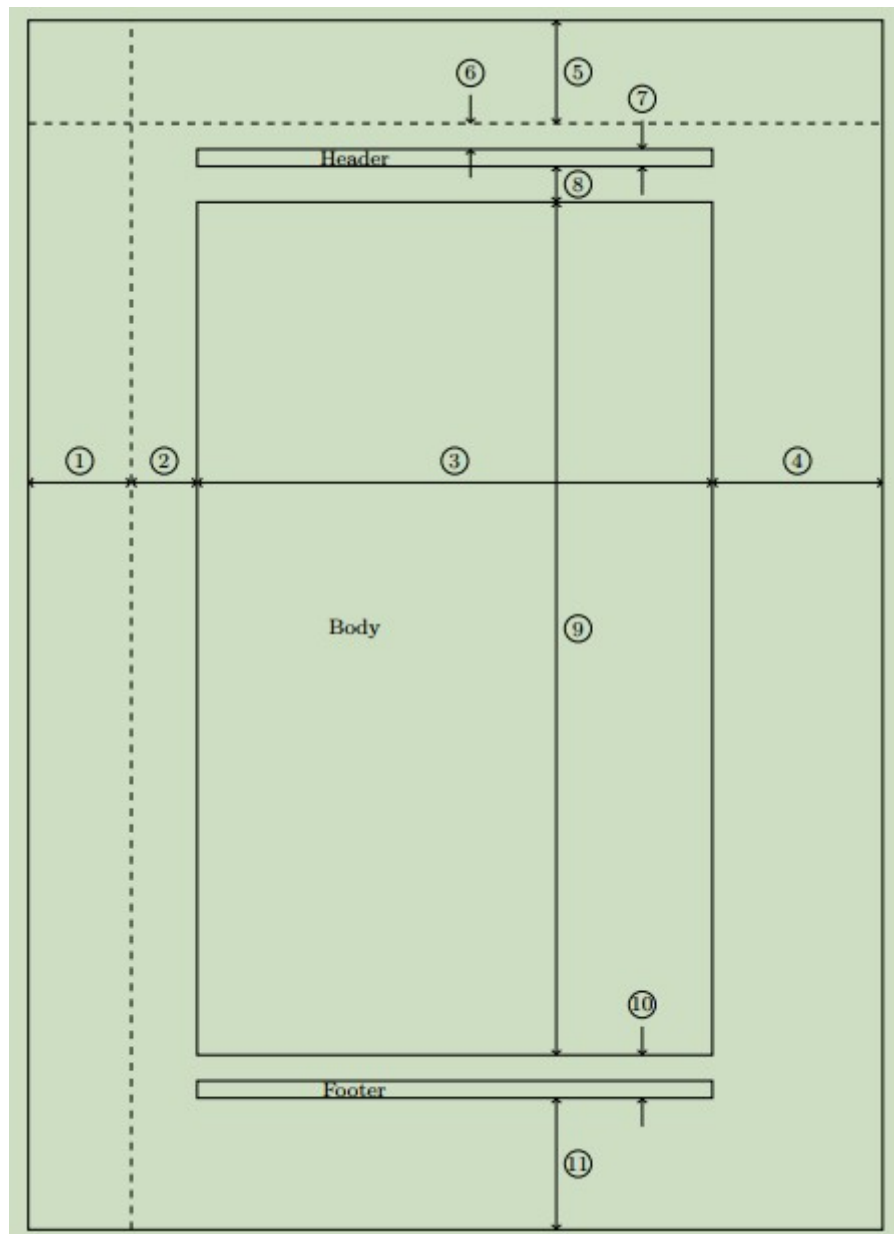
A 系列常用于公文；B 系列常用于海报和护照 (B7,88mm×125mm)；C 系列常用于信封，恰好比 A 系列大一点，如 A4 纸可以装在 C4 信封里，对折一下可以装进 C5 信封，再对折进 C6 中。

大多数国家采用 ISO 标准，美国和其他几个国家使用比 A4 宽一点的 Letter(8.5in×11in)，Legal(8.5in×14in) 主要用于法律文件，比普通文件长一大截。1996 年美国推出 ANSI Y14.1 定义了 A,B,C,D,E 规格。A 即 Letter，B 比 A 的面积大一倍，C 比 B 大一倍，以此类推。它们的长宽比不一致，B 和 C 比其他三种窄的多，它们的尺寸倒是和 A4-A0 差不多。

10.1.1 尺寸详解

图为一张 A4 版，尺寸 210mm×297mm(597pt×845pt)，body 为正文区域，Header 是页眉，Footer 为页脚。图片尺寸介绍如下：

1. 页边距，1in
2. \oddsidemargin 或 evensidemargin，奇数或偶数页左边距，46pt
3. \textwidth，正文宽度 360pt，约 32 汉字
4. 597pt 减去左边的 1in+46pt 和中间的 360pt，还剩 119pt，左右相差 1pt。如果双面打印的话，两面的正文部分恰好是重叠
5. 页边距，1in
6. \topmargin，上边距，18pt
7. \headheight，页眉高度，12pt
8. \headsep，页眉与正文间距，25pt
9. \textheight，正文高度 595pt，可以放下 38 行文字
10. \footskip，正文与页脚基线间距，30pt。比页眉的 12pt+25pt 小了 7pt



11. $\backslash 845\text{pt}$ 减去上面的全部尺寸, 还剩 93pt, 比上面的 $1\text{in}+18\text{pt}$ 多 3pt。

当字号发生变化时, 上面的某些尺寸也发生一定的变化, 当把 `oneside` 改为 `twoside`, 奇偶页的左边距就变成 22pt 和 70pt。但是奇数页右边空白恰好和偶数页左边空白相等, 不会给双面打印造成困扰。一般情况无需改动 L^AT_EX 的页面布局缺省设置, 有特殊需要时可用 `\setlength`, `\addtolength` 设置宏变量的值。geometry 宏包提供更高级的用户接口, 如下面设置页面尺寸和边距:

```
1 \usepackage[paperwidth=100mm,paperheight=150mm,margin=20mm]{geometry}
```

```
1 %也可以单独设置每个边距:
2 \usepackage[top=2in,bottom=1in,left=1in,right=1in]{geometry}
3 %把页面横过来:
4 \usepackage[landscape]{geometry}
```

10.2 页面样式

页面样式也就是页眉和页脚的内容, 有 4 个样式:

<code>empty</code>	页眉, 页脚空白
<code>plain</code>	页眉空白, 页脚含居中页码, 非 book 类文档的缺省值
<code>heading</code>	页脚空白, 页眉含章节名和页码, book 类文档缺省值
<code>myheading</code>	页脚空白, 页眉含页码和用户自定义信息

同时可以用 `\pagestyle` 和 `\thispagestyle` 命令设置整个文档或某页的样式。

```
1 \pagestyle{plain} %全局设置
2 \thispagestyle{empty} %单页设置
3
4 %自定义样式
5 %使用了@特殊符号, 第一行用\makeatletter命令声明
6 %最后一行用相应的\makeatother命令恢复现场
7 \makeatletter
8 \newcommand{\ps@permanentdamagedhead}{
9   %自定义奇偶页的页眉和页脚
10   \newcommand{\@oddhead}{奇页眉\hfill 右半边}
11   %\hfill为弹性填充命令
12   \renewcommand{\@oddfoot}{\hfill\thepage\hfill}
13   \newcommand{\@evenhead}{偶页眉\hfill 奇半边}
14   \newcommand{\@evenfoot}{\@oddfoot}
15 }
16 \makeatother
```

自定义页面样式时, 还可以使用一些宏变量来显示页码, 章节号码和名称等, 具体如下表所示: 前 5 个变量可直接定义, 左右标记需要用以下命令间接定义:

```
1 \makeboth{左标记}{右标记} %定义2个标记
2 \markboth{右标记} %定义右标记
```

表 3: 页眉页脚常用宏变量

<code>\thepage</code>	页码
<code>\thechapter</code>	章编号
<code>\thesection</code>	节编号
<code>\chaptername</code>	章起始单词名,Chapter
<code>\sectionname</code>	节起始单词名,Section
<code>\leftmark</code>	左标记, article 文档类中包含 section 信息, report 和 book 包含 chapter 信息
<code>\rightmark</code>	右标记, 在 article 包含 subsection 信息, 在 report 和 book 中包含 section 信息

```

3  %引用时用\leftmark和\rightmark, 定义时用\markboth和\markright
4
5  %book类文档中myheadings样式定义
6  \def\pa@headings{
7      \let \@oddfoot\@empty \let \@evenfoot\@empty
8      %清空页脚
9      \def\@evenhead{\thepage\hfill\slshape\leftmark}
10     %定义偶数页页眉(左页页眉), 页码居左, 左标记居右
11     \def\@oddhead{{\slshape\rightmark}\hfill\thepage}
12     %奇数页页眉(右页页眉), 右标记居左, 页码居右。
13 }

```

fancyhdr 宏包提供更灵活的控制和高级语法, 如下所示:

```

1  %怎样为myheadings样式定制左右标记
2  \documentclass{book}
3  \markboth{左}{右}
4  \pagestyle{myheadings}
5  ...
6  \begin{document}
7      左页
8      \newpage
9      右页
10     \newpage
11 \end{document}

```

10.3 分栏

```

1  \documentclass[twocolumn]{article}

```

multicol 宏包提供多达 10 个栏位, 数目可任意切换, 各栏的长度相同。注意在 multicol 环境中对浮动体的支持有限, 只能使用带 * 的版本。且浮动体跨栏位, h 选项也会失效, 最早也只出现在下一页的页首, 因此建议不让浮动体出现在栏位内。

```

1  \usepackage{multicol}

```

```

2 \setlength{\columnsep}{12pt}
3 %将栏位之间的距离设为12pt(缺省为10pt)
4 \setlength{\columnseprule}{1pt}
5 %将栏位间的分割线宽度设为1pt(缺省不显示)
6 \begin{multicols}{2}
7 ...
8 \end{multicols}

```

10.4 分页

\TeX 一般自动分页，针对浮动体较多的情况，自动分页效果可能不是我们想要的，可以手工插入分页命令。

```

1 \newpage
2 \pagebreak[3]
3 %参考数值1-4,4表示强烈分页
4 \nopagebreak[2]
5 %强烈程度1-4, 4表示强烈反对
6 \clearpage
7 %浮动体较多，此命令要求 $\text{\TeX}$ 排完此前所有的浮动体

```

```

1 print('hi')
2 for each min

```

11 符号

11.1 \TeX 标志符号

\TeX , \LaTeX , \LaTeX 2 ϵ 
 \XeLaTeX
 \AmS-TeX , \AmS- \LaTeX 
 \BibTeX , \LuaTeX
 \METAFONT , \METAPOST

```

1 %基本符号标志
2 \TeX{}, \LaTeX{}, \LaTeXe{}
3 %需要导入xltextra宏包
4 \XeLaTeX{}
5 %texnames宏包提供
6 \BibTeX{}, \LuaTeX{}
7 \AmSTeX{}, \AmS- $\text{\LaTeX}$ {}
8 %mflogo宏包提供
9 \METAFONT{}, \MF{}, \MP{}

```

12 简历的制作