

Test

Fan

2024 年 10 月 3 日

目录

1	Elementary	2
2	error handling	3
3	Environment	3
4	graph	6
5	float	6
6	math	6
6.1	theorems, lemmas, ...	7
7	引用	7
7.1	文本内相互引用	7
8	index 索引	8
9	自定义	8
9.1	newcommand	8
9.2	new environment	9
10	字体	9
11	间距	9
11.1	行间距	9
11.2	段间距	11

11.3 其他间距	11
---------------------	----

12 页面设置	11
---------	----

摘要

The abstract 阿巴斯塔塔

1 Elementary

每一行开头的空格会被忽略正常位置的一个空格当然会被视为一个空格换行会被视为一个空格多个连续的空格也会被视为一个空格(这些对于英文有效,写中文的时候都是不产生空格的)

两行文字中间的空白行会被视为另起一段的标志,多行空白页等同于一行空白

`\\`也可以用于换行,也可以使用

`\ newline`来另起一段

`*`只是换行,不另起一段

命令后的空格会被忽略,可以通过在命令后添加`{}`再添加空格的方式来
实现输出

`\quad`和`\qquad`用于生成不同长度的空格

``会产生相当于`text`中文本长度的空格

`\linebreak[n]`,`nolinebreak`,`pagebreak`,`nopagebreak`后面都可以跟`[n]`,其中`n`为0-4,表示建议的程度

`label{marker}`会将该处进行标识,在文章其他位置处使用`ref`或`pageref`可以生产跳转至该处或该页的链接

`footnote`可以生成脚注

引用的单引号和双引号不是直接键盘上的符号,左边应该是`'`,右边应该是`'`:

`dash`用连续三个的减号-

波浪号是`~`

摄氏度的小圆圈`o`

省略号不是直接打三个英文句号,而是用`\ldots`

带星号的各种命令与不带星号的命令的区别在于,在其他计数或编号相关的代码执行时,前者会被忽略.

`\newcommand{cmd}{def}`可以用于自定义命令,`cmd`中是自定义的命令,`def`中是对应的实际命令

`\rule[lift]{width}{thickness}`可以产生一条竖直方向的黑色粗条纹,其中`lift`参数是指下边的起始位置.

`#$\&_{}\backslash`都需要通过转义符`\`来实现

2 error handling

`\sloppy`可以用来消除`overfull hbox`的错误,该错误是指一行的长度超出了预定长度,该命令会强制减少该行内的字数,用`fussy`可以抵消效果

3 Environment

1. 环境可以嵌套:

- 但别嵌套太多
- 看上去有点傻

2. 因此:

Good 可以用于事物的定义

Bad 不用于事物的定义

一般而言一行中不应该有超过66个英文字母,这也是为什么默认的边距那么大

这个环境用于原样输出,也可以直接使用`\verb+text+`的形式来原样输出`text`,其中`+`用作分隔符,可以用出了`*`或者空格以外的符号来代替,不能用字母来代替

下面将介绍`\begin{tabular}[pos]{table spec}`命令

`table spec`中,可以填入`lrc`和`p{}`分别代表左对齐,右对齐,中间对齐和带有换行的文字 (正常情况下,在制表环境下,latex不会自动换行),除此之外用`—`代表竖线

`pos`中可以填入`tb`和`c`,用于指定表格在页面中的位置,分别代表顶部,底部和中间

在制表环境中,`&`可以跳至下一列,`\\`可以用于新起一行 `\hline`用于插入水平线,`\cline{i-j}`用于指定添加由`i`列到`j`列的横线

—可以用@{}来替换,括号中间的符号会用于替换横线,如果是空白的话会把横线处留出的空格消除如果没有在table spec中添加左右两侧的横线的话会默认预留出空格的位置

`\multicolumn{2}{c}{text}`用于将多列合并,用于这些列的首行位置,第一个括号内为合并的行数,第二个括号内为合并后对齐的方式,第三个括号内为合并后首行的内容

Pi expression	Value
π	3.1416
π^π	36.46
$(\pi^\pi)^\pi$	80662.7

在multline环境中,可以对数学公式进行多行排版,与equation环境的区别在于,在该环境中换行是可以人为控制,任意插入的,通过\\来实现

在align环境中,可以通过&来进行对齐,每一行默认进行编号,若某行无需编号,可在末尾添加nonumber

IEEEeqnarray环境用于数学公式多行排版

```
\begin{IEEEeqnarray}{rCl}
a&=&b+c
\\
&=&d+e+f+g+h
+i+j+k\nonumber \\
&&\negmedspace {} +l +m+ n+ o
\\
&= &p+q+r+s
\end{IEEEeqnarray}
```

$$a = b + c \tag{1}$$

$$= d + e + f + g + h + i + j + k + l + m + n + o \tag{2}$$

$$= p + q + r + s \tag{3}$$

上面是一个用该环境解决连续多个加号并且需要换行的情况,其中rCL表示3列,对齐方法分别是右对齐,居中和左对齐其中C被大写了,意味着会在该列两侧额外添加一些空格.rclRCL都是用于数学公式,而stu则是用于文字,分


```

-x & \text{if } x < 0, \\
0 & \text{if } x = 0, \\
x & \text{if } x > 0.
\end{cases}
\end{equation*}

```

matrix系列环境则是专门用于排版矩阵的环境, 分别有matrix, pmatrix, bmatrix, Bmatrix, vmatrix. 其中matrix环境不自动生产括号, 而接下来几个环境依次自动生成 \langle , $[$, $\{$, $|$ 和 $||$. 在这些环境中, 排版规则和tabular环境一致

proof环境会帮忙自动生成证明字样以及证明结束的符号, 可以使用qedhere命令来指定证明结束符号所在行, 否则它会单独占据一行.

4 graph

```
\includegraphics[angle=90,width=\textwidth,scale=1]{test.png}
```

用于插入图片, 其中 width 使得图片宽度和文本内容的宽度一致, 当然也可以设置为其他的宽度, angle为顺时针旋转角度 scale可以用来缩放图片

5 float

latex为表格和图片分别提供了两种浮动体环境, 分别是

```
\begin{figure}[placement specifier]和\begin{table}[P S]
```

ps中可以填入htbp!, 分别代表当前位置, 某页的顶部, 底部或者为浮动体单独开辟一页, 以及强制执行这些命令. 默认为tbp

$\caption[caption used in the list]{caption text}$ 可以为浮动体添加标题 \listoffigures 和 \listoftables 可以打印图片和表格的标题列表, 列表中的标题为

中的内容想对浮动体进行引用的话, 要在对于浮动体的 $\caption{\}$ 命令的 $\{\}$ 内的尾部添加 \label

6 math

$\begin{equation}$ 环境用于数学公式, 该环境中会将公式根据章节来进行编码, 用 $\tag{内容}$ 可以取消该公式的编码, 并用 $\{\}$ 内的文字代替编码

如果不想将公式进行编码,则使用`\begin{equation*}`环境,或者在`\[\]`中输入公式

在数学公式中输入文字的话要嵌套使用`\text{}`

可以在前言区使用`\DeclareMathOperator{指令}{对应代码}`来自定义一些数学符号

想要成对的分隔符,如括号,的大小根据其中的内容进行更改,需要在前后分隔符的前面分别添加`\left`和`\right`, 如果不想要其中一个分隔符,但是又想要让单独的分隔符调整大小,则在不要的分隔符用`.`来代替

数学公式模式中,使用`\,`或`\:`或`\;`或`\`加上空格或`\quad`或`\qquad`或`\!`来产生由短到长的空格,最后一个除外. 最后一个专门用于负号和数字之间的空格调整,是用于消去多余的间隙的

可以使用`\boldsymbol{symbol}`来排版粗体符号

6.1 theorems, lemmas, ...

为了在文章中生成定理,公式,注记.需要在前言区中定义相关格式.

```
\theoremstyle{definition/plain/remark}
\newtheorem{name}[counter]{text}[section]
```

其中style需三选一,definition为宽罗马字体,plain为宽意大利斜体,remark为斜罗马字体. 调用自定义的格式的格式为`\begin{name}[text]`和`\end{name}`. 以上内容中方括号内的参数是可选的. name是所自定义的格式的名称,counter在前言区中填入的是和该name属于同一类的name,编码时两者会视为同一个,在使用时填入的是具体的内容,如具体的某部法律的名称.会在text后打印在括号中 text中则是会打印出来的抬头,如“定义”. 在前言区若添加了section参数,会使得该格式的编码会结合该section来呈现,也可以替换为chapter或subsection关键词.

7 引用

7.1 文本内相互引用

如果想在文本内相互引用,首先需要在被引用的地方插入marker,插入方法是使用指令`\bibitem[label]{marker}`, 引用处插入指令`\cite{marker}`. 如果没有使用label关键词,则会自动编号.

使用案例是

```

Partl~\cite{pa} has
proposed that \ldots
\begin{thebibliography}{99}%99指的是引用条目前的编号的长度不会超
过99这个文本的长度
\bibitem{pa} H.~Partl:
\emph{German \TeX},
TUGboat Volume~9, Issue~1 (1988)
\end{thebibliography}

Partl [1] has proposed that ...

```

参考文献

- [1] H. Partl: *German T_EX*, TUGboat Volume 9, Issue 1 (1988)

8 index 索引

索引的指令为`\index{key@formatted_entry}`.其中key是用于索引分类的,而entry是索引的关键词. entry是可选的,如果缺失,key将会被使用.一般entry是呈现在索引表上的关键词,如果没有,则用key代替,key和entry之间加!说明该索引是该key下的子索引. 如果想对索引表中的页码进行排版,则将@entry改为一加上格式.

在想要打印索引表的地方使用命令`\printindex`

9 自定义

9.1 newcommand

可以通过命令`\newcommand{name}[num]{def}`来自定义指令,其中num默认为0,代表的是自定义的命令所需要的参数的数量,如果不为零,则在def中使用#1,#2...来代表各参数的占位符.

可以用命令`\renewcommand`来覆盖之前定义的同名自定义指令,其各参数与命令`\newcommand`一致.

9.2 new environment

可以使用命令`\newenvironment{name}[num]{before}{after}`来自定义环境.`num`的含义与使用方法与自定义命令一致. `before`与`after`参数分别是该自定义环境中的填写的命令之前与之后发生的命令.同样地,也有`\renewenvironment`命令.

10 字体

想打印特定字号的字体,可以调用同名的环境,也可以直接在同一个花括号内,以字体命令开头,然后输入文字.

`\emph{text}`的强调效果会根据上下文的字体状态进行调整

11 间距

11.1 行间距

可以使用命令`\linespread{factor}`来调整行间距.当`factor=1.3`时,为一行半,为1.6时为双倍.默认为1

更正规的命令为`\setlength{\baselineskip}{1.5\baselineskip}`,案例为

```
{\setlength{\baselineskip}%  
{1.5\baselineskip}  
This paragraph is typeset with  
the baseline skip set to 1.5 of  
what it was before. Note the par  
command at the end of the  
paragraph.\par}%只有当\par命令在花括号内部时,才会对于行间距才会  
产生调整,  
当字体改变时,也只有加上\par,行间距才会随字体改变  
This paragraph has a clear  
purpose, it shows that after the  
curly brace has been closed,  
everything is back to normal.
```

表 1: Fonts

<code>\textrm{text}</code>	roman	<code>\textsf{text}</code>	sans serif
<code>\texttt{text}</code>	typewriter		
<code>\textmd{text}</code>	medium	<code>\textbf{text}</code>	bold face
<code>\textup{text}</code>	upright	<code>\textit{text}</code>	<i>italic</i>
<code>\textsl{text}</code>	<i>slanted</i>	<code>\textsc{text}</code>	SMALL CAPS
<code>\emph{text}</code>	<i>emphasized</i>	<code>\textnormal{text}</code>	document font

表 2: Font Sizes

<code>\tiny</code>	tiny font	<code>\Large</code>	larger font
<code>\scriptsize</code>	very small font	<code>\LARGE</code>	very large font
<code>\footnotesize</code>	quite small font	<code>\huge</code>	huge
<code>\small</code>	small font		
<code>\normalsize</code>	normal font	<code>\Huge</code>	largest

表 3: Absolute Point Sizes in Standard Classes

size	10pt(default)	11pt option	12pt option
\tiny	5pt	6pt	6pt
\scriptsize	7pt	8pt	8pt
footnotesize	8pt	9pt	10pt
\small	9pt	10pt	11pt
\normalsize	10pt	11pt	12pt
\large	12pt	12pt	14pt
\Large	14pt	14pt	17pt
\LARGE	17pt	17pt	20pt
\huge	20pt	20pt	25pt
\Huge	25pt	25pt	25pt

表 4: Math Fonts

<code>\mathrm{text}</code>	RomanFont
<code>\mathbf{text}</code>	BoldfaceFont
<code>\mathsf{text}</code>	SansSerifFont
<code>\mathtt{text}</code>	TypewriterFont
<code>\mathit{text}</code>	<i>ItalicFont</i>
<code>\mathcal{text}</code>	$\mathcal{C} \mapsto \mathcal{D} \} \nabla \neg \sqrt{\langle \rangle} \mathcal{F} \setminus \sqcup$
<code>\mathnormal{text}</code>	<i>NormalFont</i>

This paragraph is typeset with the baseline skip set to 1.5 of what it was before. Note the par command at the end of the paragraph.
This paragraph has a clear purpose, it shows that after the curly brace has been closed, everything is back to normal.

11.2 段间距

段间距可以在前言区调整.

```
\setlength{\parindent}{0pt}  
\setlength{\parskip}{1ex plus 0.5ex minus 0.2ex}
```

这段指令将段落缩进设为0.skip设定的是段落之间的空格,plus和minus设定的上下浮动的范围以供调整. 为了避免对目录产生影响,可以将制作目录的命令放在该命令之前.

可以通过使用indent指令来缩进未缩进的段落,需要放在段落开头.类似的,也有noindent指令.

11.3 其他间距

可以分别使用hspace和vspace来调整横向和纵向的间距,常用长度单位有mm,cm,in,pt,em,ex. 特别地,stretch{n}会自动填充空格,直到充满整行或整页,若同时有多个stretch,会按照n参数按比例分配. 带*的hspace和vspace命令在发生换行或换页的情况时仍会保留未输入的空格.

若仍需额外调整行间距,可以使用\\[length].同时也有\bigskip和\smallskip.

12 页面设置

可以在前言区来设置页面的布局.可以使用\setlength{parameter}{length}或者\addtolength{parameter}{length},第一个命令时设定某参数的固定值,而第二个命令是在某参数的固定值的基础上进行增加. 各参数及其含义可以查看Ishort.pdf的115页.

可以使用命令\parbox[position]{width}{text}或者\begin{minipage}[position]{width}t生成一段在框内的文字.两者的区别在于后者更加功能强大,可以在该环境中使用几乎所有命令.