

# 美国大学生数学建模竞赛 $\text{\LaTeX}$ 模板使用指南

种田 er

种田大学

<https://www.latexstudio.net>

January 14, 2021

$\text{\LaTeX}$ 、 $\text{\TeX}$  系统都是复杂的软件包，里面包含各种各样的排版引擎、编译脚本、格式转换工具、配置文件、支持工具、字体以及文档。一个  $\text{\TeX}$  发行版就是把所有这样的部件都集合起来，打包发布的软件。

- $\text{\TeX}$ Live，跨平台，有比较齐全的宏包。
- Mik  $\text{\TeX}$  临时自动下载没有所需要的宏包，体积较小。
- Mac  $\text{\TeX}$  专门为 Mac 定制的发行版
- ~~$\text{\CTeX}$~~ 是一款过时的发行版，早在 2012 年就停止维护，对于  $\text{\TeX}$  来说，两年一更是必要的，**所以请放弃使用  $\text{\CTeX}$ 。**

关于下载，可以参考王然老师写的 `Install-LaTeX.pdf`，在官网或者相应的镜像网站中下载，也可以参考 LaTeX 工作室写的文章。

<https://www.latexstudio.net/archives/51801.html>，发行版下载完成后，请在命令行中使用 `tex -v` 检查是否安/装成功。

- 命令行，用命令行进行编译能够让人能够更加理解编译的链条。
- WinEdt，收费软件，推荐购买后使用，对初学者界面友好。
- TeXworks -常见的 TeX 套装都自带这款编辑器，界面比较清爽，支持代码和 pdf 查看，左右分屏显示。
- TeXStudio 开源免费的编辑器，界面集成度好。
- Vscode 轻量级的编辑器，可以在内写多种语言，但是配置稍许麻烦。
- Sublime 与前者一样，也是一款轻量级的编辑器，可以在内书写多种语言，但是同样配置稍许麻烦。
- overleaf 在线编辑  $\text{\LaTeX}$  的网站，投稿人常用的网站，能够省去配置的烦恼，但是由于内地使用不稳定，需要按照自己的情况选取。

如果发行版与编辑器是飞机和引擎的话，那么发行版就是引擎，它让飞机飞起来，而飞机只是一个外框，它并不起决定作用。

# 遇事不决先问文档再问百度

## 一份（不太）简短的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 介绍

或 110 分钟了解  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$

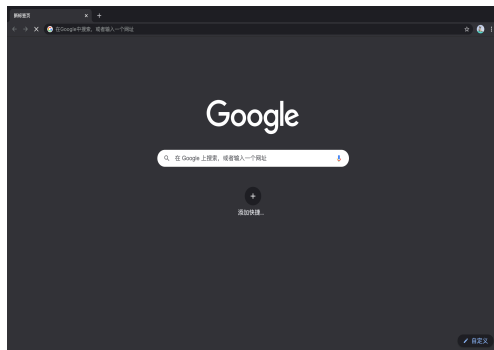
英文作者: Tobias Ottl

Hubert Partl, Ines Hyna and Elisabeth Schlegl

英文版本: Version 6.2, February 28, 2018

中文翻译: CT<sub>EX</sub> 开发小组

中文版本: 版本 6.01, 二零一九年四月



遇到事情，先找找相关宏包的手册，再去问百度能否解决，能用 Google 解决尽量用 Google 解决。对于刚入门的  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  的人来说，《112 分钟了解  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 》已经能够解决大部分问题。

# 提问是一门艺术

- 提问之前最好先尝试自己解决问题。
- 如果无法解决，请提供 MWE(Minimal Working Example) 以及报错信息、日志等材料，并提供你想实现的效果。
- 学会自己去阅读报错信息，并合理利用大段注释法解决问题。
- 找到合适的渠道提问，例如师兄师姐、QQ 群、小屋中的问答模块，`texstackexchange` 中的等等。

只要态度认真，提问内容清晰完整，再加上适当的礼貌，在互联网上得到帮助都不难。

# 最基本的层次与最高的奥义

**内容与格式分离**， $\text{\LaTeX}$  模板已经将基本的设置帮你设置完成，请一定不要自己再去修改某些设定，建模比赛就好比一次投稿，如果你不想麻烦评委老师，也不想因为自己的强迫症影响了格式而扣分，就请一定不要自行修改模板的内容。修改模板是非常困难的一件事情，对于初学者来说，无法修改出想要的效果不说，再咨询他人想实现效果的时候可能也没法提问。



**$\text{\LaTeX}$  不是 Word、WPS**，所见即所得不是它的代名词，接受它，要不放弃它。

# 资源

- QQ 1 群: 91940767
- QQ 3 群: 640633524
- LaTeX 工作室: <https://www.latexstudio.net>.
- ctan:<https://www.ctan.org>
- texstackexchange:<https://tex.stackexchange.com>
- LaTeX 工作室公众号



合理利用资源，永远保持一颗学习的心。

美国大学生数学建模  $\text{\LaTeX}$  模板可以在 Github 上下载，各高校老师有时也会在自己学校的比赛群内发，或者是赛前的几天在 LaTeX 小屋中会发布最新的模板，请自行关注。

如果想从互联网中获取资源可以从以下 2 中途径中选择：

- Github: <https://github.com/latexstudio-org/mcmthesis>
- LaTeX 小屋:  
<https://www.latexstudio.net/index/details/index/mid/63>

从 Github 下载完成后，在命令行中运行指令 `xetex mcmthesis.dtx` 以获取 demo，也可以进入 release 中直接下载 demo 编辑后直接生成 pdf。



# 一些基本设置

```
1 \mcmsetup{CTeX = false, % 使用 CTeX 套装时, 设置为 true
2 tcn = number, problem = C,
3 sheet = true, titleinsheet = true,
4 keywordsinsheet = true,
5 titlepage = false, abstract = true}
```

由于在 demo 中有两个摘要页, 需要将其中的 `titlepage = true` 改为 `titlepage = false` 这里仍然是建议不要使用 **CTeX 套装** 以免后续出现某些致命的问题。

- 命令都以反斜线 `\` 开头，后接命令名，命令名或者是一串字母，或是单个符号。命令可以带一些参数，通常用花括号括起来。如果命令参数只有一个字符（不包括空格），花括号可以省略不写。可选参数 (option) 如果出现，则用方括号括起来。例如 `\documentclass` 就是一个能带可选参数的命令。
- 环境为 `\begin{env}` 和 `\end{env}` 的形式出现，例如模板中的摘要环境：

```
1 \begin{abstract}  
2 摘要内容  
3 \end{abstract}
```

# 使用宏包

使用 `\usepackage[option]{package}` 加载相应的宏包。宏包可以以相对简单的接口实现复杂的功能，就好比其他程序语言的“库”，如果需要查询所需要的功能可以使用

```
1 \texdoc package
```

来查询你所需要的功能的实现方法。但是需要注意的是，有些宏包可能会出现冲突，你需要找到相应的解决办法。

建模模板中的已经加载的宏包基本已经能够满足你写建模文章的要求，若非必要，尽量不要再加载多余的宏包，以免冲突

## 常用的命令

- 换行: `\\[space]`、`\\linebreak`
- 分页: `\\newpage`、`\\pagebreak`
- 空格: `\\phantom{arg}`、`\\quad`、`\\qquad`
- 间距: `\\vspace{space}`、`\\hspace{space}`
- 标点符号: 引号需要 `tab` 键上面 1 键左边的按键来实现 `` ` ' 来完成而并非直接中文 `shift + "` 来实现。

无法直接输入的标点符号:

- 1 \# \quad \\$ \quad \% \quad  
|& \quad
- 2 \{ \quad \} \quad \\_ \quad
- 3 \textbackslash

# \$ % & { } \_ \

层次	名称	命令	说明
-1	part (部分)	<code>\part</code>	可选的最高层次
0	chapter (章)	<code>\chapter</code>	本模板无该曾
1	section (节)	<code>\section</code>	article 或 ctexart 类 最高层
2	subsection (小节)	<code>\subsection</code>	
3	subsubsection (小小节)	<code>\subsubsection</code>	report,book 类或 ctexrep,ctexbook 类
4	paragraph (段)	<code>\paragraph</code>	本模板基本不会用
5	subparagraph	<code>\subparagraph</code>	本模板基本不会用

# 列表环境

列表环境有有序列表和无顺列表，在建模比赛中的假设中常常会用到列表环境。

## • 有序列表

```
1 \begin{enumerate}
2 \item 条目1
3 \begin{enumerate}
4 \item 条目1.1
5 \item 条目1.2
6 \item 条目1.3
7 \end{enumerate}
8 \item 条目2
9 \item 条目3
10 \end{enumerate}
```

- ① 条目 1
  - ① 条目 1.1
  - ② 条目 1.2
  - ③ 条目 1.3
- ② 条目 2
- ③ 条目 3

- 无序列表

```
1 \begin{itemize}
2 \item 条目1
3 \begin{itemize}
4 \item 条目1.1
5 \item 条目1.2
6 \item 条目1.3
7 \end{itemize}
8 \item 条目2
9 \item 条目3
10 \end{itemize}
```

- 条目 1
  - 条目 1.1
  - 条目 1.2
  - 条目 1.3
- 条目 2
- 条目 3

# 字体

字体族	带参数命令	声明命令	效果
罗马	<code>\textrm{&lt;文字&gt;}</code>	<code>\rmfamily</code>	Roman font
无衬线	<code>\textsf{&lt;文字&gt;}</code>	<code>\sffamily</code>	Sans serif font
打字机	<code>\texttt{&lt;文字&gt;}</code>	<code>\ttfamily</code>	Typewriter font

字体形状	带参数命令	声明命令	效果
直立	<code>\textup{&lt;文字&gt;}</code>	<code>\upshape</code>	Upright shape
意大利	<code>\textit{&lt;文字&gt;}</code>	<code>\itshape</code>	<i>Italic shape</i>
倾斜	<code>\textsl{&lt;文字&gt;}</code>	<code>\slshape</code>	<i>Slanted shape</i>
小型大写	<code>\textsc{&lt;文字&gt;}</code>	<code>\scshape</code>	SMALL CAPITALS

预定义的字体系列有中等 (medium) 和加宽加粗 (bold extended) 两类:  
如 `\textmd{中等}` 对应 中等 以及 `\textbf{加宽加粗}` 对应 **加宽加粗**



# 字号及特殊符号

常用的字号有 `\tiny`、`\scriptsize`、`\footnotesize`、`\small`、`\normalsize`、`\large`、`\Large`、`\LARGE`、`\huge`、`\Huge`。  
也可以使用 `\fontsize{大小}{行距}` 和 `\selectfont` 的组合来定义字体的大小。

## 第四章 排版数学公式

准备好了！本章将介绍到  $\text{\LaTeX}$  著名的短语——排版数学公式。当然得注意了，本章的内容只是一点皮毛，虽然对大多数人来说已经够用了，但是如果不解决决定性的问题的话也不要大惊小怪。求助于搜索引擎或者有经验的人不失为一个好办法。

### 4.1 $\text{\LaTeX}$ 宏集

在介绍数学公式排版之前，简单介绍一下  $\text{\LaTeX}$  宏集。 $\text{\LaTeX}$  宏集是美国数学会 (American Mathematical Society) 提供的对  $\text{\TeX}$  原生的数学公式排版的扩展，其核心是 `amstex` 宏包，对多行公式的神速提供了有力的支持。此外，`amstex` 宏包以及基于它的 `amsmath` 宏包提供了丰富的数学符号；`amstex` 宏包扩展了  $\text{\TeX}$  宏集定理证明格式。

本章介绍的许多命令和环境依赖于 `amstex` 宏包。这些命令和环境将以蓝色示意。

### 4.2 公式排版基础

#### 4.2.1 行内和行内公式

数学公式有两种排版方式：其一是与文字混排，称为行内公式；其二是单独列为一行排版，称为行内公式。

行内公式由一对 `$` 符号包裹：

The Pythagorean theorem is  
`$a^2 + b^2 = c^2$`.

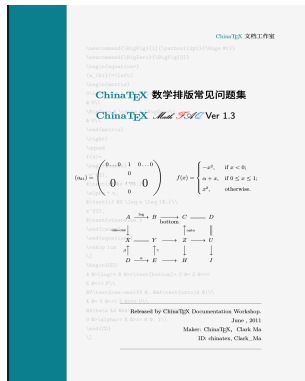
The Pythagorean theorem is  $a^2 + b^2 = c^2$ .

单独成行的行内公式在  $\text{\LaTeX}$  中由 `equation` 环境包裹，`equation` 环境为公式自动生成一个编号，这个编号可以用 `\label` 和 `\ref` 生成交叉引用。`amstex` 的 `\eqref` 命令用于引用自动加上方括号；还可以用 `\tag` 命令手动修改公式的编号，或者用 `\tagged` 命令取消为公式编号（与基本的等价的命令是 `\untagged`）。

The Pythagorean theorem is:  
`\begin{equation}`  
`a^2 + b^2 = c^2 \tag{pythagoras}`  
`\end{equation}`  
`equation \ref{pythagoras}` is  
called "Gougu theorem" in Chinese.

The Pythagorean theorem is:  
$$a^2 + b^2 = c^2 \quad (4.1)$$
  
Equation (4.1) is called "Gougu theorem" in Chinese.

对于数学公式，并不建议直接使用外部工具来获得。可以先读 `lshort-zh` 中的数学公式公式部分的章节，如果仍有问题可以阅读王昭礼老师写的数学公式排版常见问题集。有机会可以找几篇文章的数学公式敲一敲很快就能熟悉。对于特殊符号可以使用 `texdoc symbols` 寻找，如果仍有问题，可以去在相关的提问区提问。



数学模式有行内模式和行间模式，分别用 `$ $` 和 `\[ \]` 来实现。  
例如 `$ a+b = c $` 会显示  $a + b = c$ . 而 `\[ a+b = c\]` 会显示

$$a + b = c$$

## 上标与下标:

- 上标通过 `arg1^{\arg2}` 来实现，如果 `arg2` 中仅有一个字母或数字仅可以省去括号。
- 下标通过 `arg1_{\arg2}` 来实现，同样，如果 `arg2` 仅有一个字母或者数字可以省去括号。

# 分式、根式、矩阵

## 分式、根式、矩阵

- 分式: 通过 `\frac{分子}{分母}` 实现, 例如 `\frac{1}{2}` 的效果为  $\frac{1}{2}$ , 在正文中可以使用 `\dfrac{1}{2}` 展示正文分式。例如 `\dfrac{1}{2}` 显示为  $\frac{1}{2}$ 。
- 根式: 通过 `\sqrt[次数]{数字}` 来实现, 例如 `\sqrt[3]{8}` 的实现效果为  $\sqrt[3]{8}$ 。
- 矩阵:

matrix 环境

$$\begin{matrix} a & b \\ c & d \end{matrix}$$

bmatrix 环境

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

vmatrix 环境

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$$

pmatrix 环境

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Bmatrix 环境

$$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$$

Vmatrix 环境

$$\left\| \begin{matrix} a & b \\ c & d \end{matrix} \right\|$$

更多简单的数学公式请参考文档。

# 数学字体

数学公式中，只有默认的意大利体，数学常数  $e$  常用罗马体的  $\mathrm{e}$ 。

下面是标准  $\mathrm{\LaTeX}$  默认提供的数学字母字体。

类别	字体命令	输出效果
默认字体	<code>\mathnormal</code>	<i>ABCDE</i>
意大利体	<code>\mathitl</code>	<i>ABCDE</i>
罗马体	<code>\mathrml</code>	ABCDE
粗体	<code>\mathbf</code>	<b>ABCDE</b>
无衬线体	<code>\mathsf</code>	ABCDE
打字机体	<code>\mathtt</code>	ABCDE
手写体（花体）	<code>\mathcal</code>	<i>ABCDE</i>

# 定理类环境

为方便写作，模板也定义了相应的定理类环境。

中文名称	环境名称	命令
定理	Theorem	<code>\begin{Theorem} ... \end{Theorem}</code>
引理	Lemma	<code>\begin{Lemma} ... \end{Lemma}</code>
推论	Corollary	<code>\begin{Corollary} ... \end{Corollary}</code>
命题	Proposition	<code>\begin{Proposition} ... \end{Proposition}</code>
定义	Definition	<code>\begin{Definition} ... \end{Definition}</code>
例子	Example	<code>\begin{Example} ... \end{Example}</code>

```
1 \begin{Example}^^I
2 This is an example
3 \label{key}
4 \end{Example}
```

# 稍许复杂数学公式

- 多行公式可以使用 `gather`、`gather`、`flalign`、`alignat` 实现，对于 `align` 内部的编号可以使用 `\notag` 取消编号。
- 拆分单个公式，拆分公式可以使用 `split`、`multiline` 实现。
- 将公式组合为块，可以使用 `cases` 环境、`multilined`、`aligned` 等环境实现。

**更多复杂的数学公式可以参考王昭礼老师写的文档。**

如果对于数学公式真的不熟悉或者是在比赛前几天才开始学习使用模板的人来说，可以使用数学工具，但是总体上仍不建议使用数学工具，容易出现公式转换称  $\text{\LaTeX}$  代码之后出现问题你无法解决的问题。

如果仍要使用数学工具，可以使用以下的数学工具。

工具	介绍
detexify	手写符号识别，对于个别符号不会输入较为有效
mathphix	需要付费购买，相对较为便宜
Axmath	也需要付费购买，相对来说也比较便宜
mathtype	较贵，不容易承受价格
hostmath	在线识别工具，且免费
各个编辑器中的数学工具栏	需要自我探索



# 表格的基础使用

表格是  $\text{\LaTeX}$  中的基础知识，同时也是难点之一，对于基础表格排版比较简单而对于过宽过长的表格则需要的一定的基本功才能解决！

tabular 的一般格式为：

```
1 \begin{tabular}[<垂直对齐>]{<列格式说明>}  
2 <内容> & <内容> & 内容 \\  
3 ...  
4 \end{tabular}
```

列对齐格式有：

- l 左对齐
- c 居中
- r 右对齐
- p{宽} 固定宽度
- | 画一条竖线
- @{<内容>} 添加任意内容，但是会取消表列的距离
- \*{<计数>}{<列格式说明>}，符号重复多次

# 表格的基础使用

```
1 \begin{tabular}{|c|c|c|}  
2 \hline  
3 第一列 & 第二列 & 第三列 \\  
4 \hline  
5 第一列 & 第二列 & 第三列 \\  
6 \hline  
7 \end{tabular}
```

第一列	第二列	第三列
第一列	第二列	第三列

## 三线表的使用

```
1 \begin{tabular}{ccc}  
2 \toprule  
3 名称 & 作用 & 效果 \\  
4 \bottomrule  
5 a & b & c \\  
6 \bottomrule[0.1em]  
7 \end{tabular}
```

名称	作用	效果
a	b	c

# 表格的基础使用

三线表中用 `\toprule[width]`、`\midrule{width}`、`\bottomrule{width}` 表示顶部的 rule 中部的 rule 以及底部的 rule。  
`width` 为可选参数。表格的制作方法与普通表格相同。

普通的表格内部无法自动换行，如果想换行，可以使用 `\makecell{...}` 命令在需要的位置上加上 `\\`

对于想要对表格进行斜线切割，可以使用 `diagbox` 宏包中的 `\diagbox{...}{...}..` 命令进行切割。例如

```
1 \begin{tabular}{|c|c|c|}  
2 \hline  
3 \diagbox{a}{b} & c & d \\4 \hline  
5 d & e & f \\6 \hline  
7 \end{tabular}
```

a \diagup b	c	d
d	e	f

# 表格的合并（列合并）

## 列合并：

使用 `\multicolumn{列数}{新列格式}{内容}` 可用于将一行中几个不同的表格合并为一项。但是需要注意的是它会导致原来表格的列格式失效，比如原先加入竖线则需要在命令的新列格式中加入新的竖线。

```
1 \begin{tabular}{|r|r|}  
2 \hline  
3 \multicolumn{2}{|c|}{成绩} \\ \\  
4 \hline  
5 语文 & 数学 \\ \\  
6 \hline  
7 87 & 100 \\ \\  
8 \hline  
9 \end{tabular}
```

成绩	
语文	数学
87	100

# 表格的合并（列合并）

如果不在 `multicolumn` 的列选项中加入竖线则会导致这种情况的出现

```
1 \begin{tabular}{|r|r|}  
2 \hline  
3 \multicolumn{2}{c}{成绩} \\ \\  
4 \hline  
5 语文 & 数学 \\ \\  
6 \hline  
7 87 & 100 \\ \\  
8 \hline  
9 \end{tabular}
```

成绩	
语文	数学
87	100

表格中，列与列之间最小距离一半是由变量 `\tabcolsep` 和 `\arraycolsep` 控制的，在处理过长表格时可以考虑在某个环境中对这两个变量进行单独设置以使作用于专门的表格。

# 表格的合并（行合并）

行合并通常需要 `\multirow` 和 `\cline{i-j}` 配合使用，它有两种基本的用法

- `\multirow{<行数>}{<宽度>}{<内容>}`
- `\multirow{<行数>}*{<内容>}`

前者产生的表项的宽度就是输入内容的宽度，后者达到宽度后会自动换行，而表格行之间的间距由 `\arraystretch` 来控制，可以重新定义其比例系数。

```
1 \begin{tabular}{|c|r|r|}  
2 \hline  
3 \multirow{2}*{姓名} &  
4 \multicolumn{2}{c|}{成绩} \\  
5 \cline{2-3}  
6 & 语文 & 数学 \\  
7 \hline  
8 张三 & 87 & 100 \\  
9 \hline  
10 \end{tabular}
```

姓名	成绩	
	语文	数学
张三	87	100

# 长表格与表格工具的使用

对于长表格，需要使用 `\longtable` 宏包，定制起来较为麻烦，更建议使用 `csv` 相关宏包进行定制。下面是常用的表格处理工具与宏包

工具	介绍
Excel2LaTeX	一款 Excel 的宏，下载方便，使用起来也比较简单
GenerateTable	可以在线生成 $\text{\LaTeX}$ 表格的网站，使用起来也比较方便
pgfplotstable	可以读入数据和内嵌数据排版
csvsimple	通过 <code>csv</code> 文件保存数据后读入制作表格
datatool	通过 <code>csv</code> 文件保存数据后读入制作表格

# 制作表格总结

- 在制作表格时，最好将表格放入浮动题环境内即 `table` 环境内，让表格自由的浮动以适应排版
- `table` 环境内插入 `\caption` 和 `\label`，`label` 必须放在 `caption` 之后
- 三线表尽量不要再加多余竖线
- 制作表格需要靠时间和经验的积累，一朝一夕无法学成



关于插图，也有许多的奥秘，如果不会插图建议阅读下面这篇文档。



这篇文档应该能够解决你在建模时候的所有建模问题，合理利用即可。

```
1 \begin{figure}[htbp]
2 \centering
3 \includegraphics[选项]{文件名}
4 \caption{...}
5 \label{...}
6 \end{figure}
```

- 图片有矢量图和位图，矢量图建议以 \*.pdf 格式插入，位图常用的有 \*.png、\*.jpg、\*.jpeg 等。并不建议插入 eps 图形。
- 在选项中常用的主要有 height、width、scale、angle 几个选项。主要控制插入图片的高度、宽度、缩放尺寸以及角度。
- figure 与 table 是一种浮动体环境，L<sup>A</sup>T<sub>E</sub>X 主要先排版文字再排版浮动题内的内容，figure 与 table 以及 algorithm 环境都都有可选参数 h、t、b、p，当四个组合时排版效果最好。如果想限定在制定位置需要使用 [!htpb]，若排版允许，则会固定位置，若页面排版不允许，则浮动体内容再次浮动

# 并排子图共享一个标题

```
1 \begin{figure}[htbp]
2 \centering
3 \includegraphics[width=1in]{filename}
4 \hspace{1in}
5 \includegraphics[width=1in]{filename}
6 \caption{Two Graphics in One Figure}
7 \end{figure}
```

常见的并排图形有 3 种情况，此为其中一种，即两张图共享一个 caption，`\centering` 使得图形居中，而内部的 `\hspace{length}` 设置了两张图之间的水平间距，也可以使用 `\hfil` 代替，使得两张图推向两边

# 并排子图共享一个标题

```
1 \begin{figure}
2 centering \begin{minipage}[c]{0.5\textwidth}
3 \centering
4 \includegraphics[width=1in]{1.jpg}
5 \end{minipage}%
6 \begin{minipage}[c]{0.5\textwidth}
7 \centering
8 \includegraphics[width=2in]{2.jpeg}
9 \end{minipage}
10 \caption{Centers Aligned Vertically}
11 \end{figure}
```

其中 `minipage` 表示小页环境，它使得整行被分成几部分，而 `\textwidth` 表示整个页面版芯的宽度。需要注意的是在这个代码种 % 无法删去，否则第二幅图会换行。

# 并列的子图

并列的子图可以使用 subfigure 宏包中的 subfigure 命令实现

```
1 \begin{figure}
2 \subfigure[Small Box with a Long Caption]
3 {
4 \label{fig:mini:subfig:a} %% label for first subfigure
5 \begin{minipage}[b]{0.5\textwidth}
6 \centering
7 \includegraphics[width=1in]{filename}
8 \end{minipage}
9 }% ~~~~~/
10 \subfigure[Big Box]
11 {
12 \label{fig:mini:subfig:b} %% label for second subfigure
13 \begin{minipage}[b]{0.5\textwidth}
14 \centering
15 \includegraphics[width=1.5in]{filename}
16 \end{minipage}
17 }
18 \caption{Minipages Inside Subfigures}
19 \label{fig:mini:subfig} %% label for entire figure
20 \end{figure}
```

实际上也是小页环境的功劳！

实际上插图手册基本就能解决所有的问题，如果仍有问题可以在小屋的问答区或者 QQ 群询问。在这里不再列举更多图形。

对于有强迫症的人来说，尽量适应而非尝试去改变。

# 使用 thebibliography 环境

由于美赛中涉及到中文参考文献翻译的问题，可以先选择符合 gbt7714-2015 样式的引用方式之后，再利用翻译软件进行翻译，写入 bibitem 之中。

```
1 \begin{thebibliography}{99}
2 \bibitem{1} D.~E. KNUTH   The \TeX{}book   the American
3 Mathematical Society and Addison–Wesley
4 Publishing Company , 1984–1986.
5 \bibitem{2} Lamport, Leslie , \LaTeX{}: ‘‘ A Document Preparation System ’’,
6 Addison–Wesley Publishing Company, 1986.
7 \bibitem{3}\url{https://www.latexstudio.net/}
8 \end{thebibliography}
```

thebibliography 旁的参数为参考文献的上限条数，最高为 99。如果需要引用则可以引用 `\cite{key}` 指令进行引用。

# 使用 bibtex 管理参考文献

```
1 \bibliographystyle{style}  
2 \bibliography{bib file}
```

该方法需要创建 \*.bib 文件，并且需要管理 bib 文件中的文献条目，可以使用谷歌学术镜像、必应学术、百度学术、知网等导出 bib 条目，然后利用合适的编译链进行编译。对于美赛的参考条目，可以尝试将 author、title、journal 翻译改为英文，但无需使用 gbt7714 宏包。但是对于美赛中有较多的网页，引用较为不便。



由于美赛与国赛不同，要求不提交支撑材料，所以书写的代码需要以 `lstlisting` 环境插入，而使用者只需要将代码放入 `code` 文件夹中并使用 `\lstinputlisting[language=...]{./code/filename}` 命令就可以插入相应的代码，demo 中举了 `c++` 以及 `MATLAB` 的样式，如果需要设置其他语言，可以将方括号内的 `language` 改为相应的语言即可。支持的语言可以利用命令 `texdoc listings` 查看文档的第 13 页，文档中的语言都支持。

# 写在最后的话

数学建模应该是大多数国人接触  $\text{\LaTeX}$  的第一次机会，当你真正会用它之后，你会爱上用它，逐渐摒弃 word。 $\text{\LaTeX}$  原本是书写《程序的艺术》的排版语言，它也成为了一种艺术的程序，如果有机会继续使用下去，你会逐渐发现它的好处。

回头再去看一些注意事项，你会发现文档远比百度好用，学会提问的重要性。

欢迎关注  $\text{\LaTeX}$  工作室的公众号



最后祝大家在比赛中取得好成绩！