

一、模板的基本使用

要使用 \LaTeX 来完成建模论文，首先要确保正确安装一个 \LaTeX 的发行版本。

- Mac 下可以使用 MacTeX
 - Linux 下可以使用 \TeX Live ;
 - windows 下可以使用 \TeX Live 或者 MikTeX ;
- 具体安装可以参考 [Install-LaTeX-Guide-zh-cn](#)

1.1 编辑器

或者其它靠谱的文章。

1.1.1 编辑器

另外可以安装一个易用的编辑器，例如 \TeX studio 。

使用该模板前，请阅读模板的使用说明文档。下面给出模板使用的大概样式。

```
\documentclass{cumcmthesis}
%\documentclass[withoutpreface,bwprint]{cumcmthesis} %去掉封面与编号页

\title{\hspace{2em}五一杯数学建模竞赛论文\LaTeX{}模板}
\tihao{A} % 题号
\duihao{4321} % 报名号
\zubie{本科}
\schoolname{你的大学}
\membera{成员A}
\memberb{成员B}
\memberc{成员C}
\email{example@email.com}
\phone{1234567890}

\begin{document}
  \maketitle
  \keywords{关键词1\quad 关键词2\quad 关键词3}
  \begin{abstract}
    摘要的具体内容。
  \end{abstract}
  \tableofcontents
  \section{问题重述}
  \subsection{问题的提出}
  \section{模型的假设}
  \section{符号说明}
  \begin{center}
    \begin{tabular}{cc}
      \hline
      \makebox[0.3\textwidth][c]{符号} & \makebox[0.4\textwidth][c]{意义} \\
      D & \& 木条宽度 (cm)
    \end{tabular}
  \end{center}
  \section{问题分析}
```

```

\section{总结}
\begin{thebibliography}{9}%宽度9
  \bibitem{bib:one} ....
\end{thebibliography}
\begin{appendices}
  附录的内容。
\end{appendices}
\end{document}

```

根据要求，电子版论文提交时需去掉封面和编号页。可以加上 `withoutpreface` 选项来实现，即：

```
\documentclass[withoutpreface]{cumcmthesis}
```

这样就能实现了。打印的时候有超链接的地方不需要彩色，可以加上 `bwprint` 选项。另外目录也是不需要的，将 `\tableofcontents` 注释或删除，目录就不会出现了。

团队的信息填入指定的位置，并且确保信息的正确性，以免因此白忙一场。

编译记得使用 `xelatex`，而不是用 `pdflatex`。在命令行编译的可以按如下方式编译：

```
xelatex example
```

或者使用 `latexmk` 来编译，更推荐这种方式。

```
latexmk -xelatex example
```

下面给出写作与排版上的一些建议。

二、图片

建模中不可避免要插入图片。图片可以分为矢量图与位图。位图推荐使用 `jpg` `png` 这两种格式，避免使用 `bmp` 这类图片，容易出现图片插入失败这样情况的发生。矢量图一般有 `pdf`, `eps`，推荐使用 `pdf` 格式的图片，尽量不要使用 `eps` 图片，理由相同。

注意图片的命名，避免使用中文来命名图片，可以用英文与数字的组合来命名图片。避免使用 `1,2,3` 这样顺序的图片命名方式。图片多了，自己都不清楚那张图是什么了，命名尽量让它有意义。下面是一个插图的示例代码。

注意 `figure` 环境是一个浮动体环境，图片的最终位置可能会跑动。`[!h]` 中的 `h` 是 `here` 的意思，`!` 表示忽略一些浮动体的严格规则。另外里面还可以加上 `btp` 选项，它们分别是 `bottom`, `top`, `page` 的意思。只要这几个参数在花括号里面，作用是不分先后顺序的。`page` 在这里表示浮动页。

`\label{fig:circuit-diagram}` 是一个标签，供交叉引用使用的。例如引用图片 `\cref{fig:circuit-diagram}` 的实际效果是图 1。图片是自动编号的，比起手动编号，它更加高效。`\cref{label}` 由 `cleveref` 宏包提供，比普通的 `\ref{label}` 更加自动化。`label` 要确保唯一，命名方式推荐用图片的命名方式。

图片并排的需求解决方式多种多样，下面用 `minipage` 环境来展示一个简单的例子。注意，以下例子用到了 `subcaption` 命令，需要加载 `subcaption` 宏包。

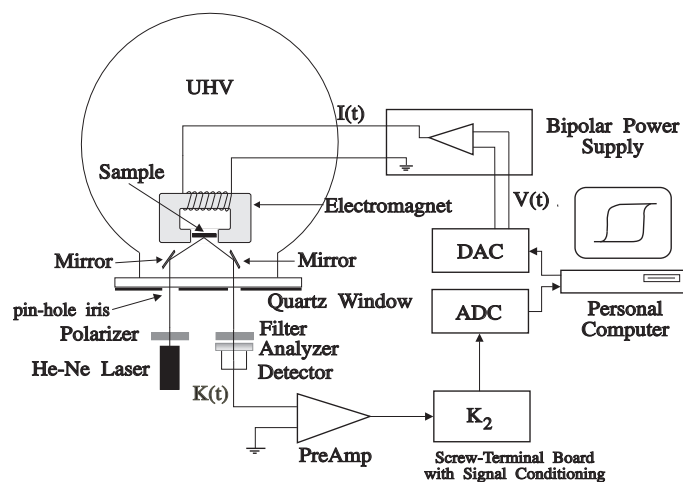


图 1 电路图

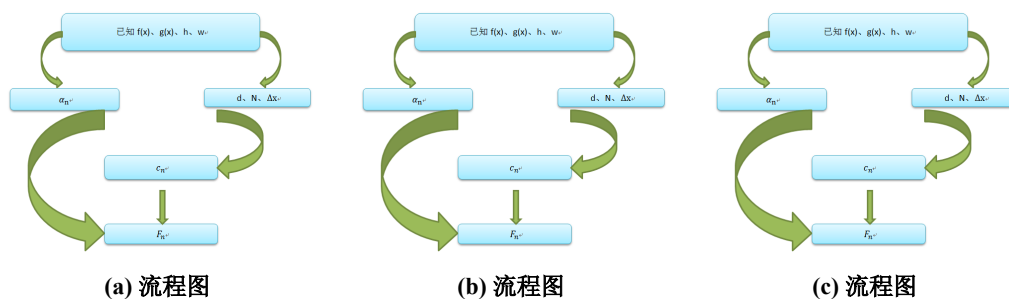
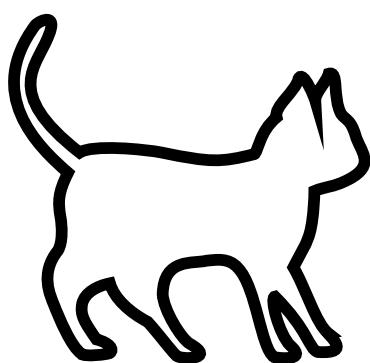


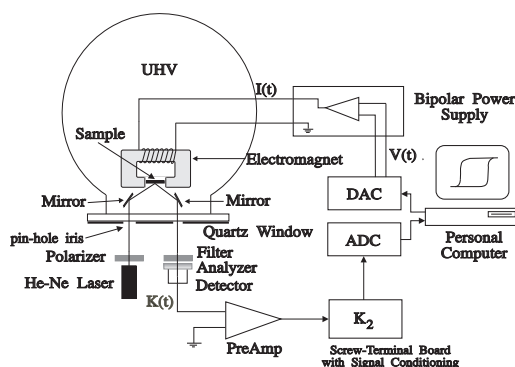
图 2 多图并排示例

这相当于整体是一张大图片，大图片引用是图 2，子图引用别分是图 2a、图 2b、图 2c。

如果原本两张图片的高度不同，但是希望它们缩放后等高的排在同一行，参考这个例子：



(a) 一只猫



(b) 电路图

图 3 多图并排示例

三、绘制普通三线表格

表格应具有三线表格式，因此常用 booktabs 宏包，其标准格式如表 1 所示。

表 1 标准三线表格

$D(\text{in})$	$P_u(\text{lbs})$	$u_u(\text{in})$	β	$G_f(\text{psi.in})$
5	269.8	0.000674	1.79	0.04089
10	421.0	0.001035	3.59	0.04089
20	640.2	0.001565	7.18	0.04089

其绘制表格的代码及其说明如下。

```
\begin{table}[!htbp]
\caption[标签名]{中文标题}
\begin{tabular}{cc...c}
\toprule[1.5pt]
表头第1个格 & 表头第2个格 & ... & 表头第n个格 \\
\midrule[1pt]
表中数据(1,1) & 表中数据(1,2) & ... & 表中数据(1,n)\\
表中数据(2,1) & 表中数据(2,2) & ... & 表中数据(2,n)\\
.....\\
表中数据(m,1) & 表中数据(m,2) & ... & 表中数据(m,n)\\
\bottomrule[1.5pt]
\end{tabular}
\end{table}
```

`table` 环境是一个将表格嵌入文本的浮动环境。`tabular` 环境的必选参数由每列对应一个格式字符所组成：`c` 表示居中，`l` 表示左对齐，`r` 表示右对齐，其总个数应与表的列数相同。此外，`@{文本}` 可以出现在任意两个上述的列格式之间，其中的文本将被插入每一行的同一位置。表格的各行以 `\\` 分隔，同一行的各列则以 `&` 分隔。`\toprule`、`\midrule` 和 `\bottomrule` 三个命令是由 `booktabs` 宏包提供的，其中 `\toprule` 和 `\bottomrule` 分别用来绘制表格的第一条（表格最顶部）和第三条（表格最底部）水平线，`\midrule` 用来绘制第二条（表头之下）水平线，且第一条和第三条水平线的线宽为 1.5pt，第二条水平线的线宽为 1pt。引用方法与图片的相同。

四、公式

数学建模必然涉及不少数学公式的使用。下面简单介绍一个可能用得上的数学环境。

首先是行内公式，例如 θ 是角度。行内公式使用 `$ $` 包裹。

行间公式不需要编号的可以使用 `\[\]` 包裹，例如

$$E = mc^2$$

其中 E 是能量， m 是质量， c 是光速。

如果希望某个公式带编号，并且在后文中引用可以参考下面的写法：

$$E = mc^2 \tag{1}$$

式 (1) 是质能方程。

多行公式有时候希望能够在特定的位置对齐，以下是其中一种处理方法。

$$P = UI \quad (2)$$

$$= I^2 R \quad (3)$$

`&` 是对齐的位置，`&` 可以有多个，但是每行的个数要相同。

矩阵的输入也不难。

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nn} \end{pmatrix}$$

分段函数这些可以用 `case` 环境，但是它要放在数学环境里面。

$$f(x) = \begin{cases} 0 & x \text{ 为无理数,} \\ 1 & x \text{ 为有理数.} \end{cases}$$

在数学环境里面，字体用的是数学字体，一般与正文字体不同。假如要公式里面有个别文字，则需要把这部分放在 `text` 环境里面，即 `\text{文本环境}`。

公式中个别需要加粗的字母可以用 `\bm{math symbol}`。如 $\alpha a \alpha a$ 。

以上仅简单介绍了基础的使用，对于更复杂的需求，可以阅读相关的宏包手册，如 `amsmath`。

希腊字母这些如果不熟悉，可以去查找符号文件 `symbols-a4.pdf`，也可以去 `de-texify` 网站手写识别。另外还有数学公式识别软件 `mathpix`。

下面简单介绍一下定理、证明等环境的使用。

定义 1 定义环境

定义 1 除了告诉你怎么使用这个环境以外，没有什么其它的意义。

除了 `definition` 环境，还可以使用 `theorem`、`lemma`、`corollary`、`assumption`、`conjecture`、`axiom`、`principle`、`problem`、`example`、`proof`、`solution` 这些环境，根据论文的实际需求合理使用。

定理 1 这是一个定理。

由定理 1 我们知道了定理环境的使用。

引理 1 这是一个引理。

由引理 1 我们知道了引理环境的使用。

推论 1 这是一个推论。

由推论 1 我们知道了推论环境的使用。

假设 1 这是一个假设。

由假设 1 我们知道了假设环境的使用。

猜想 1 这是一个猜想。

由猜想 1 我们知道了猜想环境的使用。

公理 1 这是一个公理。

由公理 1 我们知道了公理环境的使用。

定律 1 这是一个定律。

由定律 1 我们知道了定律环境的使用。

问题 1 这是一个问题。

由问题 1 我们知道了问题环境的使用。

例 1 这是一个例子。

由例 1 我们知道了例子环境的使用。

证明 1 这是一个证明。

由证明 1 我们知道了证明环境的使用。

解 1 这是一个解。

由解 1 我们知道了解环境的使用。

五、其它小功能

5.1 脚注

利用 `\footnote{具体内容}` 可以生成脚注¹。

5.2 无序列表与有序列表

无序列表是这样的：

- one
- two
- ...

有序列表是这样子的：

1. one
2. two
3. ...

¹脚注可以补充说明一些东西

5.3 字体加粗与斜体

如果想强调部分内容，可以使用加粗来实现。加粗字体可以用 `\textbf{加粗}` 来实现。例如：**这是加粗的字体。This is bold fonts。**

中文字体没有斜体设计，但是英文字体有。斜体 *Italics*。

六、参考文献与引用

参考文献对于一篇正式的论文来说是必不可少的，在建模中重要的参考文献当然应该列出。 \LaTeX 在这方面的功能也是十分强大的，下面介绍一个比较简单的参考文献制作方法。有兴趣的可以学习 `bibtex` 或 `biblatex` 的使用。

\LaTeX 的入门可以看《 \LaTeX 入门》[1]。这是一个简单的引用，用 `\cite{bibkey}` 来完成。要引用成功，当然要维护好 `bibitem` 了。下面是个简单的例子。

参考文献

[1] 刘海洋. \LaTeX 入门[J]. 电子工业出版社, 北京, 2013.

[2] 全国大学生数学建模竞赛论文格式规范 (2023 年修改).

[1] <https://www.latexstudio.net>

附录 A 排队算法—matlab 源程序

```
kk=2; [mdd, ndd]=size(dd);
while ~isempty(V)
    [tmpd, j]=min(W(i, V)); tmpj=V(j);
    for k=2:ndd
        [tmp1, jj]=min(dd(1, k)+W(dd(2, k), V));
        tmp2=V(jj); tt(k-1, :)= [tmp1, tmp2, jj];
    end
    tmp=[tmpd, tmpj, j; tt]; [tmp3, tmp4]=min(tmp(:, 1));
    if tmp3==tmpd, ss(1:2, kk)= [i; tmp(tmp4, 2)];
    else, tmp5=find(ss(:, tmp4)~=0); tmp6=length(tmp5);
    if dd(2, tmp4)==ss(tmp6, tmp4)
        ss(1:tmp6+1, kk)= [ss(tmp5, tmp4); tmp(tmp4, 2)];
    else, ss(1:3, kk)= [i; dd(2, tmp4); tmp(tmp4, 2)];
    end; end
    dd= [dd, [tmp3; tmp(tmp4, 2)]]; V(tmp(tmp4, 3))=[];
    [mdd, ndd]=size(dd); kk=kk+1;
end; S=ss; D=dd(1, :);
```

附录 B 规划解决程序—lingo 源代码

```
kk=2;
[mdd, ndd]=size(dd);
while ~isempty(V)
    [tmpd, j]=min(W(i, V)); tmpj=V(j);
    for k=2:ndd
        [tmp1, jj]=min(dd(1, k)+W(dd(2, k), V));
        tmp2=V(jj); tt(k-1, :)= [tmp1, tmp2, jj];
    end
    tmp=[tmpd, tmpj, j; tt]; [tmp3, tmp4]=min(tmp(:, 1));
    if tmp3==tmpd, ss(1:2, kk)= [i; tmp(tmp4, 2)];
    else, tmp5=find(ss(:, tmp4)~=0); tmp6=length(tmp5);
    if dd(2, tmp4)==ss(tmp6, tmp4)
        ss(1:tmp6+1, kk)= [ss(tmp5, tmp4); tmp(tmp4, 2)];
    else, ss(1:3, kk)= [i; dd(2, tmp4); tmp(tmp4, 2)];
    end;
end
    dd= [dd, [tmp3; tmp(tmp4, 2)]]; V(tmp(tmp4, 3))=[];
    [mdd, ndd]=size(dd);
    kk=kk+1;
end;
S=ss;
D=dd(1, :);
```