

(请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。)

赛区评阅编号：\_\_\_\_\_  
(由赛区填写)

全国评阅编号：\_\_\_\_\_  
(全国组委会填写)

**2022 高教社杯全国大学生数学建模竞赛**

**编 号 专 用 页**

赛区评阅记录（可供赛区评阅时使用）：

评 阅 人						
备 注						

送全国评阅统一编号：  
(赛区组委会填写)

(请勿改动此页内容和格式。此编号专用页仅供赛区和全国评阅使用，参赛队打印后装订到纸质论文的第二页上。注意电子版论文中不得出现此页。)

# 全国大学生数学建模竞赛编写的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 模板

## 摘要

cumcmthesis 是为全国大学生数学建模竞赛编写的  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  模板, 旨在让大家专注于论文的内容写作, 而不用花费过多精力在格式的定制和调整上. 本手册是相应的参考, 其中提供了一些环境和命令可以让模板的使用更为方便. 同时需要注意, 使用者需要有一定的  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  的使用经验, 至少要会使用常用宏包的一些功能, 比如参考文献, 数学公式, 图片使用, 列表环境等等. 例子文件参看 `example.tex`.

### 2020 年建模比赛格式变化说明

今年的格式变化主要就是三个地方, 如下:

1. 论文第一页为承诺书, **内容进行了调整**。
2. 编号页格式进行了格式调整。
3. 这是 19 年调整了, 这里延续说明下。论文正文 (**不要目录**, 尽量控制在 20 页以内); 正文之后是论文附录 (页数不限)。

<https://www.latexstudio.net> 陆续推出了更优质的资源, 欢迎学习。

欢迎大家到 QQ 群里沟通交流: 91940767/478023327/640633524。我们也开通了问答区交流  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  技术: <https://ask.latexstudio.net>, 欢迎大家前来交流, 有问题就来这里, 与大神零距离。

关注我们的微信公众号:



关键词:  $\text{T}_{\text{E}}\text{X}$  图片 表格 公式

## 一、模板的基本使用

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

- Mac 下可以使用  $\text{MacTeX}$
- Linux 下可以使用  $\text{TeXLive}$  ;
- windows 下可以使用  $\text{TeXLive}$  或者  $\text{MikTeX}$  ;

具体安装可以参考 [Install-Latex-Guide-zh-cn](#) 或者其它靠谱的文章。另外可以安装一个易用的编辑器，例如  $\text{TeXstudio}$  。

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。

```
1 \documentclass{cumcmthesis}
2 %\documentclass[withoutpreface,bwprint]{cumcmthesis} ←
   %去掉封面与编号页
3
4 \title{论文题目}
5 \tiahao{A}           % 题号
6 \baominghao{4321}   % 报名号
7 \schoolname{你的大学}
8 \membera{成员A}
9 \memberb{成员B}
10 \memberc{成员C}
11 \supervisor{指导老师}
12 \yearinput{2017}    % 年
13 \monthinput{08}     % 月
14 \dayinput{22}       % 日
15
16 \begin{document}
17     \maketitle
18     \begin{abstract}
19         摘要的具体内容。
20         \keywords{关键词1\quad 关键词2\quad 关键词3}
21     \end{abstract}
22     \tableofcontents
23     \section{问题重述}
24     \subsection{问题的提出}
25     \section{模型的假设}
26     \section{符号说明}
```

```

27 \begin{center}
28 \begin{tabular}{cc}
29 \hline
30 \makebox[0.3\textwidth][c]{符号} & \leftrightarrow \\
31 \makebox[0.4\textwidth][c]{意义} & \\\ \hline
32 D & 木条宽度 (cm) \\\ \hline
33 \end{tabular}
34 \end{center}
35 \section{问题分析}
36 \section{总结}
37 \begin{thebibliography}{9}%宽度9
38 \bibitem{bib:one} ....
39 \end{thebibliography}
40 \begin{appendices}
41 附录的内容。
42 \end{appendices}
43 \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：

```
1 \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：

```
1 xelatex example
```

或者使用 `latexmk` 来编译，更推荐这种方式。

```
1 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。

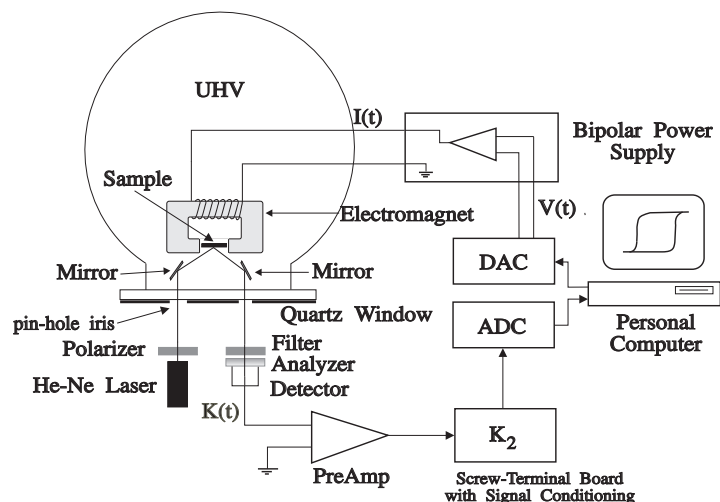


图 1 电路图

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{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。

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：

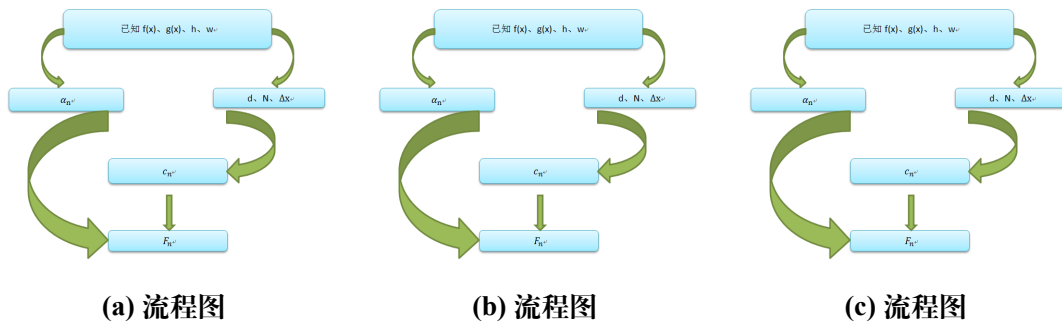
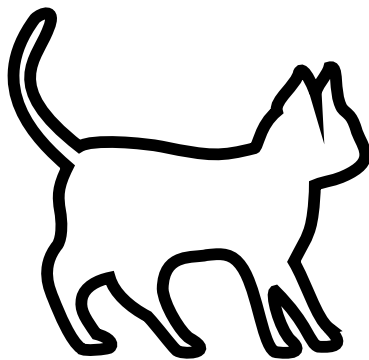
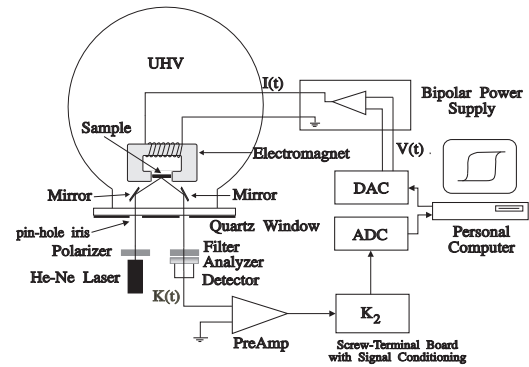


图 2 多图并排示例



(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

$D(\text{in})$	$P_u(\text{lbs})$	$u_u(\text{in})$	$\beta$	$G_f(\text{psi.in})$
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。

```

1 \begin{table}[!htbp]
2   \caption[标签名]{中文标题}
3   \begin{tabular}{cc...c}

```

```

4      \toprule[1.5pt]
5      表头第1个格    & 表头第2个格    & ... & 表头第n个格    \\
6      \midrule[1pt]
7      表中数据(1,1) & 表中数据(1,2) & ... & 表中数据(1,n)\\
8      表中数据(2,1) & 表中数据(2,2) & ... & 表中数据(2,n)\\
9      .....\\
10     表中数据(m,1) & 表中数据(m,2) & ... & 表中数据(m,n)\\
11     \bottomrule[1.5pt]
12     \end{tabular}
13     \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  $\theta$  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。

$$P = UI \tag{2}$$

$$= I^2 R \tag{3}$$



`&` 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 `detexify` 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<sup>1</sup>。

### 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. ...

---

<sup>1</sup>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,可以使用加粗的手段来实现。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。 $\text{\LaTeX}$  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。

$\text{\LaTeX}$  的入门书籍可以看《 $\text{\LaTeX}$  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.  $\text{\LaTeX}$  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 (2020 年 8 月 25 日修改).

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

## 附录 A 模板所用的宏包

表 2 宏包罗列

模板中已经加载的宏包				
ambsy	amsfonts	amsgen	amsmath	amsopn
amssymb	amstext	appendix	array	atbegshi
atveryend	auxhook	bigdelim	bigintcalc	bigstrut
bitset	bm	booktabs	calc	caption
caption3	CJKfntef	cprotect	ctex	ctexhook
ctexpatch	enumitem	etexcmds	etoolbox	everyisel
expl3	fix-cm	fontenc	fontspec	fontspec-xetex
geometry	getttitlestring	graphics	graphicx	hobsub
hobsub-generic	hobsub-hyperref	hopatch	hxdetex	hycolor
hyperref	ifluatex	ifpdf	ifthen	ifvtex
ifxetex	indentfirst	infwarrerr	intcalc	keyval
kvdefinekeys	kvoptions	kvsetkeys	l3keys2e	letltxmacro
listings	longtable	lstmisc	ltxcaption	ltxcmds
multirow	nameref	pdfescape	pdftexcmds	refcount
rerunfilecheck	stringenc	suffix	tabularx	titletoc
tocloft	trig	ulem	uniquecounter	url
xcolor	xcolor-patch	xeCJK	xeCJKfntef	xeCJK-listings
xparse	xtemplate	zhnumber		

以上宏包都已经加载过了，不要重复加载它们。

## 附录 B 排队算法—matlab 源程序

```

1  kk=2;[mdd,ndd]=size(dd);
2  while ~isempty(V)
3  [tmpd,j]=min(W(i,V));tmpj=V(j);
4  for k=2:ndd
5  [tmp1,jj]=min(dd(1,k)+W(dd(2,k),V));
6  tmp2=V(jj);tt(k-1,:)=[tmp1,tmp2,jj];
7  end
8  tmp=[tmpd,tmpj,j;tt];[tmp3,tmp4]=min(tmp(:,1));
9  if tmp3==tmpd, ss(1:2,kk)=[i;tmp(tmp4,2)];
10 else,tmp5=find(ss(:,tmp4)~=0);tmp6=length(tmp5);
11 if dd(2,tmp4)==ss(tmp6,tmp4)
12 ss(1:tmp6+1,kk)=[ss(tmp5,tmp4);tmp(tmp4,2)];
13 else, ss(1:3,kk)=[i;dd(2,tmp4);tmp(tmp4,2)];
14 end;end
15 dd=[dd,[tmp3;tmp(tmp4,2)]];V(tmp(tmp4,3))=[];
16 [mdd,ndd]=size(dd);kk=kk+1;
17 end; S=ss; D=dd(1,:);

```

## 附录 C 规划解决程序—lingo 源代码

```

1  kk=2;
2  [mdd,ndd]=size(dd);
3  while ~isempty(V)
4      [tmpd,j]=min(W(i,V));tmpj=V(j);
5  for k=2:ndd
6      [tmp1,jj]=min(dd(1,k)+W(dd(2,k),V));
7      tmp2=V(jj);tt(k-1,:)=[tmp1,tmp2,jj];
8  end
9      tmp=[tmpd,tmpj,j;tt];[tmp3,tmp4]=min(tmp(:,1));
10 if tmp3==tmpd, ss(1:2,kk)=[i;tmp(tmp4,2)];
11 else,tmp5=find(ss(:,tmp4)~=0);tmp6=length(tmp5);
12 if dd(2,tmp4)==ss(tmp6,tmp4)
13     ss(1:tmp6+1,kk)=[ss(tmp5,tmp4);tmp(tmp4,2)];
14 else, ss(1:3,kk)=[i;dd(2,tmp4);tmp(tmp4,2)];
15 end;
16 end
17     dd=[dd,[tmp3;tmp(tmp4,2)]];V(tmp(tmp4,3))=[];

```

```

18     [mdd,ndd]=size(dd);
19     kk=kk+1;
20 end;
21 S=ss;
22 D=dd(1,:);

```

Listing 1: 2.py

```

1  import os.path
2
3  filename='student.txt'
4
5  def main():
6      while True:
7          menu()
8          choice=int(input('请选择'))
9          if choice in [0,1,2,3,4,5,6,7]:
10             if choice ==0:
11                 answer=input('您确定要退出系统吗? y/n')
12                 if answer=='y'or answer=='Y':
13                     print('感谢您的使用!!!')
14                     break #退出系统
15             else:
16                 continue
17             elif choice==1:
18                 insert()#录入学生信息
19             elif choice==2:
20                 search()
21             elif choice==3:
22                 delete()
23             elif choice==4:
24                 modify()
25             elif choice==5:
26                 sort()
27             elif choice==6:
28                 total()
29             elif choice==7:
30                 show()
31         else:

```

```

32         continue
33
34 def menu():
35     print('=====学生信息管理系统=====')
36     print('-----功能菜单-----')
37     print('\t\t\t\t\t\t\t1、录入学生信息')
38     print('\t\t\t\t\t\t\t2、查找学生信息')
39     print('\t\t\t\t\t\t\t3、删除学生信息')
40     print('\t\t\t\t\t\t\t4、修改学生信息')
41     print('\t\t\t\t\t\t\t5、排序')
42     print('\t\t\t\t\t\t\t6、统计学生总人数')
43     print('\t\t\t\t\t\t\t7、显示所有学生信息')
44     print('\t\t\t\t\t\t\t0、退出')
45     print('-----')
46
47 def insert():
48     student_list=[]
49     while True:
50         id=input('请输入id（如1001）')
51         if not id:
52             break
53         name=input('请输入姓名')
54         if not name:
55             break
56
57         try:#加一个小循环
58             englist=int(input('请输入英语成绩：'))
59             python=int(input('请输入python成绩：'))
60             java=int(input('请输入Java成绩：'))
61         except:
62             print('输入无效，不是整数类型，请重新输入。')
63             continue
64         #将录入的学术信息保存到字典当中
65         student={'id':id,'name':name,'english':englist,'python':python←
        , 'java':java}
66         #将学生信息添加到列表中
67         student_list.append(student)
68         answer=input('是否继续添加？y/n\n')
69         if answer=='y':
70             continue

```

```

71         else:
72             break
73
74     #调用save () 函数
75     save(student_list)
76     print('学生系统录入完毕!!!! ')
77
78 def save(lst):
79     try:
80         stu_txt=open(filename,'a',encoding='utf-8')#创建或追加
81
82     except:
83         stu_txt=open(filename,'w',encoding='utf-8')#创建或覆盖
84     for item in lst:
85         stu_txt.write(str(item)+'\n')
86     stu_txt.close()
87
88
89 def search():
90     while True:
91         flag=False
92         student_query=[]
93         num=int(input('输入姓名查询请按1，输入ID查询请按2'+'\n'))
94         if num in[1,2]:
95             if num==1:
96                 student_name=input('请输入要查询的学生姓名:')
97                 if os.path.exists(filename):
98                     with open(filename, 'r', encoding='utf-8') as file↵
99                         :
100                             student_old = file.readlines() # 输出结果为列↵
101                             表 ['{1001}']
102             else:
103                 student_old = []
104             if student_old:
105                 with open(filename, 'r', encoding='utf-8') as ↵
106                     rfile:
107                         d = {}
108                         for item in student_old:
109                             d = dict(eval(item)) # 将字符串转成字典
110                             if d['name'] == student_name:

```



```

108         student_query.append(d)
109         show_student(student_query)
110         flag=True
111         break
112     else:
113         continue
114 if flag:
115     print(f'已经查询到名字为{student_name}相关←
        的学生信息')
116 else:
117     print(f'抱歉，没有查询到名字为{←
        student_name}相关的学生信息')
118 else:
119     print('无学生信息')
120 elif num==2:
121     student_id = input('请输入要查询的学生ID:')
122     if os.path.exists(filename):
123         with open(filename, 'r', encoding='utf-8') as file←
124             :
125             student_old = file.readlines() # 输出结果为列←
126             表 ['{1001}']
127     else:
128         student_old = []
129     if student_old:
130         with open(filename, 'r', encoding='utf-8') as ←
131             rfile:
132                 d = {}
133                 for item in student_old:
134                     d = dict(eval(item)) # 将字符串转成字典
135                     if d['id'] == student_id:
136                         student_query.append(d)
137                         show_student(student_query)
138                         flag = True
139                         break
140                     else:
141                         continue
142 if flag:
143     print(f'已经找到ID为{student_id}的学生信息←
        ')
144 else:

```

```

142         print(f'抱歉，没有查询到ID为{student_id}的←
           学生信息')
143     else:
144         print('无学生信息')
145     answer=input('是否继续查询学生信息y/n:')
146     if answer=='y':
147         continue
148     else:
149         break
150
151     else:
152         print('输入错误，请重新输入')
153         continue
154
155 def show_student(lst):
156     format_title='{:^6}\t{:^12}\t{:^8}\t{:^10}\t{:^10}\t{:^8}'
157     print(format_title.format('ID','姓名','英语成绩','python成绩','←
           java成绩','总成绩'))
158     format_data='{:^6}\t{:^12}\t{:^8}\t{:^10}\t{:^10}\t{:^8}'
159     for item in lst:
160         print(format_data.format(item['id'],
161                                   item['name'],
162                                   item['english'],
163                                   item['python'],
164                                   item['java'],
165                                   int(item['english'])+int(item['←
           python'])+int(item['java'])
166                                   )
167         )
168
169 def delete():
170     while True:
171         student_id=input('请输入要删除的学生ID')
172         if student_id!='':
173             if os.path.exists(filename):
174                 with open(filename,'r',encoding='utf-8')as file:
175                     student_old=file.readlines()#输出结果为列表 ←
176                     ['{1001}']
177             else:
178                 student_old=[]

```

```

178         flag=False #标记是否删除
179     if student_old:
180         with open(filename,'w',encoding='utf-8') as wfile:
181             d={}
182             for item in student_old:
183                 d=dict(eval(item))#将字符串转成字典
184                 if d['id']!=student_id:
185                     wfile.write(str(d)+'\n')
186                 else:
187                     flag=True
188             if flag:
189                 print(f'id为{student_id}的学生信息已被删除')
190             else:
191                 print(f'没有找到ID为{student_id}的学生信息')
192     else:
193         print('无学生信息')
194         break
195     show()#删除之后要重新显示所有学生信息
196     answer=input('是否继续删除?y/n\n')
197     if answer=='y':
198         continue
199     else:
200         break
201 def modify():
202     show()
203     if os.path.exists(filename):
204         with open(filename,'r',encoding='utf-8')as rfile:
205             student_old=rfile.readlines()
206     else:
207         return
208     student_id=input('请输入要修改的学员的ID: ')
209     with open(filename,'w',encoding='utf-8') as wfile:
210         for item in student_old:
211             d=dict(eval(item))
212             if d['id']==student_id:
213                 print('找到学生信息，可以修改他的相关信息了! ')
214                 show()
215                 while True:
216                     try:
217                         d['name']=input('请输入姓名:')

```

```

218         d['english']=input('请输入英语成绩:')
219         d['python']=input('请输入python成绩:')
220         d['java']=input('请输入Java成绩:')
221     except:
222         print('您的输入有误，请重新输入!!!')
223     else:
224         break
225     wfile.write(str(d)+'\n')
226     print('修改成功!!!')
227 else:
228     wfile.write(str(d)+'\n')
229     answer=input('是否继续修改其他学生的信息: y/n')
230 if answer=='y':
231     modify()
232
233
234
235 def sort():
236     show()
237     if os.path.exists(filename):
238         with open(filename,'r',encoding='utf-8') as rfile:
239             student_list=rfile.readlines()
240             student_new=[]
241             for item in student_list:
242                 d=eval(item)
243                 student_new.append(d)
244     else:
245         return
246     asc_or_desc=int(input('请选择 (0.升序 1.降序): '))
247     if asc_or_desc==0:
248         asc_or_desc=False
249     elif asc_or_desc==1:
250         asc_or_desc=True
251     else:
252         print('你的输入有误，请重新输入')
253     sort()
254     mode=input('请选择排序方式 (1.按英语成绩排序 2、按python成绩排序 3←
        、按Java成绩排序 0、按总成绩排序: ')
255     if mode=='1':
256         student_new.sort(key=lambda x:int(x['english']),reverse=↵

```

```

        asc_or_desc)
257 elif mode=='2':
258     student_new.sort(key=lambda x: int(x['python']), reverse=asc_or_desc)
        asc_or_desc)
259 elif mode=='3':
260     student_new.sort(key=lambda x: int(x['java']), reverse=asc_or_desc)
        asc_or_desc)
261 elif mode=='0':
262     student_new.sort(key=lambda x: int(x['english'])+int(x['java'])+int(x['python']), reverse=asc_or_desc)
263 else:
264     print('你的输入有误, 请重新输入!!! ')
265     sort()
266 show_student(student_new)
267 def total():
268     if os.path.exists(filename):
269         with open(filename, 'r', encoding='utf-8') as rfile:
270             students=rfile.readlines()
271             if students:
272                 print(f'一共有{len(students)}名学生')
273             else:
274                 print('还没有录入学生信息')
275     else:
276         print('暂未保存数据')
277
278 def show():
279     student_lst=[]
280     if os.path.exists(filename):
281         with open(filename, 'r', encoding='utf-8') as rfile:
282             students=rfile.readlines()
283             for item in students:
284                 student_lst.append(eval(item))
285             if student_lst:
286                 show_student(student_lst)
287     else:
288         print('暂未保存数据!!! ')
289
290
291 if __name__ == '__main__':
292     main()

```