

# CCNUthesis 用户手册

夏康玮 林康益

kangweixia\_xdyy@163.com

2022-01-31 v0.0.1<sup>\*</sup>

## 目录

<b>1</b>	<b>计算机基础知识</b>	<b>2</b>
1.1	编译环境和代码编辑器	2
1.1.1	计算机语言	2
1.1.2	代码编辑器	2
1.1.3	发行版	3
1.2	命令行	4
1.2.1	GUI 与 CLI	4
1.2.2	Windows 10 系统下的命令行基本操作	4
1.2.3	Windows Terminal	6
1.2.4	重要的环境变量 Path	6
1.3	编码	7
1.4	PDF 阅读器	7
1.4.1	Adobe Acrobat Reader	7
1.4.2	SumatraPDF	7
1.4.3	WPS	8

---

<sup>\*</sup><https://github.com/xkwxddy/CCNUthesis>

<sup>†</sup><https://gitee.com/xkwxddy/CCNUthesis>

<b>2</b>	<b>L<sup>A</sup>T<sub>E</sub>X 基础知识</b>	<b>9</b>
2.1	安装相关 . . . . .	9
2.1.1	安装 T <sub>E</sub> XLive . . . . .	9
2.1.2	安装外置 PDF 阅读器 . . . . .	9
2.1.3	安装 Visual Studio Code . . . . .	9
2.2	L <sup>A</sup> T <sub>E</sub> X 知识补充 . . . . .	9
2.2.1	表格 . . . . .	9
2.2.2	选择选项排版 . . . . .	9

# 1 计算机基础知识

开源排版系统  $\text{\LaTeX}$  与大家之前熟悉的软件有较多的不同，在使用过程中很可能遇到各种困难。没有任何编程基础的同学最好先阅读此部分。

## 1.1 编译环境和代码编辑器

常看到这样的提问“为什么我的  $\text{\LaTeX}$  界面和别人不一样? ”，“非得安装  $\text{TeXstudio}$  吗? ”，“安装完桌面为什么没有图标? ”；学习  $\text{LaTeX}$  时也会碰到许多相似的概念如  $\text{TeX Live}$ ， $\text{CTEX}$ ， $\text{xelatex}$ ， $\text{TeXworks}$  等，使人感到迷惑。这些问题的根源在于， $\text{\LaTeX}$  是一种计算机语言，与 Word 等软件并不一样，下面就来一一阐述。

### 1.1.1 计算机语言

人类社会有汉语，英语，法语等多种语言。然而，计算机并不能读懂人类的语言，要想指挥计算机完成各种任务，就要使用计算机语言。计算机能直接识别运行的只有由 0 和 1 组成的二进制代码，称为**机器语言**，由于其可读性太差，科学家又创造了 C，Java 等采用贴近人类语言的语法格式（主要是英语）描述程序的**编程语言**，并开发了对应的**编译器**，可以将编程语言翻译为计算机能识别的二进制代码来运行，这个翻译的过程称作**编译**，在编译前的文件称为**源代码**。

$\text{\LaTeX}$  是一种宏语言，在排版时，通过各种各样的指令对文档的各种格式（比如字体，行距，居中，编号）进行控制，一份  $\text{\LaTeX}$  源代码中既含有需要输出在文档中的具体内容，也含有控制指令。通过调用对应的**编译器**进行**编译**，最后得到排版完成的 PDF 文档。根据不同的需求， $\text{\LaTeX}$  系统有不同的编译器，下面将其称为**编译引擎**。主要有  $\text{xelatex}$ （主要处理中文文档）， $\text{pdflatex}$ （主要处理英文文档）， $\text{bibtex/biber}$ （用于引入参考文献）， $\text{lualatex}$  等。我们说“采用  $\text{xelatex}$  方式编译文档”，指的就是调用**编译引擎**  $\text{xelatex}$  编译源代码文件。

### 1.1.2 代码编辑器

通过上面的介绍，你会发现，当你编写好代码后，要得到排版好的文件，起作用的其实是编译器，而采用什么样的代码编辑器写源代码，其实是不重要的。实际上，如果你安装正确，你也可以使用记事本来编写代码，只要知道如何调用编译引擎。

那么代码编辑器有什么用呢？前面提到，代码需要通过编译才能得到我们要的 PDF 文档，但是，可能因为疏忽，编写某些指令有语法错误，那么这个时候，编译就会报错，也就无法得到 PDF 文档（当然，有可能报错了，但是得到了正确排版的 PDF 文档）。此时就需要查找错误，查找错误的过程被程序员称为 **Debug**。

如果使用记事本这种编辑器，你会发现查找错误非常痛苦，因为代码和你要排版的内容直接混在了一起，都是白纸黑字。许多代码编辑器都含有**语法高亮**的功能，

会把不同的代码自动变成不同的颜色，这样，查找起来就比较方便了，语法高亮的效果如图 (以 VS Code 为例)。

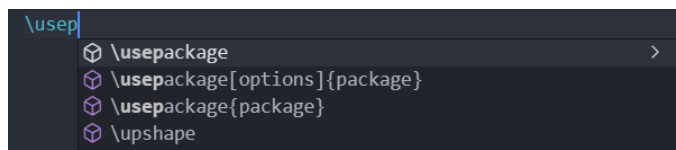
```
\LaTeX{}是一种宏语言，在排版时，通过各种各样的指令对文档的各种格式（比如字体，行距，居中，编号）进行控制，一份\LaTeX{}源代码中既含有需要输出在文档中的具体内容，也含有控制指令，通过调用对应的\textbf{}编译器进行\textbf{}编译，最后得到排版完成的PDF文档。根据不同的需求，\LaTeX{}系统有不同的编译器，下面将其称为\textbf{}编译引擎。主要有xelatex（主要处理中文文档），pdfelatex（主要处理英文文档），bibtex/biblatex（用于引入参考文献），lualatex等。我们说“采用xelatex方式编译文档”，指的就是调用\textbf{}编译引擎xelatex编译源代码文件。
```

**\subsubsection{代码编辑器}**

通过上面的介绍，你会发现，当你编写好代码后，要得到排版好的文件，起作用的其实是编译器，而采用什么样的代码编辑器写源代码，其实是不重要的。实际上，如果你安装正确，你也可以使用记事本来编写代码，只要知道如何调用编译引擎。

那么代码编辑器有什么用呢？前面提到，代码需要通过编译才能得到我们要的PDF文档，但是，可能因为疏忽，编写某些指令有语法错误，那么这个时候，编译就会报错，也就无法得到PDF文档（当然，有可能报错了，但是得到了正确排版的PDF文档）。此时就需要查找错误，查找错误的过程被程序员称为\textbf{}Debug。

另一个重要的功能是**语法补全**，在输入各种指令时，非常容易打错，或者掉个把符号比如括号，许多代码编辑器在设定好你的编程语言后，当你输入某些指令的前几个字符时，就会自动帮你联想可能的指令（很多时候就是你想要的），这样就不容易打错，并且很多时候，插入（等括号字符时，会自动成对出现，避免漏掉。效果如图



第三个方面是编译运行， $\text{\LaTeX}$  有些代码编辑器如 TeXstudio，不仅可以写代码，也提供了一键编译的按钮，如果你的编辑器没有编译的功能，就需要使用命令行编译（详见1.2节），还有一些功能只能通过命令行使用。这样对初学者来说上手起来就比较快。在安装  $\text{\LaTeX}$  时，一般会附带安装一个编辑器 TeXworks，同样提供了高亮，补全，一键编译的效果，但整体比较简陋，没有 TeXstudio 功能那么多，所以就会出现很多人推荐 TeXstudio，导致很多人误解安装  $\text{\LaTeX}$  是安装 TeXstudio。

个人比较喜欢的代码编辑器是 VS Code，这是微软开发的开源编辑器，具有很强的自定义性，还有各种各样实用的插件，可根据自己的需求配置各种指令和快捷键，不过编译环境需要自己配置，可以参考这个链接：[使用 VSCode 编写 LaTeX](#)，推荐对  $\text{\LaTeX}$  有一定了解后，再更换代码编辑器提高效率。

### 1.1.3 发行版

发行版指的是  $\text{\LaTeX}$  整个软件包的版本， $\text{\TeX}$  Live,  $\text{\CTEX}$ ,  $\text{\MiKTeX}$  指的就是发行版，通常问安装哪个版本，指的是哪个发行版。有些模版和代码只能特定的发行版下编译，比如本论文模版基于  $\text{\TeX}$  Live，邓老师的旧模板基于  $\text{\CTEX}$ ，不同的发行版通常不能兼容，因此在一般情况下**请勿安装超过一个发行版！**

一个发行版主要有三个部分：各种编译器，许许多多的宏包，宏包的说明文档。宏包可以理解为指导集，提供了更多的排版指令，当你需要对应指令时，只要加载对应的宏包就可以调用许多方便的指令。

因此，你会发现其实安装  $\text{\LaTeX}$ ，是安装这一门语言的**编译环境**，使得你可以在自己的电脑上编译  $\text{\LaTeX}$  源文件。前面说过，如果编辑器没有编译的功能，就需要使用命令行编译（详见1.2节），本质是电脑运行程序的另一种方式。因此，**安装完成后，桌面没有图标！**需要自己尝试编译一个含中文的文件，如果编译成功，才能说明安装成功，或者依照手册《install-latex-guide-zh-cn》。

## 1.2 命令行

### 1.2.1 GUI 与 CLI

我们现在的计算机操作用户界面采用图形方式显示，允许用户使用鼠标等输入设备操纵屏幕上的图标或菜单选项，这种界面称作**图形用户界面 (Graphical User Interface, 简称 GUI)**。我们平常使用电脑主要都是在这样的界面进行的，通过鼠标打开文件，运行程序，非常方便。

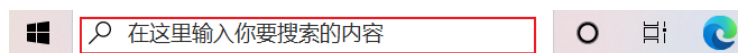
$\text{\LaTeX}$  是由美国计算机学家莱斯利·兰伯特 (Leslie Lamport) 在 20 世纪 80 年代初期开发的。在那个年代，图形界面的运用并不广泛，甚至鼠标也不普及，使用最为广泛的用户界面是**命令行界面 (Command-line Interface, 简称 CLI)**，这种界面通常不支持鼠标，用户通过键盘输入指令，计算机收到指令后，予以执行。

正是因为这样， $\text{\LaTeX}$  是按照命令行界面的用户设计的，这就是为什么**无论什么编译方式 (包括 *Texstudio* 的一键编译)**，其本质都是通过**命令行运行指令调用相应的编译引擎**，因此，我们需要学习一定的命令行知识。实际上，大部分的计算机语言都是类似这样，通过编译得到我们见到的软件。

### 1.2.2 Windows 10 系统下的命令行基本操作

下面的操作均在 Windows 10 家庭版上完成，Mac 和 Linux 上的操作类似。

在 Windows 中，命令行程序是“命令提示符”或“Windows Powershell”，可以利用菜单栏的搜索框（如图）查找。



运行时，建议右键选择“以管理员身份运行”，两个程序如图



打开命令行窗口后（以“命令提示符为例”），会显示命令提示符（图中的红方框），由当前盘符、目录（即文件夹）和一个大于号 > 组成，图中表示**当前目录为 C 盘 WINDOWS 文件夹下的 system32 文件夹**。

```

ca 管理员: 命令提示符
Microsoft Windows [版本 10.0.19042.928]
(c) Microsoft Corporation。保留所有权利。

C:\WINDOWS\system32>

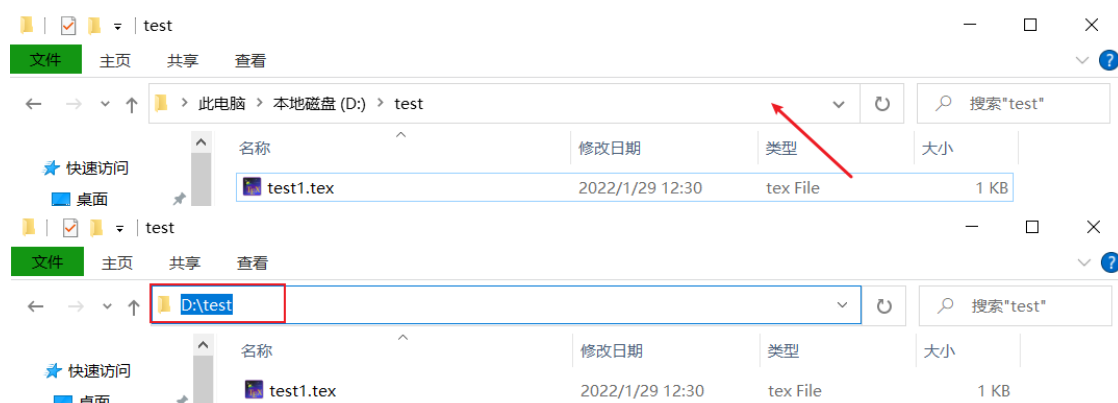
```

打开后在 > 的后面会有一个光标闪烁，等待输入命令，Windows 命令行命令和文件名不区分大小写，输入一行命令后按回车键执行，**请注意把输入法调成英文半角，尤其是输入: 和\的时候。**

下面以“用命令 `xelatex --shell-escape test1.tex` 编译 D 盘下一个 tex 文件”为例进行演示

### Step1: 确定要编译的文件路径

首先找到你要编译的文件，在文件的目录的地址栏单击鼠标左键，被选中的部分就是这个文件的路径。



如图所示，源文件的存放路径是 `D:\test`

### Step2: 将命令行的当前目录切换为要编译的文件路径

就像我们打开文件夹一样，只有命令行的当前目录为要编译的文件路径，才能进行编译，不然编译器根本找不到要编译的文件。

首先要从 C 盘更改为 D 盘，输入命令 `D:` 后按回车键运行，会发现当前目录变成 D 盘。然后输入命令 `cd \test` 即可进入 D 盘下的 test 文件夹（目录）。

```

C:\WINDOWS\system32>D:
D:\>cd \test
D:\test>

```

### Step3: 编译

输入命令 `xelatex --shell-escape test1.tex` 后按回车键开始编译，“test1.tex”是要编译的文件的文件名。

等到再次出现命令提示符时（如图所示），说明编译完成，此时可以回到原来的目录查看编译得到的 PDF 文件。

```
Output written on test1.pdf (1 page).
Transcript written on test1.log.

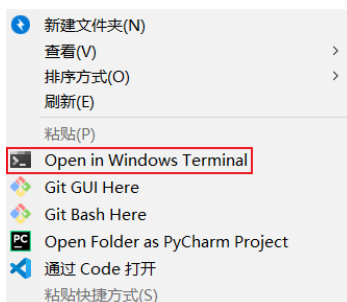
D:\test>
```

如果迟迟没有出现命令提示符，或者没有 PDF 文件，或者排版的效果非常奇怪，说明编写的代码有错误，需要进行排查。可以阅读当前目录下生成的`test1.log`日志文件，也可以直接阅读命令行窗口里的输出信息进行错误排查。由于报错信息都是英语，需要随时准备使用翻译软件或者搜索引擎。

这个例子展示了命令行的基本用法，通过输入磁盘的字母和`cd`指令进入需要运行命令的目录，再执行相应的命名，也有很多命令行命令可以直接运行而不需要切换目录。希望大家看到“命令行下运行 `xx`”这样的字眼时，能够不再害怕，而是上手操作。

### 1.2.3 Windows Terminal

使用命令行时每次都要手动切换目录未免让人感到厌烦，Windows 10 用户可以到微软商店（Microsoft Store）下载软件“Windows Terminal”。安装完成后重新启动电脑，你会发现右键菜单增加了选项“Open in Windows Terminal”。



此时，只要在你需要进入的路径下按右键选择“Open in Windows Terminal”，弹出来的命令行窗口的当前目录就是你进入的路径！但这种方式默认不是“以管理员身份运行”，因此，在一些情况下还是需要采用1.2.2节的方法。

### 1.2.4 重要的环境变量 Path

你可能会想，能不能用命令行运行其他程序比如 Word 呢？然而，当你输入`Word.exe`时，会得到如下的报错信息：`'Word.exe'` 不是内部或外部命令，也不是可运行的程序或批处理文件。

然而，你会发现，当你成功安装  $\text{\LaTeX}$  后，编译器的调用命令如`xelatex`，`pdflatex` 可以通过命令行在任意目录下使用；一些教程常常能看见这样的字眼“将 `xx` 添加到 Path/环境变量”使人摸不着头脑，其中的原因涉及到本节所讲的环境变量。本节内容主要参考这个视频：『教程』什么是环境变量。

### 1.3 编码

当你打开某些文件时，你可能发现眼前是完全无法阅读的一团乱码（比如用 TeXstudio 打开邓老师的旧模版），这里涉及到了编码的问题，下面的介绍主要参考了这个视频：『教程』文字频频乱码这背后是显卡的扭曲还是规则的沦丧？

人类语言有各种各样文字，承载了大量的信息，那么如何在计算机中储存文字呢，由于计算机内部只有 0 和 1，故问题转化为用数字表示文字，科学家通过建立数字和文字的一一映射，比如英语是由 26 个字母组成的

### 1.4 PDF 阅读器

PDF，全称为 Portable Document Format，意为“可携带文档格式”，是由 Adobe Systems 用于与应用程序、操作系统、硬件无关的方式进行文件交换所发展出的文件格式。这种格式最大的优点在于，不论在任何操作系统，手机还是电脑，最终的显示效果都是统一的，这就极大的方便了资料的传阅。L<sup>A</sup>T<sub>E</sub>X 代码在成功编译后，得到的便是 PDF 文件，下面介绍几个 PDF 阅读器，用于满足两个需求，一是在写论文时随时编译查看效果，二是查看最终效果。

#### 1.4.1 Adobe Acrobat Reader

Adobe Acrobat Reader 是由 PDF 格式的设计公司 Adobe 推出的一款免费的 PDF 阅读器（请与收费的 Pro 版区分！）。这个阅读器的显示效果最好，且支持 PDF 的所有功能（如 JavaScript 脚本、动画、3D 对象等），L<sup>A</sup>T<sub>E</sub>X 的一些高级功能的效果只有使用这个阅读器才能完全显示。并且可以查看 PDF 的许多属性比如使用的字体，在精细排版中会用到。

但有两个缺点，首先，安装时强制安在 C 盘，所以要记得腾空间。其次，**Adobe Acrobat Reader** 会锁定 **PDF**！在 **Windows** 中重复编译时，要先关闭已经由 **Adobe Reader** 打开的 **PDF** 文档，否则 **PDF** 文件会被锁定而不能更新（同时会报错），因此一般在写论文途中采用其他的 PDF 阅读器。

下载地址：<https://www.adobe.com/cn/acrobat/pdf-reader.html>

#### 1.4.2 SumatraPDF

SumatraPDF 是一个很小的开源 PDF 阅读器，具有免安装版，可以放在 U 盘随身携带，并且打开速度非常快，适合在写作途中查看文档效果，可以通过设置使得 PDF 可以反向搜索（定位到对应的代码）。当然，如果只是预览，Texwork 编译后会自动弹出预览窗口。

下载地址：<https://www.sumatrapdfreader.org/download-free-pdf-viewer> 或 <https://sourceforge.net/projects/sumatrapdf-reader.mirror/>



### 1.4.3 WPS

WPS 就不需要过多介绍了，与 Office 不同，它不强制安装在 C 盘。WPS 包括了文字 (Word)，表格 (Excel)，演示 (PPT) 和 PDF 四个功能，安装好之后就可以直接阅读 PDF 文件。顺带一提，WPS PDF 可以给没有书签的 PDF 文档手动加书签 (**不需要会员!**)，对于习惯使用电子书的同学会方便很多。

下载地址: <https://www.wps.cn/>

## 2 L<sup>A</sup>T<sub>E</sub>X 基础知识

### 2.1 安装相关

因为没有用过 Linux 系统，所以下面仅针对 Windows 系统和 Mac 系统展开，Linux 系统是类似的。

#### 2.1.1 安装 T<sub>E</sub>XLive

安装必读的中文官方文档：[install-latex-guide-zh-cn](#)

这篇文档已经千锤百炼，非常成熟了。其中 Windows 用户遇到的问题是最多的，Windows 用户一定要先把 **Windows** 系统的部分完整认真读完再去跟着安装，要心里有数，自己的电脑是个什么状态。

Mac 用户的就很简单了，pkg 包下载直接一键安装即可。

#### 2.1.2 安装外置 PDF 阅读器

其实不妨可以先试试编辑器自带的阅读器，如果觉得体验不好再下载也不迟。

前面介绍过 SumatraPDF，这是 Windows 用户推荐使用的，而 Mac 用户推荐使用的是 Skim 阅读器。

#### 2.1.3 安装 Visual Studio Code

虽然你安装完 T<sub>E</sub>XLive 后会有一个编辑器，但是个人是非常推荐使用 Visual Studio Code 的，有几个理由：

1. Visual Studio Code 打开速度快，比 T<sub>E</sub>Xstudio 是肉眼可见地快
2. 配置非常简单，下载一个插件，把别人弄好的

## 2.2 L<sup>A</sup>T<sub>E</sub>X 知识补充

### 2.2.1 表格

### 2.2.2 选择题选项排版