一份简短的关于 LaTeX 安装的介绍[1]

王然[2]

2026 年 11 月 1 日

[1] https://github.com/OsbertWang/install-latex-guide-zh-cn

[2] ranwang.osbert@outlook.com

前言

以“啸行”名义加入 QQ 群 91940767、478023327、640633524 和 200392395 后，经常有群友咨询如何安装 LaTeX. 实际上，用户安装的是 LaTeX 的**发行版**和相关的**编辑器**. 本手册将介绍在**不存在其他 LaTeX 发行版（如 CTeX 套装）的前提下**，在 Windows 11、Ubuntu 26.04 和 macOS 系统中安装 TeX Live (macOS 中介绍 MacTeX)、升级宏包、编译简易文档等相关操作，并多以介绍命令行操作为主. 有关 MiKTeX 的安装，可以参考 MiKTeX 的基本使用.

本手册还将简要介绍几款常见编辑器的使用方法，其他编辑器如 VS Code, Vim，用户可自行了解它们的使用方法. 鉴于 WSL 较为特殊，本手册提供了一份 VS Code 配合 LaTeX Workshop 的自用配置单，仅供参考.

除在本地安装 LaTeX 发行版和编辑器之外，本手册还额外介绍在线 LaTeX 平台的相关内容，如国际知名的 Overleaf 等.

本手册所涉及到的代码需结合上下文说明，不能简单地复制粘贴. 粉色文字都是可点的超链接，可直接跳转. 菜单 表示软件菜单. k 表示键盘按键. 建议用户阅读 texlive-zh-cn 和 lshort-zh-cn 以更全面地了解基础内容.

必须声明的是，即便你已经通读本手册和上面提到的两本书，有很大概率还会碰到很多问题. 这时我希望用户可以按照正确方式提问，例如提供**最小工作示例**，并在论坛上面留下你的问题和相应解答以方便后来者.

本手册大部分内容是个人过去一段时间的使用总结，其中难免有不甚合理或晦涩难懂的部分. 若用户在阅读本手册的过程中有任何意见和建议，请发邮件或在 GitHub 中提 issue. 本手册另在码云有备份，并于 2020 年 7 月提交至 CTAN.

本手册发布后，Dongsheng Deng, muzimuzhi, Xiangdong Zeng, tauyoung, myhsia, 对本手册提出了很好的建议，并提供了帮助，其中，有关 macOS 的内容最初由 Xiangdong Zeng 草拟完成，而后 Dongsheng Deng, tauyoung, myhsia 进行了补充，最近一次更新则由 tauyoung 提供. 在此一并感谢.

本手册自 2024 年 4 月开通捐赠渠道. 如果你认为本手册曾经帮助过你，并且你还愿意继续支持本手册，本着自愿原则，你可以扫描以下二维码为本手册捐赠固定金额. 需要说明的是，捐赠行为不会影响本手册的更新. 衷心感谢你的支持.




目录

1 Windows 11 系统 **5**

1.1 安装前准备 . . . 5

1.1.1 显示扩展名 . . . 5

1.1.2 检查环境 . . . 5

1.1.3 安全软件 . . . 5

1.1.4 下载与校验 . . . 6

1.2 安装 TeX Live . . . 6

1.2.1 下载镜像一直不成功的处理办法 . . . 9

1.2.2 需要管理员权限的处理办法 . . . 9

1.2.3 系统用中文作为用户名 . . . 9

1.3 升级宏包 . . . 9

1.3.1 `tlmgr` 本身无法成功升级 . . . 10

1.3.2 升级到一半停止 . . . 10

1.3.3 升级宏包的必要性 . . . 10

1.4 安装宏包 . . . 10

1.5 调出宏包手册 . . . 10

1.6 编译文档 . . . 11

1.6.1 LaTeX 2$_\varepsilon$ 版本不匹配导致 `xelatex` 失败 . . . 12

1.6.2 编译出现 `l3backend-*` 问题 . . . 12

1.7 卸载 TeX Live . . . 13

1.7.1 使用批处理文件 . . . 13

1.7.2 手动卸载 . . . 13

1.8 跨版本升级 TeX Live . . . 13

2 Ubuntu 26.04 系统 **14**

2.1 安装前准备 . . . 14

2.1.1 选择安装范围 . . . 14

2.1.2 更换 Ubuntu 源与确认 `fontconfig` . . . 14

2.1.3 卸载源内 TeX Live . . . 14

2.1.4 下载与校验 . . . 15

2.2 安装 TeX Live . . . 15

2.2.1 为所有用户安装 . . . 15
2.2.2 为个人用户安装 . . . 17
2.2.3 安装后的收尾 . . . 17
2.2.4 设置环境变量 . . . 17
2.2.5 配置字体缓存 . . . 18
2.2.6 验证安装 . . . 19
2.3 升级宏包 . . . 19
2.3.1 tlmgr 权限问题 . . . 19
2.3.2 利用 tlmgr 升级 . . . 20
2.4 安装宏包 . . . 20
2.5 调出宏包手册 . . . 21
2.6 编译文档 . . . 21
2.7 卸载 TeX Live . . . 22
2.7.1 手动卸载 . . . 22
2.7.2 使用 tlmgr 工具 . . . 22
2.8 跨版本升级 TeX Live . . . 22

3 macOS 23
3.1 安装 MacTeX . . . 23
3.1.1 直接下载 MacTeX 安装包进行安装 . . . 23
3.1.2 借助 Homebrew 命令行安装 . . . 23
3.2 宏包的管理与维护 . . . 24
3.3 编译文档 . . . 25
3.4 卸载 MacTeX . . . 25
3.5 跨版本升级 MacTeX . . . 25

4 Windows Subsystem for Linux 27
4.1 安装 TeX Live . . . 27
4.2 升级宏包 . . . 28
4.3 安装宏包 . . . 28
4.4 调出宏包手册 . . . 28
4.5 编译文档 . . . 29
4.6 卸载 TeX Live . . . 30
4.7 跨版本升级 TeX Live . . . 30

5 使用编辑器 31
5.1 Windows 11 系统 . . . 31
5.1.1 TeXworks . . . 31
5.1.2 TeXstudio . . . 33
5.2 Ubuntu 26.04 系统 . . . 37
5.2.1 使用源内安装 . . . 37

5.2.2 使用 AppImage . . . 37
5.2.3 额外设置路径 . . . 38
5.3 macOS . . . 38
5.3.1 TEXShop . . . 38
5.3.2 TEXstudio . . . 38
5.4 Windows Subsystem for Linux . . . 39
5.5 其他编辑器 . . . 39
5.6 多款编辑器对比参考 . . . 39

6 在线 LATEX 平台 41
6.1 Overleaf . . . 41
6.1.1 注册 Overleaf . . . 41
6.1.2 使用 Overleaf . . . 41
6.1.3 选择不同发行版版本 . . . 42
6.1.4 学习与帮助 . . . 42
6.2 国内的在线 LATEX 平台 . . . 42
6.2.1 TeXPage . . . 42
6.2.2 TeXHub . . . 42
6.2.3 LoongTeX . . . 43
6.2.4 Keepresearch . . . 43
6.2.5 FlyLaTeX . . . 43
6.3 科研院所搭建平台 . . . 43

A 大陆地区的源 44

B 有关 Ubuntu 和 WSL 的补充 46
B.1 更换 Ubuntu 26.04 源 . . . 46
B.2 安装字体 . . . 47
B.2.1 获取字体 . . . 47
B.2.2 使用 fontconfig . . . 48
B.2.3 使用 xfonts-utils . . . 48
B.2.4 直接安装源内字体 . . . 49
B.3 反向代理 PPA . . . 49
B.4 用 VS Code 作为 WSL 中的编辑器 . . . 50
B.5 迁移 WSL 的安装位置 . . . 52

C 离线安装宏包 54

D 版本信息 55

1 Windows 11 系统

Windows 系统是目前国内使用最广的操作系统，然而很多用户对这个常见的操作系统并不熟悉，尤其是在安装 TeX Live 时，常常会遇到很多意想不到的问题．本章先从安装前准备开始，分步写出安装 TeX Live 的过程．有些操作可能显得繁琐，但它确实可以极大概率规避安装过程中可能遇到的麻烦．

1.1 安装前准备

1.1.1 显示扩展名

在安装 TeX Live 之前，在文件夹菜单栏 查看 > 显示 中选择 文件扩展名．

1.1.2 检查环境

打开 cmd 窗口[1]，将会见到诸如以下内容

```
C:\Users\username>
```

它显示了当前所在路径并等待用户下一步操作，通常在开启 cmd 或上一程序运行结束时会显示．在 cmd 执行

```
echo %path:;=&echo.%
```

查看环境变量．若 `C:\Windows\system32` 不在结果中[2]，则关闭 cmd 窗口，将 `C:\Windows\system32` 添加到环境变量中[3]．**根据以往经验，安装老版 CTeX 套装的用户往往会有 `system32` 丢失的问题**．

另外，如果环境变量中有 `mingw`、`jdk`、`Cygwin` 相关的内容，也请暂时删除[4]．同时国内常见的 2345 好压也会对安装构成影响，建议删除它并在安装 TeX Live 后更换其他同类软件[5]．

1.1.3 安全软件

安装 TeX Live 最重要的是，**不安装第三方安全软件**．Windows 11 中已经内置了安全软件，用户尤其没有必要安装国产第三方杀毒软件，根据经验，它们会干扰 TeX Live 安装．

[1] 按 ⊞，或在任意文件夹中按 Ctrl+L，输入 cmd，点击 Enter

[2] 这里默认系统盘为 C 盘

[3] 在 桌面 > 此电脑 > 属性 > 高级系统设置 > 环境变量 > 系统变量 > Path 中添加

[4] 前两者具体原因参见 StackExchange 的解释，待安装结束后，再将环境变量重新添加至 TeX Live 的环境变量之后

[5] 推荐 WinRAR，7-Zip，NanaZip

1.1.4 下载与校验

用户可以从最近的 CTAN 源[6]下载 TeX Live 的 iso 镜像文件, 也可以找大陆地区的源自行下载[7]. 通过大陆地区的源下载 iso 文件的方法请参考附录 A.

假设镜像文件被下载到本地后, 其存放路径为 X:\Y[8].

安装前需要确保下载到的 iso 文件没有损坏, 一种常用做法是检查其 MD5 值. 打开 cmd 窗口依次执行以下代码

```
cd /d X:\Y
certutil -hashfile texlive2026.iso md5
```

若系统显示

```
MD5 的 texlive2026.iso 哈希:
f5872cb2dec838670f91ed5c62493553
CertUtil: -hashfile 命令成功完成.
```

则表明下载的镜像文件正常. 除 MD5 值外, 还可检查 SHA512 值. 依然是在 cmd 执行

```
cd /d X:\Y
certutil -hashfile texlive2026.iso sha512
```

若系统显示

```
SHA512 的 texlive2026.iso 哈希:
4a9071bb567c3bdd6443378dedc8e485aea4a2f1203ec8ed7c17f6787093b9c37636a037032c0be63352e3d0bf98cf5616d
    ab19fdcd7cb83f766b3e085b620ff
CertUtil: -hashfile 命令成功完成.
```

即为验证成功.

1.2 安装 TeX Live

接下来开始安装. 推荐用户先手动建立 TeX Live 的安装路径 (文件夹) 再进行安装, 注意路径的名称是**不带空格的英文**, 默认路径为 C:\texlive\2026\. 假如系统用了中文作为用户名, 那么用户需要先按照 1.2.3 节的方法更改临时环境变量后再安装. 将镜像文件装载至虚拟光驱[9]. 假设加载后的路径为 Z:\. 打开 cmd 窗口并执行

```
Z:\install-tl-windows.bat --no-gui
```

即可看到如下结果

```
======================> TeX Live installation procedure <======================

======> Letters/digits in <angle brackets> indicate <=======
```

[6]这里所谓的最近, 其实是由系统判定的, 实际上系统可能会误判

[7]新版本发布时, 各镜像网站同步的进度会有不同

[8]这里, X 表示盘符, 如 C、D、E 等; Y 表示文件夹; 默认路径名称为不带空格的英文

[9]Windows 11 默认双击镜像文件便可装载; 受某些压缩软件影响双击失效时, 可在文件夹菜单栏 装载; 第三方虚拟光驱软件可考虑 WinCDEmu 或 UltraISO; 不鼓励解压缩镜像文件

```
====== > menu items for actions or customizations <=======
= help> https://tug.org/texlive/doc/install-tl.html <=======

 Detected platform: Windows (64-bit)

 <B> set binary platforms: 1 out of 15

 <S> set installation scheme: scheme-full

 <C> set installation collections:
     41 collections out of 41, disk space required: 9615 MB (free: 203277 MB)

 <D> set directories:
   TEXDIR (the main TeX directory):
     C:/texlive/2026
   TEXMFLOCAL (directory for site-wide local files):
     C:/texlive/texmf-local
   TEXMFSYSVAR (directory for variable and automatically generated data):
     C:/texlive/2026/texmf-var
   TEXMFSYSCONFIG (directory for local config):
     C:/texlive/2026/texmf-config
   TEXMFVAR (personal directory for variable and automatically generated data):
     ~/.texlive2026/texmf-var
   TEXMFCONFIG (personal directory for local config):
     ~/.texlive2026/texmf-config
   TEXMFHOME (directory for user-specific files):
     ~/texmf

 <O> options:
   [ ] use letter size instead of A4 by default
   [X] allow execution of restricted list of programs via \write18
   [X] create all format files
   [X] install macro/font doc tree
   [X] install macro/font source tree
   [X] adjust search path
   [1] add menu items, shortcuts, etc.
   [1] update file associations
   [X] install TeXworks front end
   [X] after install, set CTAN as source for package updates

 <V> set up for portable installation

Actions:
 <I> start installation to hard disk
 <P> save installation profile to 'texlive.profile' and exit
 <Q> quit

```

```
Enter command:
```

这时一定要关注其中的

```
disk space required: 9615 MB (free: 203277 MB)
```

它代表了安装所需空间 9615 MB, 而目前空余空间 203277 MB. 根据每台电脑的实际情况, 数字可能不同, 只有空余空间大于安装所需空间时, 安装才能顺利. 确定空余空间大于安装所需空间后, 用户可直接按键盘 I 在默认路径中直接安装 TeX Live, 也可按键盘 D 来更改安装路径. 按 D 后, 用户可看到

```
================================================================================
Directories customization:

<1> TEXDIR: C:/texlive/2026
    main tree: C:/texlive/2026/texmf-dist

<2> TEXMFLOCAL: C:/texlive/texmf-local
<3> TEXMFSYSVAR: C:/texlive/2026/texmf-var
<4> TEXMFSYSCONFIG: C:/texlive/2026/texmf-config

<5> TEXMFVAR: ~/.texlive2026/texmf-var
<6> TEXMFCONFIG: ~/.texlive2026/texmf-config
<7> TEXMFHOME: ~/texmf

Note: ~ will expand to %USERPROFILE%

Actions:
<R> return to main menu
<Q> quit

Enter command:
```

这时按键盘 1, 将看到

```
New value for TEXDIR [C:/texlive/2026]:
```

这时可更改路径, 如 D:/texlive/2026. 接下来, 按键盘 R 即可回到初始的安装界面, 再按键盘 I 便可在刚才指定的路径安装. 用户可通过阅读英文执行其他操作, 这里不再赘述. 至此, 用户只需耐心等待安装完成, 并且不要点击 cmd 窗口[10]. 安装完成后, 关闭再打开 cmd 窗口[11], 执行

```
tex -v
```

显示

```
TeX 3.141592653 (TeX Live 2026)
kpathsea version 6.4.2
Copyright 2026 D.E. Knuth.
```

[10] Windows 11 默认采取快速编辑模式, 一旦点击 cmd 窗口, 将显示 “选择命令提示符” 而导致进程暂停

[11] 正常安装 TeX Live 后, 环境变量会变化, 关闭原 cmd 窗口再重新打开才能获取新的环境变量

```
There is NO warranty. Redistribution of this software is
covered by the terms of both the TeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the TeX source.
Primary author of TeX: D.E. Knuth.
```

即可认为安装顺利完成[12]. 弹出前面装载的光盘镜像, 安装到此结束.

1.2.1 下载镜像一直不成功的处理办法

目前发现有些用户使用某些浏览器时无法正常下载镜像文件. 这时, 用户可以选择使用 wget 对镜像进行下载. 具体用法是: 将 wget 所在目录添加到环境变量中. 打开 cmd 窗口执行以下代码

```
wget http://mirrors.ctan.org/systems/texlive/Images/texlive2026.iso
```

即可下载最近的镜像. 类似地, 也可以指定国内的源进行下载, 例如选择清华大学镜像

```
wget https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/texlive2026.iso
```

其他国内镜像只需要将链接替换即可, 链接和方法请参考附录 A.

1.2.2 需要管理员权限的处理办法

有些用户反馈安装过程中需要管理员权限. 以管理员身份打开 cmd 窗口的方法是点击 ⊞, 输入 cmd, 按 Ctrl+⇧+Enter 三个键. 一般而言, 在系统盘内直接安装 TeX Live 会碰到此类问题, 因此鼓励用户在安装前先手动建立安装路径. 直接使用管理员身份打开 cmd 窗口进行安装可能会对以后的升级造成影响.

1.2.3 系统用中文作为用户名

很多中文 Windows 用户习惯于用中文作为用户名, 但是这样会导致 TeX Live 安装失败. 更改系统的 TEMP 和 TMP 这两个环境变量可以避免这样的问题. 在 cmd 中执行

```
mkdir C:\temp
set TEMP=C:\temp
set TMP=C:\temp
```

便可更改这两个环境变量为 C:\temp. 更改后按照前述内容安装 TeX Live 即可. 这样的更改是临时的, 待 cmd 窗口关闭, 环境变量将恢复为原始状态.

1.3 升级宏包

安装完成后, 用户可以升级宏包以获得更好的使用体验. 下面将介绍使用命令行升级宏包的方法. 打开 cmd 窗口, 首先执行下面命令指定升级使用的镜像源. ctan 表示系统在升级时将自动寻求最近的源进行下载.

[12]若未能正确显示结果, 则可能是因为环境变量未改变, 用户可手动添加 <TeXPath>\bin\windows 至环境变量, <TeXPath> 需与你方才的安装路径一致

```
tlmgr option repository ctan
```

用户同样可以指定其他的镜像源, 方法见附录 A.

接下来, 用户执行命令

```
tlmgr update --list
```

可查看目前源上可升级的宏包都有哪些. 高级用户可以根据自己的需求选择升级特定宏包. 建议初级用户直接升级全部宏包. 用户只需执行

```
tlmgr update --self --all
```

同时升级 tlmgr 本身和全部宏包.

1.3.1 tlmgr 本身无法成功升级

遇到这种情况时, 用户需自行下载并运行 update-tlmgr-latest.exe, 然后再执行升级命令即可[13].

1.3.2 升级到一半停止

这种情况下, 用户需要执行另外一个命令继续升级

```
tlmgr update --reinstall-forcibly-removed --all
```

1.3.3 升级宏包的必要性

自本手册撰写以来, 每年镜像文件打包时都可能存在一些问题, 具体讨论见这里. 用户安装完毕后, 请按照本节提供的方法升级.

1.4 安装宏包

在默认状态下, 用户将完整安装 TEX Live, 因此用户极少碰到需要手动安装宏包的情形. 同时, 在 lshort-zh-cn 中也明确提到, **如非万不得已, 尽量不要手动安装宏包**. 因此在这里我只介绍从源处安装宏包的命令. 假设用户想安装 mcmthesis 宏包, 只需在 cmd 执行

```
tlmgr install mcmthesis
```

需要注意的是, 用户一定要清楚所要安装的宏包名称, 并且在安装宏包前先确保镜像源设置正确.

1.5 调出宏包手册

当正确安装后, 用户可以调出宏包手册以查阅相应内容. 例如在 cmd 执行

```
texdoc texlive-zh-cn
```

[13] 这里依旧让系统自动选择镜像源, 用户可以自行选择国内镜像以加快下载速度

或

```
texdoc lshort-zh-cn
```

就可分别调出 texlive-zh-cn.pdf 和 lshort-zh-cn.pdf, 这两本手册中都有讲解安装的相关内容, 建议用户阅读.

本手册业已被 TEX Live 收录, 调出本手册的命令为

```
texdoc install-latex-guide-zh-cn
```

1.6 编译文档

升级宏包完成后, 用户可以编译文档. 这里依旧使用命令行来完成这一过程.

首先, 用户需要在指定位置[14]建立一个 tex 文件:

```
mkdir D:\work-latex
cd /d D:\work-latex
notepad main.tex
```

第 1 行表示创建一个工作路径 D:\work-latex, 第 2 行表示进入工作路径, 第 3 行表示用记事本打开 main.tex 文件, 若文件不存在, 系统将询问用户是否创建该文件[15]. 在打开的记事本中编写一个最小示例

```
\documentclass{article}
\begin{document}
  Hello \LaTeX\ World!
\end{document}
```

保存并退出. 接下来执行

```
pdflatex main
```

等待系统完成编译过程. 待编译完成后, 我们即可看到在 D:\work-latex\ 中出现了 main.pdf 文件和其他同名的辅助文件 main.aux 与 main.log.

对于中文文档, 可以在记事本中编写以下最小示例[16]

```
\documentclass[fontset = windows]{ctexart}
\begin{document}
  你好 \LaTeX\ 世界!
\end{document}
```

保存并退出. 接下来执行

```
xelatex main
```

[14] 以下称工作路径

[15] Windows 11 对中文名变得友好一些了, 但这里不建议用户使用中文命名工作路径和文件名

[16] 自 Windows 10 1903 版本开始, 记事本默认使用 UTF-8 编码; Windows 11 继承了这个特性. 若使用其他编辑器, 请注意规定文档编码为 UTF-8, 这与 CTEX 宏集默认使用 UTF8 选项有关

等待系统完成编译过程. xelatex 编译命令配合 UTF-8 编码和 CTEX 宏集[17] 已经是**目前主流的中文处理手段**. xelatex 可以使用系统内安装的字体, 用户在安装字体时需右键字体文件的快捷菜单 显示更多选项 〉 为所有用户安装.

总结一下, 在使用 LATEX 时, 用户首先要在工作路径[18]中建立一个完整的 tex 文件[19]. 当 tex 文件内容确定后, 再保存文件并使用编译命令[20], 将 tex 文件编译成 pdf 文件.

编译命令有很多可选参数, 如用户能够从 pdf 文件跳回 tex 文件, 便是因为执行编译时添加了参数

```
pdflatex -synctex=1 main
```

有些编译还需要额外调用系统命令, 例如使用 svg 包就有这类要求, 因此可以在执行编译时添加参数

```
pdflatex -shell-escape main
```

此外还有很多其他参数, 用户若感兴趣, 可自行阅读手册.

在很多时候, 我个人更倾向于使用 latexmk 来编译 tex 文档, 如执行

```
latexmk -pdf -synctex=1 -interaction=nonstopmode main
```

意味着使用

```
pdflatex -synctex=1 -interaction=nonstopmode main
```

来编译文档 main.tex, 并在有需要时完成其他步骤[21].

1.6.1 LATEX 2ε 版本不匹配导致 xelatex 失败

使用 xelatex 编译文档, 系统有时会提示目前 LATEX 2ε 的版本低于某宏包的要求版本. 这种情况是由于 fmt 文件未能更新所导致. 此时可考虑在命令行运行

```
fmtutil-user --byfmt xelatex
```

待结束后再用 xelatex 编译文档. 更多内容参考 GitHub 上的讨论.

另外也可考虑重装 xetex. 在命令行执行

```
tlmgr install --reinstall xetex
```

1.6.2 编译出现 l3backend-* 问题

这是 fmt 文件未能更新所导致的问题. 可以用

```
fmtutil-sys --all
```

刷新 fmt 文件后再编译 tex 文件.

[17] CTEX 宏集与发行版 CTEX 套装并非是同一事物

[18] 建议不同的工程放入不同的工作路径

[19] 该文件必须包含\ documentclass, \begin{document} 和\ end{document}, 这里不讨论包含子文件的情况, 文件名需为不含空格的英文

[20] 目前常用的编译命令为 pdflatex 和 xelatex

[21] 如使用 bibtex 或 biblatex 处理参考文献时需要多次编译, 详情见相关文档

1.7 卸载 TEX Live

在跨版本升级 TEX Live 时, 通常需要卸载旧版 TEX Live. 下面介绍几种卸载方法.

1.7.1 使用批处理文件

在命令行执行

```
kpsewhich -var-value TEXMFROOT
```

可以查看 TEXMFROOT 的值, 该值即为 TEX Live 的安装路径[22]. 接下来, 用户只需执行[23]安装路径中的卸载批处理文件即可实现卸载, 如默认安装时执行

```
C:\texlive\2026\tlpkg\installer\uninst.bat
```

结束以上步骤后, 手动删除用户文件夹中的.texlive2026. 如果用户安装 TEX Live 时需要管理员权限, 那么在卸载时同样需要管理员权限.

卸载后, 用户需要清理注册表

```
HKEY_CURRENT_USER\Software\Classes\TL.DVIOUT.view.2026
HKEY_CURRENT_USER\Software\Classes\TL.TeXworks.edit.2026
HKEY_CURRENT_USER\Software\Classes\TL.PSViewer.view.2026
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ApplicationAssociationToasts\TL.
    TeXworks.edit.2026_.tex
```

但若在安装时使用管理员权限 “为所有人安装”, 则注册表位置将变为

```
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\TL.DVIOUT.view.2026
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\TL.TeXworks.edit.2026
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\TL.PSViewer.view.2026
```

1.7.2 手动卸载

如果执行批处理文件 uninst.bat 出错, 用户也可手动删除安装文件夹, 之后再清理 TEX Live 的环境变量, 并且清理注册表.

1.8 跨版本升级 TEX Live

目前在 tug.org 上面提供的说法是: Windows 没有类似 Unix 的升级程序, 需要进行新安装[24].

[22] 除这里提供的 TEXMFROOT, 还有很多其他变量, 可在 texlive-zh-cn 第 2.3 节处找到

[23] 双击或在 cmd 中运行

[24] 原文是: There is no comparable upgrade procedure for Windows. Doing a new installation is necessary

2 Ubuntu 26.04 系统

本章介绍如何在 Ubuntu 26.04 上使用镜像安装 TEX Live 2026. 鉴于 Ubuntu 的特殊性, 建议用户在安装前先完成若干准备工作. 以下命令未特别说明均在 Terminal 中执行.

2.1 安装前准备

2.1.1 选择安装范围

TEX Live 支持两种安装范围:

为所有用户安装 默认安装到 `/usr/local/texlive/2026`, 需要 `sudo` 权限;

为个人用户安装 安装到用户主目录下, 例如 `~/.local/texlive/2026`, 无需 `sudo` 权限.

这两种方式的安装命令略有不同, 后续配置环境变量和字体时也有区别.

2.1.2 更换 Ubuntu 源与确认 `fontconfig`

鉴于大陆地区的网络环境, 建议在执行命令前更换 Ubuntu 源, 参考 B.1 节.

由于后续内容涉及字体, 建议用户先确定系统内是否已经有 `fontconfig`, 经测试, Ubuntu 26.04 默认安装 `fontconfig`. 如需安装 `fontconfig` 只需执行

```
sudo apt install fontconfig
```

2.1.3 卸载源内 TEX Live

为保证用户顺利使用 TEX Live 2026, 建议首先卸载从源内安装的 TEX Live 的相关包. 个人比较推荐使用 `synaptic package manager`. 执行

```
sudo apt install synaptic
```

即可安装. 安装后打开, 搜索 texlive 即可看到与之相关的包, 右键标记以删除即可. 当然, 用户也可以直接使用

```
dpkg -l | grep '^ii' | grep -i -E 'texlive|tex live'
```

找到相应的包名 `<package name>`, 之后使用类似

```
sudo apt autoremove --purge <package name>
```

删除各个包. 通常

```
sudo apt autoremove --purge tex-common
```

就可以删除源内版本.

2.1.4 下载与校验

下载 iso 镜像文件的方法见第 1.1.4 节. 下载完毕后, 验证 MD5 值, 即执行以下命令

```
cd ~/Downloads
md5sum texlive2026.iso
```

若显示

```
f5872cb2dec838670f91ed5c62493553 texlive2026.iso
```

则镜像文件下载正确. 同样可验证 SHA512 值, 执行

```
cd ~/Downloads
sha512sum texlive2026.iso
```

正确的返回结果为

```
4a9071bb567c3bdd6443378dedc8e485aea4a2f1203ec8ed7c17f6787093b9c37636a037032c0be63352e3d0bf98cf5616d
    ab19fdcd7cb83f766b3e085b620ff texlive2026.iso
```

2.2 安装 TEX Live

将下载的镜像文件进行挂载[1],

```
sudo mkdir /mnt/texlive
sudo mount ~/Downloads/texlive2026.iso /mnt/texlive
```

2.2.1 为所有用户安装

执行

```
sudo /mnt/texlive/install-tl
```

进行安装. 在屏幕上应该能见到以下安装界面

```
======================> TeX Live installation procedure <======================

======> Letters/digits in <angle brackets> indicate <=======
======> menu items for actions or customizations <=======
= help> https://tug.org/texlive/doc/install-tl.html <=======
```

[1]右键快捷菜单会将镜像直接挂载到 /run/media/username/texlive2026, 省去了手动 mount 的过程; 记住系统提供的挂载路径即可直接执行安装过程

```
 Detected platform: GNU/Linux on x86_64

 <B> set binary platforms: 1 out of 15

 <S> set installation scheme: scheme-full

 <C> set installation collections:
    40 collections out of 41, disk space required: 9388 MB (free: 974100 MB)

 <D> set directories:
   TEXDIR (the main TeX directory):
     /usr/local/texlive/2026
   TEXMFLOCAL (directory for site-wide local files):
     /usr/local/texlive/texmf-local
   TEXMFSYSVAR (directory for variable and automatically generated data):
     /usr/local/texlive/2026/texmf-var
   TEXMFSYSCONFIG (directory for local config):
     /usr/local/texlive/2026/texmf-config
   TEXMFVAR (personal directory for variable and automatically generated data):
     ~/.texlive2026/texmf-var
   TEXMFCONFIG (personal directory for local config):
     ~/.texlive2026/texmf-config
   TEXMFHOME (directory for user-specific files):
     ~/texmf

 <O> options:
   [ ] use letter size instead of A4 by default
   [X] allow execution of restricted list of programs via \write18
   [X] create all format files
   [X] install macro/font doc tree
   [X] install macro/font source tree
   [ ] create symlinks to standard directories
   [X] after install, set CTAN as source for package updates

 <V> set up for portable installation

Actions:
 <I> start installation to hard disk
 <P> save installation profile to 'texlive.profile' and exit
 <Q> quit

Enter command:
```

这时一定要关注其中的

```
disk space required: 9388 MB (free: 974100 MB)
```

它代表了安装所需空间 9388 MB, 而目前空余空间 974100 MB. 根据每台电脑的实际情况, 数字可能不同, 只有空余空间大于安装所需空间时, 安装才能顺利.

点击键盘 I 进行默认安装, 即是为所有用户安装. 如果用户对于 Ubuntu 系统比较了解, 可以根据提示, 更改安装设置, 如安装路径等. 为简化起见, 以下记录均假定为 x86_64 上的默认安装.

2.2.2 为个人用户安装

为个人用户安装 TEX Live 不调用权限, 直接执行

```
/mnt/texlive/install-tl
```

只不过这时安装路径需要选择在个人用户文件夹下. 在安装界面点击键盘 D, 更改安装路径, 例如

```
~/.local/texlive/2026/
```

然后点击键盘 R 返回, I 安装.

2.2.3 安装后的收尾

安装完毕后, 系统会提示用户添加环境变量, 例如

```
export PATH=/usr/local/texlive/2026/bin/x86_64-linux:$PATH
export MANPATH=/usr/local/texlive/2026/texmf-dist/doc/man:$MANPATH
export INFOPATH=/usr/local/texlive/2026/texmf-dist/doc/info:$INFOPATH
```

此处显示信息可能会略有不同, 比如电脑使用了基于 arm64 架构的处理器, 则 x86_64-linux 将被替换为 aarch64-linux. 为安全起见, 用户需将此处的系统提示记录.

确认安装完成后, 将挂载的光盘镜像弹出并删除文件夹, 即执行

```
sudo umount /mnt/texlive
sudo rm -r /mnt/texlive
```

2.2.4 设置环境变量

这里列出两种在 Ubuntu 26.04 设置 TEX Live 的环境变量的方法. 以下命令默认针对为所有用户安装, 若为个人用户安装, 直接去掉 sudo, 并将路径替换为 ~/.local/texlive/2026/.

一步法

这种方法只需要一条命令, 不仅能设置环境变量, 还能免去后续使用 tlmgr 需要额外设置权限的麻烦. 直接执行

```
sudo /usr/local/texlive/2026/bin/x86_64-linux/tlmgr path add
```

本质上, 它在 /usr/local/bin 中创建了一系列符号链接, 这些链接指向当前 TEX Live 的二进制目录. **为个人用户安装禁止使用本方法**.

常规法

这种方法适用性更广, 可以为其他软件的环境变量设置提供参考. 打开设置文件

```
gnome-text-editor ~/.profile
```

在文件末尾添加刚刚记录的系统提示

```
# Add TeX Live to the PATH, MANPATH, INFOPATH
export PATH=/usr/local/texlive/2026/bin/x86_64-linux:$PATH
export MANPATH=/usr/local/texlive/2026/texmf-dist/doc/man:$MANPATH
export INFOPATH=/usr/local/texlive/2026/texmf-dist/doc/info:$INFOPATH
```

保存退出. 如果用户更改了安装路径, 要注意将

```
/usr/local/texlive/2026/
```

在各处替换.

若**为个人用户**安装, 则相应改为

```
# Add TeX Live to the PATH, MANPATH, INFOPATH
export PATH=~/.local/texlive/2026/bin/x86_64-linux:$PATH
export MANPATH=~/.local/texlive/2026/texmf-dist/doc/man:$MANPATH
export INFOPATH=~/.local/texlive/2026/texmf-dist/doc/info:$INFOPATH
```

设置好环境变量后, 退出当前用户并重新登录.

2.2.5 配置字体缓存

接下来处理字体. 执行

```
sudo cp /usr/local/texlive/2026/texmf-var/fonts/conf/texlive-fontconfig.conf /etc/fonts/conf.d/09
    -texlive.conf
```

将配置文件复制到系统, 然后继续执行

```
sudo fc-cache -fsv
```

刷新字体缓存. 这样一来, TEX Live 中的字体才能够被正确调用. 更多内容请参考 texlive-zh-cn.

为个人用户安装时, 有关字体的处理变为[2]

```
mkdir -p "${XDG_CONFIG_HOME:-$HOME/.config}/fontconfig/conf.d"
cp ~/.local/texlive/2026/texmf-var/fonts/conf/texlive-fontconfig.conf "${XDG_CONFIG_HOME:-$HOME/.
    config}/fontconfig/conf.d/09-texlive.conf"
```

然后执行

```
fc-cache -fv
```

实际上, 无论将 TEX Live 安装到哪里, 字体的配置文件都可以仿照上面只复制于个人用户文件夹中.

[2]这里并未使用 texlive-zh-cn 中建议的路径 `~/.fonts.conf.d/`, 因为根据 freedesktop 的说法, 该路径已废弃, 故此处采用新的路径, 即 `$XDG_CONFIG_HOME/fontconfig/conf.d/`

2.2.6 验证安装

执行

```
tex -v
```

若显示

```
TeX 3.141592653 (TeX Live 2026)
kpathsea version 6.4.2
Copyright 2026 D.E. Knuth.
There is NO warranty. Redistribution of this software is
covered by the terms of both the TeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the TeX source.
Primary author of TeX: D.E. Knuth.
```

即认为安装成功.

2.3 升级宏包

2.3.1 tlmgr 权限问题

默认安装情况下, 由于权限问题, 直接使用 tlmgr 升级宏包会失败, 但如果使用 sudo 就牵扯到 tlmgr 命令能否在 sudo 下执行的问题. 至于为个人用户安装, 下述的 tlmgr 命令可以脱离 sudo 使用. 后续内容类似, 不再赘述.

若采用一**步法**设置环境变量, 则已经允许用户在 sudo 下直接使用 tlmgr. 例如执行

```
sudo tlmgr option repository ctan
```

让系统自动选择源, 用户也可以选择大陆地区的源进行升级, 详情见附录 A.

若使用**常规法**设置环境变量, 则会因为 sudo 默认没有加载 tlmgr 为环境变量而导致失效, 下面提供几种方法解决该问题.

使用绝对路径

此方法通过调用 which 命令来查找 tlmgr 的绝对路径, 例如让系统自动选择源

```
sudo $(which tlmgr) option repository ctan
```

保持环境变量

由于前期已经配置好了 PATH, 因此可以直接用

```
sudo env PATH="$PATH" tlmgr option repository ctan
```

修改目录权限

首先更改安装路径所有权为当前用户

```
sudo chown -R $USER:$USER /usr/local/texlive/2026
```

然后就可直接脱离 sudo 运行

```
tlmgr option repository ctan
```

如果想恢复权限, 执行

```
sudo chown -R root:root /usr/local/texlive/2026
sudo chmod -R 755 /usr/local/texlive/2026
```

添加系统权限

此种方法是我在个人电脑上最常用的方法, 但因为它有可能造成安全漏洞, 因此**不建议新用户使用**. 执行

```
sudo visudo
```

用 nano 打开文件, 在 secure_path 后面添加

```
/usr/local/texlive/2026/bin/x86_64-linux:
```

按 Ctrl+O, Enter 保存, Ctrl+X 退出. 接下来的用法与一**步法**一致, 不再赘述.

2.3.2 利用 tlmgr 升级

执行命令

```
sudo tlmgr update --list
```

可查看目前源上可升级的宏包都有哪些. 高级用户可以根据自己的需求选择升级特定宏包. 建议初级用户执行

```
sudo tlmgr update --self --all
```

同时升级 tlmgr 本身和全部宏包. 使用**常规法**时, 还要根据不同的 tlmgr 权限使用对应命令.

若**为个人用户安装**, 请去掉 sudo.

2.4 安装宏包

在 Ubuntu 中安装宏包的要求与 Windows 11 没有多少区别, 依旧使用 tlmgr, 只需注意权限, 例如

```
sudo tlmgr install mcmthesis
```

即安装了 mcmthesis 宏包. 若为个人用户安装, 去掉 sudo.

2.5 调出宏包手册

与 Windows 11 类似, 当正确安装后, 用户可以调出宏包手册以查阅相应内容. 例如执行

```
texdoc texlive-zh-cn
texdoc lshort-zh-cn
texdoc install-latex-guide-zh-cn
```

就可分别调出 texlive-zh-cn.pdf, lshort-zh-cn.pdf 或本手册.

2.6 编译文档

首先, 用户需要在工作路径建立一个 tex 文件. 执行

```
mkdir ~/Documents/work-latex
cd ~/Documents/work-latex/
gnome-text-editor main.tex
```

在打开的文件输入一个最小示例

```
\documentclass{article}
\begin{document}
  Hello \LaTeX\ World!
\end{document}
```

保存并退出. 接下来执行

```
pdflatex main
```

等待系统完成编译过程. 待编译完成后, 我们即可看到在 ~/Documents/work-latex/ 中出现了 main.pdf 文件和其他同名的辅助文件 main.aux 与 main.log. 执行

```
papers main.pdf
```

即可打开 pdf 文件.

对于中文文档, 可以在 gnome-text-editor 中编写以下最小示例[3]

```
\documentclass{ctexart}
\begin{document}
  你好 \LaTeX\ 世界!
\end{document}
```

保存并退出. 接下来执行

```
xelatex main
```

等待系统完成编译过程. xelatex 可以使用系统内安装的字体, 安装字体的方法可参考 B.2 节.

编译命令可添加参数, 这里与 1.6 节中的情形一致.

[3] gnome-text-editor 默认使用 UTF-8 编码

2.7 卸载 TeX Live

下面提供几种卸载方法, 卸载后需注意清理环境变量.

2.7.1 手动卸载

从光盘镜像安装, 直接删除文件夹即可. 先执行

```
kpsewhich -var-value TEXMFROOT
```

来查询安装路径, 进而通过 `sudo rm -rf` 命令进行删除. 默认安装的用户直接运行

```
sudo rm -rf /usr/local/texlive/2026
rm -rf ~/.texlive2026
```

卸载完成后, 可以进一步移除之前设置的环境变量. 若用一步法设置环境变量, 只需执行

```
sudo tlmgr path remove
```

若用常规发设置环境变量, 执行

```
gnome-text-editor ~/.profile
```

然后移除之前在文件末尾添加的

```
# Add TeX Live to the PATH, MANPATH, INFOPATH
export PATH=/usr/local/texlive/2026/bin/x86_64-linux:$PATH
export MANPATH=/usr/local/texlive/2026/texmf-dist/doc/man:$MANPATH
export INFOPATH=/usr/local/texlive/2026/texmf-dist/doc/info:$INFOPATH
```

并保存退出. 再执行

```
sudo visudo
```

在 `secure_path` 中删除

```
/usr/local/texlive/2026/bin/x86_64-linux:
```

依次按 Ctrl+O, Enter, Ctrl+X 保存退出.

2.7.2 使用 tlmgr 工具

同样, 用户也可以使用 `tlmgr` 工具卸载[4],

```
sudo tlmgr remove --all
```

2.8 跨版本升级 TeX Live

在 tug.org 网站上提供了相应的方法. 但网站也声明: 默认情况下, 请通过执行新安装来获取新版本的 TeX Live[5].

[4] `sudo tlmgr` 无法使用的情况可以参考第 2.3.1 节解决

[5] 原文是: By default, please get the new TL by doing a new installation instead of proceeding here

3 macOS

3.1 安装 MacTEX

MacTEX 是 TEX Live 针对 macOS 环境的再打包版本, 并额外集成了若干辅助程序. 在 macOS 上安装该发行版主要有以下两种方式:

直接下载安装包 手动下载安装镜像并在系统内执行安装程序.

借助 Homebrew 命令行安装 先部署 Homebrew 包管理器, 随后通过终端执行在线安装.

安装建议: 当前阶段, 建议用户优先选择直接下载安装包的方式. 由于中国大陆用户访问 GitHub 的稳定性较差, 使用 Homebrew 脚本安装时可能会因网络波动导致失败.

3.1.1 直接下载 MacTEX 安装包进行安装

下载 `MacTeX.pkg` 文件后, 双击运行并按照系统弹出的安装向导指引完成操作. 其流程与安装 macOS 上的常规软件一致. 用户自主选择大陆各个镜像下载 `MacTeX.pkg` 的方法请参考附录 A.

3.1.2 借助 Homebrew 命令行安装

Homebrew 是 macOS 上广泛使用的包管理工具, 功能类似于 Ubuntu 的 `apt-get` .

安装 Homebrew

在**终端**[1]执行以下命令[2] :

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

执行后, 脚本会说明即将进行的改动并暂停, 用户需根据屏幕指令确认执行.

中国大陆用户可以使用镜像以提高访问速度. 以清华大学镜像源为例[3]

```
export HOMEBREW_BREW_GIT_REMOTE="https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/brew.git"
export HOMEBREW_CORE_GIT_REMOTE="https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-core.git"
```

[1]打开方法为: ⌘+Space, 输入 terminal 并打开 终端 应用

[2]参见 Homebrew 官方文档, Homebrew 支持最新三个 macOS 年度版本

[3]此处涉及的环境变量参见 Homebrew 官方文档, 各镜像站的安装说明参见 Homebrew 软件仓库镜像使用帮助, Homebrew Bottles 软件仓库镜像使用帮助

```
echo 'export HOMEBREW_API_DOMAIN="https://mirrors.cernet.edu.cn/homebrew-bottles/api"' >> ~/.zprofile
echo 'export HOMEBREW_BOTTLE_DOMAIN="https://mirrors.cernet.edu.cn/homebrew-bottles"' >> ~/.zprofile
export HOMEBREW_API_DOMAIN="https://mirrors.cernet.edu.cn/homebrew-bottles/api"
export HOMEBREW_BOTTLE_DOMAIN="https://mirrors.cernet.edu.cn/homebrew-bottles"
```

若使用 Bash, 请将上述配置中的 `.zprofile` 替换为 `.bash_profile`.

用户亦可使用 CunKai 在码云平台提供的自动化脚本, 该脚本针对国内网络环境进行了优化并简化了流程:

```
/bin/zsh -c "$(curl -fsSL https://gitee.com/cunkai/HomebrewCN/raw/master/Homebrew.sh)"
```

安装 MacTEX

安装完 Homebrew 后, 在**终端**执行相应命令即可完成部署:

```
brew install mactex
```

如果用户不需要 MacTEX 的附加组件 (图形化应用程序), 也可以考虑仅安装其命令行工具, 通过在终端键入命令或者从其他文本编辑器调用.

```
brew install mactex-no-gui
```

完整的 MacTEX 会占据大量空间. 如果磁盘空间实在紧张, 也可以考虑安装轻量化的 BasicTEX:

```
brew install basictex
```

BasicTEX 仅包含最基础的组件和宏包, 手动安装会比较麻烦, 所以不推荐没有经验的用户使用.

注意, BasicTEX 默认不安装宏包文档和源码. 如有需要, 可以执行

```
# 1 为安装, 0 为不安装
sudo tlmgr option docfiles 1 # 安装文档, 推荐
sudo tlmgr option srcfiles 1 # 安装源码, 可选
```

配置 `tlmgr`. 在配置修改时已经安装的包, 需要重新安装才能应用新配置, 例如可以执行

```
sudo tlmgr install --reinstall \
  $(tlmgr list --only-installed | sed -e 's/^i \([^:]*\): .*$/\1/')
```

3.2 宏包的管理与维护

升级和安装宏包通过终端工具 `tlmgr` 进行, 操作时通常需要 `sudo` 管理员权限.

升级宏包 参见 2.3 节.

安装宏包 参见 2.4 节.

查阅手册 参见 2.5 节.

3.3 编译文档

使用文本编辑器 (如 TextEdit.app) 编写 `main.tex`:

```
\documentclass{article}
\begin{document}
  Hello \LaTeX\ World!
\end{document}
```

在**终端**执行编译命令

```
pdflatex main
```

编译完成后, 工作路径内将生成 `main.pdf` 结果文件, 以及 `main.aux` 和 `main.log` 辅助文件.

对于中文文档, 可以编写以下最小示例

```
\documentclass{ctexart}
\begin{document}
  你好 \LaTeX\ 世界!
\end{document}
```

在**终端**执行

```
xelatex main
```

`xelatex` 可调用系统字体, 为系统安装字体的方法请参考在 Mac 上的字体册中安装和验证字体. 安装完成后, 需要刷新字体缓存. `xelatex` 会调用系统的 CoreText 库来管理字体, 因此可以访问 `~/Library/Fonts/` 目录下的字体文件.

注意默认状态下 `xelatex` 只能通过文件名来调用发行版预装的字体, 在**终端**执行下面两步即可用字体名调用发行版预装的字体

```
ln -s /usr/local/texlive/2026/texmf-dist/fonts/opentype ~/Library/Fonts/
ln -s /usr/local/texlive/2026/texmf-dist/fonts/truetype ~/Library/Fonts/
```

3.4 卸载 MacTEX

如果用户直接下载了 MacTEX 安装包进行安装, 可以对照这里的介绍来卸载, 通常我个人会比较建议在跨版本升级前卸载旧版本.

如果用户借助 Homebrew 安装了 MacTEX, 亦可以通过 Homebrew 卸载. [4]

```
brew uninstall --zap mactex (或 mactex-no-gui)
```

3.5 跨版本升级 MacTEX

如果用户直接下载 MacTEX 安装包进行安装, 新版本安装器不会覆盖旧版本, 需要自行卸载旧版本, 或选择同时保留多个版本, 可以参见这里.

[4] `--zap` 选项会额外删除 `/usr/local/texlive/texmf-local` 和 GUI 应用相关的文件, 以实现无残留卸载

如果用户借助 Homebrew 安装了 MacTEX, 跨版本升级 (MacTEX 的版本与 TEX Live 保持一致) 可在**终端**借助 Homebrew 完成:

```
brew update
brew upgrade mactex (或 mactex-no-gui)
```

4 Windows Subsystem for Linux

目前微软推出了 Windows Subsystem for Linux (WSL) 供开发人员使用. 在这里简要介绍如何在 WSL 中安装 TeX Live. 我选择的是 Windows 11 微软商店中的 Ubuntu. 此篇安装教程仅供参考. 这里称 WSL 中使用的命令行为 bash. **不建议**对操作系统了解不多的用户阅读本章内容.

4.1 安装 TeX Live

在 Windows 11 中, WSL 已经能够自动完成可视化工作, 这为用户提供了一些便利. 但更多的安装操作涉及 apt 操作. 为避免安装速度过慢, 可仿照第 B.1 节更改文件 sources.list. 不习惯在 WSL 中做文本操作的 Windows 用户, 可参考第 B.4 节. 其余安装前的准备工作见第 2.1 节.

在主系统[1]中下载 iso 镜像文件, 可选择国内源以加快下载速度. 下载完毕后, 在 cmd 中验证文件是否正常. 下载及文件验证方法同第 1.1.4 节.

接下来挂载镜像. 这里需要先通过一个例子解释 WSL 的路径和主系统路径的对应关系. 例如, iso 镜像文件在主系统中的路径是 C:\Users\username\Downloads\texlive2026.iso, 那么在 WSL 中其路径是 /mnt/c/Users/username/Downloads/texlive2026.iso. 理解上述路径对应关系后, 进入 bash 并执行如下命令

```
sudo mkdir -p /mnt/iso
sudo mount -o loop /mnt/c/Users/username/Downloads/texlive2026.iso /mnt/iso
```

如此便可在 bash 中找到装载的光盘镜像.

后续安装过程与第 2.2 节一样, 安装完毕后, 需继续在 bash 中执行

```
sudo umount /mnt/iso
sudo rmdir /mnt/iso
```

弹出已装载的光盘镜像.

安装完成后, 用户需要设置环境变量, 方法同第 2.2.4 节. 退出 WSL 再进入, 执行

```
tex -v
```

将显示

```
TeX 3.141592653 (TeX Live 2026)
kpathsea version 6.4.2
Copyright 2026 D.E. Knuth.
```

[1]在 Windows 11 中直接进行的操作即为在主系统中的操作

```
There is NO warranty. Redistribution of this software is
covered by the terms of both the TeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the TeX source.
Primary author of TeX: D.E. Knuth.
```

即为安装成功.

后续配置字体缓存的方法同第 2.2.5 节.

4.2 升级宏包

WSL 中的方法同第 2.3 节.

4.3 安装宏包

WSL 中的方法同第 2.4 节.

4.4 调出宏包手册

由于 WSL 的 bash 默认是无窗口化的纯命令行, 因此将 TEX Live 安装至 WSL 后, 用户无法直接通过命令 `texdoc` 打开相应的手册. 下面几种方法均为收集而来的方案, 其中比较推荐更改 `texdoc.cnf`.

更改 `texdoc.cnf`

此方案源自本手册的 issue. 首先在 bash 中执行

```
texdoc --files
```

系统返回类似如下内容

```
/usr/local/texlive/2026/texmf-dist/scripts/texdoc/texdoclib.tlu 4.1.1
Configuration file(s):
    active /usr/local/texlive/2026/texmf-dist/texdoc/texdoc.cnf
Recommended file(s) for personal settings:
    /home/osbertwang/texmf/texdoc/texdoc.cnf
```

这里给出了 `texdoc` 的配置文件所在位置. 打开用户文件夹下的配置文件

```
mkdir -p ~/texmf/texdoc && touch ~/texmf/texdoc/texdoc.cnf
gnome-text-editor ~/texmf/texdoc/texdoc.cnf
```

在文件内写入

```
viewer_pdf = (powershell.exe -Command Start-Process $(wslpath -w %s)) &
```

并且保存退出. 然后再执行 `texdoc` 即可打开相应文档.

explorer 浏览

先找到手册所在路径, 例如寻找 lshort-zh-cn

```
texdoc -l lshort-zh-cn
```

系统会显示

```
 1 /usr/local/texlive/2026/texmf-dist/doc/latex/lshort-chinese/lshort-zh-cn.pdf
   = [zh] The document itself
Enter number of file to view, RET to view 1, anything else to skip:
```

这时按其他键 (如 x) 退出. 然后分别执行

```
cd /usr/local/texlive/2026/texmf-dist/doc/latex/lshort-chinese/
explorer.exe .
```

主系统的资源管理器便会打开. 这时用户便可以在主系统中使用 PDF 阅读器打开手册进行阅读.

everything 搜索

通过 `texdoc -l` 找到相应文件名, 而后直接使用 everything 搜索. 具体方法这里不再赘述.

借助其他工具

目前有很多开源的 WSL 辅助程序, 相关讨论见本手册的 issue.

4.5 编译文档

编译一个最小示例 `main.tex`.

在主系统的 cmd 中执行

```
mkdir D:\work-latex
cd /d D:\work-latex
notepad main.tex
```

建立文档, 内容为

```
\documentclass{article}
\begin{document}
  Hello \LaTeX\ World!
\end{document}
```

接下来在 cmd 中执行

```
bash -l -c "pdflatex main"
```

实现编译.

也可以直接在 WSL 中建立文档并编译, 在 bash 中依次执行

```
mkdir /mnt/d/work-latex
cd /mnt/d/work-latex
gnome-text-editor main.tex
```

建立文档, 内容同上, 然后在 bash 中执行

```
pdflatex main
```

实现编译.

待编译完成后, 可看到在工作路径中生成了 `main.pdf` 文件和其他同名的辅助文件 `main.aux` 与 `main.log`. 在主系统中可以打开 `main.pdf` 查看内容.

对于中文文档, 可以编写以下最小示例[2]

```
\documentclass{ctexart}
\begin{document}
  你好 \LaTeX\ 世界!
\end{document}
```

在 cmd 中执行

```
bash -l -c "xelatex main"
```

或在 bash 中执行

```
xelatex main
```

等待系统完成编译过程. `xelatex` 同样可以使用 WSL 内安装的字体[3], 在 WSL 中安装字体的方法见 B.2 节.

编译命令可添加参数, 这里与 1.6 节中的情形一致, 不再赘述.

4.6 卸载 TeX Live

WSL 中的方法同 2.7 节.

4.7 跨版本升级 TeX Live

WSL 中的方法同 2.8 节.

[2]注意使用 UTF-8 编码

[3]注意主系统中的字体默认不能直接被调用, 如果打算使用主系统中的字体, 见 GitHub 上的讨论, 鉴于 WSL2 和主系统文件系统互访速度慢, 该方法很可能拖慢 `xelatex` 速度

5 使用编辑器

5.1 Windows 11 系统

在实际操作中，用户会发觉使用记事本编写 `tex` 文件十分不便，因此很多用户都将其他编辑器作为自己的首选. 这里向大家介绍随 TEX Live 一同发行的轻量级编辑器 TEXworks 和功能更加丰富的 TEXstudio. 其他编辑器无法一一介绍，还请自己在网上寻求更多帮助.

5.1.1 TEXworks

TEXworks 是一款轻量级的 LATEX 编辑器，个人认为非常适合入门级用户使用[1].

和大多数国产软件不同，TEXworks 不会自动在桌面生成快捷方式，新人往往不知道如何打开它. 实际上 TEXworks 在 TEX Live 安装路径的 `\bin\windows` 中，可在 `cmd` 中执行 `texworks` 打开，也可直接在 Windows 搜索栏里搜 TEXworks 打开.

刚开始使用 TEXworks 的用户不必过多配置，在默认状态下打开软件、编写代码、保存文件后，用户可使用 TEXworks 的“排版”按钮进行文档编译而不必回到命令行[2]. 按钮旁边是编译命令菜单，用户可根据需要自行选择.

下面介绍几个常用的功能.

显示行号

打开 编辑 》 首选项 》 编辑器，选择 行号 即可.

调整字体

TEXworks 有时会擅自更改编辑器所使用的字体，很多字体“长相”诡异，让用户误以为是乱码. 用户可在 编辑 》 首选项 》 编辑器默认配置 和 格式 》 字体 处进行更改. Windows 11 中字体列表见这里.

自动补全

自动补全主要是 tab 键的功能，使用前需确定 首选项 》 编辑器 中是否勾选了 启动自动补全. 有关自动补全的更多内容用户可参考 TEXworks 的自动补全功能.

[1] 很多用户在编写 `tex` 文件时喜欢借助其他工具，这不是个好习惯，这里鼓励用户多手动敲代码

[2] 实际上编辑器的按钮也是调用命令行来编译文档，TEXworks 默认排版快捷键是 ctrl+T

使用模板

TEXworks 支持从模板新建文档, 在 文件 菜单中可见.

更改编译命令设置

TEXworks 默认提供了若干编译命令, 具体见 编辑 ▸ 首选项 ▸ 排版. 实际上, 用户可以根据自己的需要, 添加、更改其中的设置. 首先明确 TEXworks 中已有一些定义, 见表 5.1[3]. 接下来可以定义 latexmkpdf, 它调用 latexmk 并使用 pdflatex 等命令编译文档. 在 排版 上点击 +, 输入名称 latexmkpdf 和程序 latexmk, 相关参数需要再次点击 + 来予以添加, 例如添加以下这些

```
-e
$pdflatex=q/pdflatex $synctexoption %O %S/
-pdf
$fullname
```

这里的 $synctexoption 和 $fullname 均是 TEXworks 预定义的内容, 而其他的- e, $pdflatex=... 和- pdf 需要用户自己阅读 latexmk 手册了解. 然后选择**运行后查看 pdf 文件**, 再点击 OK, 即可发现已经添加成功. 之后用户可以调用 latexmkpdf 来编译文档. 当然, 在 排版 处还可以将 latexmkpdf 设置为默认.

类似地, 如果要定义 latexmkxe, 可以考虑以下参数

```
-e
$xelatex=q/xelatex $synctexoption %O %S/
-xelatex
$fullname
```

程序依然是 latexmk.

表 5.1: TEXworks 已有定义

Option	Description
$synctexoption	expands to "-synctex=1" if your tools support SyncTeX
$fullname	expands to the full name of your root document (e.g. rootfile.tex)
$basename	expands to the name (without extension) of your root document (e.g. rootfile)
$suffix	expands to the extension of your root document (e.g. tex)
$directory	expands to the absolute path to the directory containing your root document

拼写检查

默认情况, TEXworks 没有搭载拼写检查字典, 需要用户自己配置. 首先, 访问 OpenOffice 下载.oxt 文件. 将.oxt 文件重命名为.zip 文件, 解压缩得到里面的.dic 和.aff 文件. 接下来将它们安装到 <ResourcesPath>\dictionaries[4]. 安装完成后, 在 编辑 ▸ 拼写 中选择即可.

[3] 参考 Advanced Typesetting Tools

[4] <ResourcesPath> 的具体位置可以在 帮助 ▸ TeXworks 配置与资源 中找到, 若不存在该 dictionaries 文件夹, 用户可自行建立

魔法注释

魔法注释可以直接规定主文档、文件编码和编译命令等. 例如用户写学位论文时, 往往有一个主文档和若干子文档. 在子文档的开头写上魔法注释

```
% !TeX root = ../main.tex
```

即可告诉系统上一级目录中的 `main.tex` 是主文档[5].

```
% !TeX encoding = UTF8
```

表示该文档使用 “UTF-8” 编码[6].

```
% !TeX program = xelatex
```

表示该文档使用 `xelatex` 编译命令进行编译.

```
% !TeX spellcheck = <language_code>
```

表示该文档使用 `<language_code>` 进行拼写检查, 它的具体名称请用户到 编辑 ⟩ 拼写 中查看.

转编码

实际上转编码是很多编辑器都具备的功能. 在 TeXworks 右下角有三个按钮, 左边按钮控制换行符类型, 中间按钮控制文档编码, 右边按钮控制行跳转. 目前大部分中文用户主要面临的是 “UTF-8” 和 “GBK” 之间的转换. 如果文档是 “GBK” 编码, 使用 TeXworks 打开文件后, 文档会出现乱码. 这时, 点中间按钮, 选择编码类型 “GBK”, 再点击按钮, 选择 **使用所选编码重载文档**, 若文档中乱码消失, 则再次点击按钮, 选择 “UTF-8”, 最后保存文档, 完成转码工作.

除以上内容外, TeXworks 还支持正则表达式、自定义快捷键等. 这些内容都写在 TeXworks 自带的手册中. 手册不长, 用户可以通读一遍以了解更多内容. 很多功能可以通过**菜单栏**找到, 因此建议用户稍微了解一下它的菜单栏.

5.1.2 TeXstudio

相较于 TeXworks, TeXstudio 功能更丰富, 用法更多. 在使用 TeXstudio 前, 用户一定要查询**环境变量**中是否有 `system32`. 具体方法在前面已有说明, 此处不再赘述. 以下有关 TeXstudio 的用法介绍, 还请用户提前在 Options ⟩ configure TeXstudio 中勾选 Show Advanced Options.

下载方法

TeXstudio 有自己的官方网站, 但因网络原因, 打开不便. 网站上提供的下载链接跳转到了 GitHub 上, 有兴趣的朋友可以直接到 GitHub 上查看源码和可执行文件; 而清华大学, 北京外国语大学, 河南省教育科研网, 南京大学, 南阳理工学院也提供了相应的镜像, 便于用户下载. 此外, 有第三方网站可以辅助用户下载 GitHub 上的发行文件, 网站提供了下载教程, 这里不再展开.

[5] 主文档和子文档的更多内容请参考\ include 和\ input 的用法

[6] 目前建议用户多用 “UTF-8” 编码, 尤其是中文文档

更改默认编译命令

在 Options » configure TeXstudio » build » default compiler 中选择默认编译命令.

另外用 `latexmk` 和 `biblatex` 做编译时, 需要在 Options » Configure TeXstudio » build » Build Options 中取消 Check and update bibliography before compiling.

显示行号

在 Editor » Show Line Numbers 选择行号如何显示.

数学符号表

在 TeXstudio 左下角有两个按钮: Side Panel 和 Messages / Log file. 打开 Side Panel, 点击软件左侧边栏第 4 个图标 Symbols.

插入、改写调整

与很多其他编辑器一样, TeXstudio 也可以通过键盘 insert 改变插入、改写状态. 分别处于插入、改写状态时, 编辑器光标会呈现不同的样子.

调整缩进

在 Options » Configure TeXstudio » Editor » Indentation Mode 处可以选择是否保持缩进. 如果需要让缩进完全变为空格, 勾选 Replace Indentation Tab by Spaces.

在 Adv. Editor » Appearance 中可以更改 Tab Widths.

块选择模式

同时按 Ctrl+Alt 便可通过鼠标左键进行块选择.

调整图标大小

有些用户屏幕分辨率高, 因此软件上图标会显得很小. 在 Options » Configure TeXstudio » Gui Scaling 处可以调整三处的显示比例.

自动补全和未识别代码

TeXstudio 自动补全功能会先通过用户引用的宏包来判断代码是否需要自动补全, 同时它也可以以此判定用户输入的命令是否正确. 若系统判断命令无法识别, 则该命令会出现红色背景.

TeXstudio 能够识别的命令全部被写入了 `cwl` 文件中. 用户可以根据自己的需要更改 `cwl` 文件, 然后将其放入 `<SettingsPath>\completion\user` 文件夹[7].

自动补全后有可能生成文本框, 使用 Ctrl+→ 可以跳转至下一个文本框, Ctrl+← 则可以跳转至上一个.

[7] 点击 Help » Check LaTeX Installation, 在生成的 `System Report.txt` 中找到 `texstudio.ini` 文件, 其所在文件夹即为 `<SettingsPath>`

拼写检查

在 Options > Configure TeXstudio > Language Checking > Default Language 处选择 en_US 即可默认进行英文拼写检查.

魔法注释

除了像 TeXworks 中使用魔法注释外, TeXstudio 还有很多其他的魔法注释, 例如

```
% !TeX TXS-program:compile = txs:///latexmk/{}[-xelatex -synctex=1 -interaction=nonstopmode ?me"]
```

表示使用 latexmk 编译命令, {} 表示无视编辑器赋予该命令的一切参数, 而 [] 表示添加其中的参数进行编译, 本例中添加- xelatex -synctex=1 -interaction=nonstopmode 作为参数. 如果用户不清楚魔法注释都可以写哪些类型, 可以点击 LaTeX > Add magic comments 查看五种魔法注释类型. 更多内容需要阅读软件手册.

渲染方式

默认的渲染方式对中文括号的支持不够好. 用户可以在 Options > Configure TeXstudio > Adv. Editor > Hacks/Workarounds 中取消 Try to automatically choose best display options, 选择 Disable cache of character width 和 Disable fixed pitch mdoe. 这样能够比较好地渲染中文括号.

转编码

TeXstudio 右下角有一处显示文档编码 (Encoding). 依旧以 “UTF-8” 和 “GBK” 之间的转换为例. 如果文档是 “GBK” 编码, 使用 TeXstudio 打开文件后, 点击 Encoding 处, 选择 More Encodings. 打开窗口后, 选择 “GBK / ...”, 点击 Reload With. 若这时文档没有乱码, 再点击 Encoding > More Encodings, 选择 “UTF-8”, 点击 Change To, 保存文件, 完成转码. 在 Edit > Setup Encoding ... 处同样可以打开编码窗口.

自定义命令并生成按钮

TeXstudio 允许用户自定义命令, 并将命令做成按钮放置于面板上. 如在 Options > Configure TeXstudio > Build > User commands 中, 填写名称 User1:Build-xe-view 和对应的功能

```
latexmk -pdfxe -silent -synctex=1 -interaction=nonstopmode ?me" | txs:///view-pdf-internal
    --embedded
```

点击 OK. 接下来我们在 Tools > User 中即可看到自定义的命令和相应快捷键.

接下来设置按钮. 打开 Options > Configure TeXstudio > Toolbars, 在两个下拉菜单中分别选择 Tools 和 All menus, 在右边找到 Tools > User > Build-xe-view 并将其添加至左端, 点击 OK. 这时, 在面板中将添加新的按钮.

生成宏指令

TeXstudio 允许用户生成宏指令. 在 Macros > Edit Macros 中, 用户可以根据自己的需要, 给出宏指令的名称、快捷键、内容等等. 例如我们将 latexmk-pdf 填写在 name 中, 将

```
% !TeX TXS-program:compile = txs:///latexmk/{}[-pdf -synctex=1 -interaction=nonstopmode ?me"]
```

填写在 LaTeX Content 中，再将 Shift+F1 填写在 Shortcut. 设置完毕后，点击 OK. 这时，我们可以看到 Macros 》latexmk-pdf 出现. 在文档特定位置点击它或者直接使用 ⇧+F1，即可看到魔法注释出现.

更改颜色方案

有一些用户喜欢深色的颜色方案. 手动修改颜色不方便的话，我们可以从网上找到一些现成的颜色方案，例如 StackExchange 网站和某些用户的博客. 更改方案只需要打开 ini 文件，将这些颜色方案复制到 [formats] 部分. 在 Help 》Check LaTeX Installation 中有 ini 文件的具体位置.

窗口化/扩大化内嵌的 PDF 阅读器

TEXstudio 已经内嵌了 PDF 阅读器供用户预览编译结果，通常居于编辑器右侧. 在该阅读器上有一个 Windowsed Viewer 按钮，点击它，位于右侧的 PDF 阅读器可以变为一个单独的窗口出现. 重新恢复到内嵌模式只需要点击 Embedded Viewer 按钮. 另外还有 Enlarge Viewer 按钮，可以扩大 PDF 阅读器到整个编辑区，恢复大小只需要点击 Shrink Viewer 按钮.

调用外部 PDF 阅读器

一些用户喜欢使用外部 PDF 阅读器，这里以 SumatraPDF 为例介绍外部 PDF 阅读器的设置，用户直接安装或下载便携版的 SumatraPDF 均可，假定它在本地的位置为 <SumatraPDFPath>，而 TEXstudio 在本地的位置是 <TeXStudioPath>.

首先设置 TEXStudio。打开 Options 》Configure TeXstudio 窗口，在 Commands 》External PDF Viewer 中写入

```
dde:///"<SumatraPDFPath>\SumatraPDF.exe":SUMATRA/control/[ForwardSearch(?p{pdf}:ame",?c:rme",@
    ,0,0,1)]
```

接下来，将 Build 》PDF Viewer 改为 txs:///view-pdf-external. 这时，用户使用快捷键 F7 可打开 SumatraPDF 并实现了正向搜索. 另一种更贴近命令行的设置方法，在 Commands 》External PDF Viewer 中写入

```
"<SumatraPDFPath>\SumatraPDF.exe" -reuse-instance -forward-search ?c:rme" @ ?p{pdf}:ame" &
```

注意 & 是近期 TEXStudio 强制要求的.

接下来设置 SumatraPDF. 在菜单栏 Settings 》Advanced Options 中找到并设置其键值

```
EnableTeXEnhancements = true
ReuseInstance = true
ReloadModifiedDocuments = true
InverseSearchCmdLine = "<TeXStudioPath>\texstudio.exe" "%f" -line %l
```

至此完成了逆向搜索设置，双击 PDF 文件便可回到 TEXstudio 中对应代码的行首.

若要恢复内置的阅读器，将 Build 》PDF Viewer 改为 txs:///view-pdf-internal --embedded 即可.

5.2 Ubuntu 26.04 系统

简化起见, 这里只介绍如何使用 TeXstudio.

5.2.1 使用源内安装

根据官网推荐, 我们安装源内的 TeXstudio. 安装前可参照 B.1 节的介绍换源以增速.

在 Terminal 中执行

```
sudo apt install texstudio
```

即可安装 TeXstudio. 注意安装过程中会产生一些依赖, 它们包括了源内的 TeX Live 包, 如 `tex-common`, `texlive-base`, `texlive-binaries`, `texlive-latex-base` 和 `texlive-latex-recommended`. 用户需要卸载它们, 参见 2.1.3 节.

当然用户也可以使用 PPA 来安装 TeXstudio. 以此在 Terminal 中执行

```
sudo add-apt-repository ppa:sunderme/texstudio
sudo apt update
sudo apt install texstudio
```

鉴于网络原因, 直接从 PPA 下载容易丢包, 因此可以使用反向代理, 具体见 B.3 节.

安装完毕后, 用户可以在 Terminal 中执行

```
texstudio main.tex
```

使用 TeXstudio 编辑文档. 也可直接双击 `main.tex` 文件.

5.2.2 使用 AppImage

目前 TeXstudio 也提供了 AppImage 版本, 下载方法同 5.1.2 节. 下载后, 首先执行

```
sudo apt install libfuse2
```

再右键 TeXstudio 的 AppImage 文件, 在 Properties 》 Permissions 中勾选 Allow executing files as program. 随后可双击打开 TeXstudio.

AppImage 文件本身不会自动集成到系统中, 要将 TeXstudio 的 AppImage 设置为 `.tex` 文件的默认打开程序, 核心步骤是先创建一个桌面条目 `.desktop` 文件, 然后将 `.tex` 文件关联到这个条目. 执行

```
  gnome-text-editor ~/.local/share/applications/texstudio.desktop
```

并在其中复制

```
  [Desktop Entry]
  Type=Application
  Name=TeXstudio
  Comment=LaTeX Editor
  Exec=<TeXStudio.AppImagePath>/texstudio.AppImage %f
  Icon=<TeXStudio.AppImagePath>/texstudio-icon.png
  Terminal=false
```

```
Categories=Office;Publishing;Qt;
MimeType=text/x-tex;
```

这里的 <TeXStudio.AppImagePath> 是用户保存 TeXstudio 的 AppImage 文件的实际路径, 注意文件名填写正确. Icon 路径可以指向你下载的图标, 也可以留空或使用通用图标名. 保存退出. 接下来执行

```
xdg-mime default texstudio.desktop text/x-tex
update-desktop-database ~/.local/share/applications
```

完成设置. 此后即可用

```
xdg-open filename.tex
```

通过 TeXstudio 打开文件.

若要恢复系统默认状态, 依次执行

```
xdg-mime default org.gnome.TextEditor.desktop text/x-tex
rm ~/.local/share/applications/texstudio.desktop
update-desktop-database ~/.local/share/applications
```

即可.

5.2.3 额外设置路径

如果 TeXstudio 未能正确寻得引擎, 可在 Options 》Configure TeXstudio 中点击 Show Advanced Options, 接下来在 Build 》Build Options 》Commands ($PATH) 中添加路径, 默认安装时为

```
/usr/local/texlive/2026/bin/x86_64-linux
```

其他设置可参考前文, 这里不再赘述.

5.3 macOS

这里介绍两款 macOS 下的编辑器: TeXShop 和 TeXstudio.

5.3.1 TeXShop

MacTeX 自带 TeXShop 编辑器 (注意不要与其他自带的程序混淆), 通常在安装完毕后便可立即使用. 通过编辑器打开 main.tex, 将排版程序 (工具栏上面左数第二个下拉菜单) 由默认的 LaTeX 切换到 pdflatexmk, 再点击旁边的排版 排版 (Typeset), 待编译完成后会弹出 PDF 预览界面.

编译中文文档, 通常需要将排版程序切换到 XeLaTeX. 注意可能需要多次编译以生成正确的交叉引用等内容.

5.3.2 TeXstudio

如果希望使用 TeXstudio, 也可以通过 Homebrew 安装:

```
brew install --cask texstudio
```

注意因为网络问题可能会下载失败.

也可参照 5.1.2 节下载并手工安装 TeXstudio, 此处不再赘述. 其他设置与 Windows 系统中大致相仿.

无法显示文字

TeXstudio 在 macOS 下有可能无法显示文字, 这是其他系统下目前没有发现的问题. 具体解决方法见李阿玲在知乎的文章.

5.4 Windows Subsystem for Linux

简化起见, 只介绍 TeXstudio 的用法. TeXstudio 在 WSL 中的安装与使用同 5.2 节所介绍的一致. 只不过此时只能使用 WSL 来安装、启动、编写和编译文档, 在主系统中的 TeXstudio 无法在 WSL 起作用.

5.5 其他编辑器

VS Code 配合插件 LaTeX Workshop 可以作为一款跨平台的编辑器来使用, 具体设置方式建议阅读插件 Wiki 了解更多内容, 一些网友的教程可以参考, 如 EthanDeng 的教程. 用户可以根据自己的实际需求、使用习惯和个人能力, 自行决定是否使用. 一份个人使用的配置单可参考 B.4 节.

Vim 是一款优秀的编辑器, 它的用法较目前常见的编辑器有很大不同, 因此用户上手时容易感觉无所适从, 但熟练掌握用法后, 用户将很容易对其进行定制. Vim-LaTeX 尝试提供一套全面的工具来查看、编辑和编译 LaTeX 文档, 用户可以考虑使用.

更多编辑器可参考 TUG 的列表.

5.6 多款编辑器对比参考

显然, 能够支持 LaTeX 的编辑器并不局限于本手册所介绍的几款. 表 5.2 是 EthanDeng 在 LaTeX 小圈子内部对 LaTeX 编辑器所进行的用户体验调查的结果, 比较主观, 仅供参考. 其中 Sublime Text 和 Visual Studio Code 为文本编辑器, 它们需要借助插件才能支持 LaTeX 编译. 对自己网络非常自信的用户也可以参考 Wikipedia 上的对比结果.

表 5.2: LaTeX 编辑器对比

	WinEdt	TeXstudio	TeXworks	Sublime Text	VS Code
插件依赖				LaTeXTools	LaTeX Workshop
主流系统	Win	全平台	Linux/Win	全平台	全平台
软件类型	商业软件	开源软件	开源软件	商业软件	开源软件
软件价格	179 元	0	0	99 美元	0
授权方式	终身/教育			终身/个人	
代码高亮	★★★☆☆	★★★☆☆	★★☆☆☆	★★★★☆	★★★★½
颜色主题	★★☆☆☆	★★☆☆☆	★☆☆☆☆	★★★★☆	★★★★☆
自动补全	★★★☆☆	★★★½☆	★★☆☆☆	★★★½☆	★★★★☆
代码片段	★★★☆☆	★★½☆☆	★☆☆☆☆	★★★★☆	★★★★☆
辅助输入	★★★★☆	★★★½☆	★☆☆☆☆	★★☆☆☆	★★★☆☆
开发完成	★★★★☆	★★★★☆	★★★★½	★★★½☆	★★★★☆
推荐指数	★★★☆☆	★★★★☆	★★☆☆☆	★★★☆☆	★★★★½

6 在线 LaTeX 平台

在特定场合, 有些用户并不需要也没条件在本地安装发行版, 因此这里额外补充一些在线 LaTeX 平台的相关内容, 如国外的 Overleaf, 和国内的一些平台. 另外还有一些科研院所自己搭建的在线 LaTeX 平台也可以在一定程度上满足用户的需要.

6.1 Overleaf

6.1.1 注册 Overleaf

Overleaf 是全球范围内首屈一指的在线 LaTeX 编辑平台. 它为每位用户提供了 Ubuntu 系统下的 TeX Live. 它优秀的协作功能、丰富的模板仓库已吸引全球科研工作者成为它的用户. Overleaf 提供了包括中文在内的多种语言供用户使用. 2026 年 8 月 18 日, 它将后台的 TeX Live 升级为 2026 版本, 同时, Overleaf 还允许用户自主选择项目中的 TeX Live 版本.

目前 Overleaf 允许用户使用中文, 并为用户预先准备了一些中文字体. 这些中文字体可通过 CTeX 宏集调用. 特别推荐用户使用思源宋体和思源黑体这两种开源中文字体.

遗憾的是, 国内网络环境会对 Overleaf 所使用的 reCaptcha 造成影响. 这也使得很多用户在直接注册 Overleaf 时就遇到了问题.

目前, 比较好的替代方案是借助 ORCID 来进行注册. 目前使用国内网络访问 ORCID 还比较流畅. 用户, 尤其是科研工作者, 可以先注册一个 ORCID 账号. 未来投稿时, 可将 ORCID 账号与自己的期刊网站账号进行绑定. 同时, 用户也可逐步将自己所发表论文列在 ORCID 网站以便管理.

6.1.2 使用 Overleaf

用户通过 ORCID 注册 Overleaf 后便可进入自己的项目列表页面进行使用.

新建项目是用户首先使用的功能. Overleaf 提供了多种渠道为用户新建项目: 可以通过 Overleaf 中的模板, 也可以通过上传本地的 `zip` 压缩文件包.

新建项目后, 用户便可进入编辑界面. 在编辑界面用户需要先在左上角 File » Settings » Compiler 中, 选择合适的编译命令和主文件. 由于默认字体和文件编码等原因, 强烈建议用户在处理中文文档时使用 XeLaTeX. 编写文档后, 用户可通过鼠标点击按钮进行编译, 也可使用快捷键 ctrl + enter.

另外, Overleaf 在 2022 年 9 月 30 日更新了新的功能, Stop on first error, 这个功能一旦开启, 出现报错就立即停止编译. 它可以让用户高度关注报错信息并改正, 也可以帮助纠正一些会导致超时 (timeout) 的代码错误, 例如 TikZ 里的`\ draw` 忘了最后的`;` .

用户编写的文件会保存在网站. 编写完成后, 用户只需点击 File 左边的图标回到项目列表. 这时可

以看到新增项目右侧有五个图标, 它们分别是 Copy、Download .zip file、Download PDF、Archive 和 Trash. 用户可根据自己的需求点击合适的图标.

6.1.3 选择不同发行版版本

Overleaf 将后台 TeX Live 升级后, 用户新建项目默认使用 TeX Live 2024, 而老项目还是使用 TeX Live 的老版本. 如果用户不打算使用 TeX Live 2024, 只需在 File 》Settings 》Compiler 选择版本即可.

6.1.4 学习与帮助

Overleaf 的模板和文档对全网公开, 用户可以自行学习. 另外 Overleaf 有着专业的技术支援团队, 用户可发送邮件至 support@overleaf.com 咨询使用过程中遇到的问题, 在邮件中请注意文明用语. 部分问题会超出免费服务的范畴, 用户需谨记这点.

6.2 国内的在线 LaTeX 平台

相较于因网络问题难以注册和使用 Overleaf，国内的在线 LaTeX 平台则不存在这一障碍。因此，本手册将对其进行简要介绍，以便用户选择。

6.2.1 TeXPage

用户只需访问 TeXPage 的主页, 使用邮箱注册即可, 具体过程不再赘述.

新用户注册 TeXPage 后会在页面看到一份使用教程. 这份教程简明扼要地概括了一般中文用户会遇到的常见问题, 然而从代码的角度而言, 我个人不欣赏在浮动体中使用 H 选项的做法.

在页面右上角的位置有许多菜单, 在**设置**菜单中, 用户可以选择默认的编译命令、发行版版本等等. 设置完毕, 再点击编译即可.

TeXPage 有自己的文档中心, 文档数量不多, 但涵盖了相当一部分基础知识. 同时, 它也提供了联系方式 support@texpage.com, 用户如果遇到了一些使用上的问题也可以直接发邮件咨询. 当然用户在提问前也可以先尝试使用平台自带的 AI 调试和 AI 润色功能.

目前在 TeXPage 上有一些颇具特色的模板, 例如建模比赛的一些模板就已经被收录在 TeXPage 当中. TeXPage 的产品定价涵盖了基础版、专业版、旗舰版和企业版, 用户可以按需购买.

6.2.2 TeXHub

TeXHub 是一个专为学术研究者打造的下一代 LaTeX 写作平台, 集团队协作与直观界面于一体. 它整合了学术研究者所需的各类工具, 让用户能够专注于内容创作, 而无需为繁琐的格式调整费心. 据称, 该平台计划不按算力收费, 即没有编译时间限制. 此外, 网站还以建立全国高等教育和科研图库为目标, 愿景较为宏大.

6.2.3 LoongTeX

LoongTeX 集成了白板、类 Notion 笔记管理、异步协作和 AI 辅助功能, 助力高效完成论文与文档. 新用户可通过微信、谷歌账号或邮箱快速注册登录, 即刻进入专属工作台界面.

LoongTeX 工作台提供创建项目、新建笔记、发起白板三类核心创作入口, 创建项目支持三种高效启动方式: 上传 `zip` 压缩包、从毕业论文库生成、或绑定 Git 仓库实现云端同步协作.

新建项目后, 左侧导航栏集成文件树管理、可视化 Git 版本历史及协作成员面板, 中央编辑区搭载智能 LaTeX 编辑器与分屏 PDF 预览, 右侧功能面板提供 AI 助手和智能批注系统.

LoongTeX 官网内嵌帮助中心, 提供分场景的 LaTeX 写作指南、异步协作操作手册及 AI 助手使用技巧. 用户可发送邮件至 `loongtex@gmail.com` 咨询使用过程中遇到的问题. 订阅方案涵盖了专业版和团队版, 用户可以根据实际需要考虑购买.

6.2.4 Keepresearch

Keepresearch 是一个较新的国内平台, 据官网介绍, 该平台的特色是融合了人工智能技术, 例如 AI 分析、AI 演示文稿等. 此外, 该平台还具备较强的文件管理功能, 并提供了若干拓展工具. 目前该平台已经涵盖了 2025 国自然基金模板, 相信能够在一定程度上提升科研工作者的工作效率.

6.2.5 FlyLaTeX

FlyLaTeX 是最近才出现的国内平台, 有关它的介绍还不多, 官网也并未对其特色进行太详尽的说明. 该网站的用户手册内容较为丰富, 用户可以参考.

6.3 科研院所搭建平台

大陆地区部分高校 (例如清华大学和中国科学技术大学) 搭建了供内部师生使用的平台.

中国科学院计算机网络信息中心科技云运行与技术发展部曾发邮件宣称开始提供论文协同编辑服务, 据我猜测, 它利用了早年 Sharelatex 的部分内容. 试用可点击这里.

附录 A　大陆地区的源

由于一些未知的原因, 下载 TEX Live 的 `iso` 文件 (在 MacOS 中下载 MacTEX) 或升级宏包时, 系统自动选择的源并没有满足用户关于 “附近” 的要求. 因此, 很多时候, 用户需要手动选择 CTAN 的源. 在 CTAN 可以找到亚洲镜像列表和连接情况. 我将列表中位于大陆地区的源整理到表 A.1, 并将我个人收集到的其他大陆地区的源整理到表 A.2. 其中, 高校联合镜像是一个仿照 CTAN 镜像的国内镜像源, 它也可以 “自动” 跳转到 “附近” 的高校镜像. 更多内容参见高校联合镜像说明. 另有一些源仅限单位内部访问, 我在表 A.3 中单独列出.

使用最近的 CTAN 源下载 `iso` 文件, 实际上是下载

```
https://mirrors.ctan.org/systems/texlive/Images/texlive2026.iso
```

使用大陆的源下载它, 只需将以上网址中的

```
https://mirrors.ctan.org/systems/texlive/
```

替换为相应的源的地址, 而将

```
Images/texlive2026.iso
```

予以保留. 以清华大学为例, 从清华大学源中下载 `iso` 镜像文件, 实际上是下载

```
https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/texlive2026.iso
```

MacTEX 的镜像路径需将 TEX Live 路径中的 `.../texlive/Images` 替换为 `.../mac/mactex`. 例如, 从清华大学源中下载 `MacTeX.pkg`, 即为下载

```
https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/mac/mactex/MacTeX.pkg
```

其他源类似, 不再列举.

升级宏包也只需要将对应的网址替换为源地址即可, 并且 TEX Live 与 MacTEX 用法统一. 正文中

```
tlmgr option repository ctan
```

等价于

```
tlmgr option repository https://mirrors.ctan.org/systems/texlive/tlnet/
```

使用大陆的源升级宏包, 只需将

```
https://mirrors.ctan.org/systems/texlive/
```

替换为大陆源的地址而保留其他部分即可. 以清华大学为例, 设置将变为

```
tlmgr option repository https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/tlnet/
```

表 A.1: CTAN 列表中大陆地区目前可用的源 (名称按拼音排序)

阿里云	https://mirrors.aliyun.com/CTAN/systems/texlive/
北京大学	https://mirrors.pku.edu.cn/ctan/systems/texlive/
北京外国语大学	https://mirrors.bfsu.edu.cn/CTAN/systems/texlive/
重庆大学	https://mirrors.cqu.edu.cn/CTAN/systems/texlive/
哈尔滨工业大学	https://mirrors.hit.edu.cn/CTAN/systems/texlive/
华中科技大学	https://mirrors.hust.edu.cn/CTAN/systems/texlive/
吉林大学	https://mirrors.jlu.edu.cn/CTAN/systems/texlive/
南方科技大学	https://mirrors.sustech.edu.cn/CTAN/systems/texlive/
南京大学	https://mirrors.nju.edu.cn/CTAN/systems/texlive/
清华大学	https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/
上海交通大学	https://mirrors.sjtug.sjtu.edu.cn/ctan/systems/texlive/
腾讯云	https://mirrors.cloud.tencent.com/CTAN/systems/texlive/
浙江大学	https://mirrors.zju.edu.cn/CTAN/systems/texlive/

表 A.2: 自己额外收集的大陆地区目前可用的源 (名称按拼音排序)

重庆邮电大学	https://mirrors.cqupt.edu.cn/ctan/systems/texlive/
高校联合镜像	https://mirrors.cernet.edu.cn/CTAN/systems/texlive/
河南省教育科研网	https://mirrors.ha.edu.cn/CTAN/systems/texlive/
华为云	https://mirrors.huaweicloud.com/CTAN/systems/texlive/
兰州大学	https://mirror.lzu.edu.cn/CTAN/systems/texlive/
南阳理工学院	https://mirror.nyist.edu.cn/CTAN/systems/texlive/
齐鲁工业大学	https://mirrors.qlu.edu.cn/CTAN/systems/texlive/
山东大学	https://mirrors.sdu.edu.cn/CTAN/systems/texlive/
上海科技大学	https://mirrors.geekpie.club/CTAN/systems/texlive/
西北农林科技大学	https://mirrors.nwafu.edu.cn/ctan/systems/texlive/
中国科学技术大学	https://mirrors.ustc.edu.cn/CTAN/systems/texlive/
中山大学	https://mirror.sysu.edu.cn/CTAN/systems/texlive/

表 A.3: 仅限单位内部访问的源 (名称按拼音排序)

北京理工大学	https://mirrors.bit.edu.cn/CTAN/systems/texlive/

附录 B　有关 Ubuntu 和 WSL 的补充

B.1　更换 Ubuntu 26.04 源

鉴于大陆网络现状, 建议更换 Ubuntu 26.04 源至国内, 例如更换至清华大学. 在 Terminal 中执行

```
sudo cp /etc/apt/sources.list.d/ubuntu.sources /etc/apt/sources.list.d/ubuntu.sources.bak
```

备份 ubuntu.sources 文件. 接下来执行

```
sudo gnome-text-editor /etc/apt/sources.list.d/ubuntu.sources
```

打开文件, 将文件内容替换为[1]

```
Types: deb
URIs: https://mirrors.tuna.tsinghua.edu.cn/ubuntu
Suites: resolute resolute-updates resolute-backports
Components: main restricted universe multiverse
Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg

# 默认注释了源码镜像以提高 apt update 速度, 如有需要可自行取消注释
# Types: deb-src
# URIs: https://mirrors.tuna.tsinghua.edu.cn/ubuntu
# Suites: resolute resolute-updates resolute-backports
# Components: main restricted universe multiverse
# Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg

# 以下安全更新软件源为官方源配置
Types: deb
URIs: http://security.ubuntu.com/ubuntu/
Suites: resolute-security
Components: main restricted universe multiverse
Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg

# Types: deb-src
# URIs: http://security.ubuntu.com/ubuntu/
# Suites: resolute-security
# Components: main restricted universe multiverse
# Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg
```

[1] 本段文字可至清华大学镜像网站获取

```

# 预发布软件源，不建议启用

# Types: deb
# URIs: https://mirrors.tuna.tsinghua.edu.cn/ubuntu
# Suites: resolute-proposed
# Components: main restricted universe multiverse
# Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg

# # Types: deb-src
# # URIs: https://mirrors.tuna.tsinghua.edu.cn/ubuntu
# # Suites: resolute-proposed
# # Components: main restricted universe multiverse
# # Signed-By: /usr/share/keyrings/ubuntu-archive-keyring.gpg
```

保存退出. 然后执行

```
sudo apt update && sudo apt upgrade
```

换源并更新. 若更改错误, 可执行

```
sudo cp /etc/apt/sources.list.d/ubuntu.sources.bak /etc/apt/sources.list.d/ubuntu.sources
```

恢复文件.

B.2 安装字体

有用户指出, Ubuntu 中安装字体比较麻烦. 这里给出一些常用的方法, 可供参考[2].

B.2.1 获取字体

获取需要安装的字体文件, 可以分为两种情况.

个人用户

若只对个人用户开放字体使用, 可将字体文件保存在 `~/.local/share/fonts/` 路径下, 若路径不存在, 可以使用以下命令创建

```
mkdir -p ~/.local/share/fonts/
```

推荐将不同的字体文件保存在 `~/.local/share/fonts/` 的不同子路径中, 方便管理字体. 下述涉及路径的内容均类似, 不再重复.

WSL 用户甚至可以将字体文件保存在主系统路径中, 例如 `C:\Users\admin\somefont`, 然后对字体路径做链接

```
ln -s /mnt/c/Users/admin/somefont ~/.local/share/fonts/windows-fonts
```

[2] 关于此处字体的更多讨论请见这里.

所有用户

若对所有用户开放字体使用, 可将字体文件保存在 `/usr/local/share/fonts/` 路径下. 同样地, 可能需要创建目录

```
sudo mkdir -p /usr/local/share/fonts
```

再次推荐将不同的字体文件保存在不同的子路径中. 值得注意的是, 此时字体文件需要具有足够的读取权限, 例如对字体文件 `fontname` 和字体文件路径 `fontdirectory` 分别设置权限如下

```
sudo chmod 644 /usr/local/share/fonts/fontname
sudo chmod -R 755 /usr/local/share/fonts/fontdirectory
```

这里设置 644 和 755 只为后续使用更便利, 严格来讲, 如果只涉及字体读取, 444 和 555 即可.

WSL 用户若要将主系统路径中的字体文件对所有用户开放字体使用, 可以做如下链接

```
sudo ln -s /mnt/c/Users/admin/somefont /usr/local/share/fonts/windows-somefonts
```

B.2.2 使用 fontconfig

对于绝大多数现代 TrueType (TTF), OpenType (OTF) 等字体, `fontconfig` 是唯一需要的工具.

在获取字体后, 使用 `fontconfig` 非常简单, 只需执行

```
fc-cache -fv
```

即可建立字体缓存, 注意刷新 `/usr/local/share/fonts/` 内的字体缓存时需要添加 `sudo`. 随后执行

```
fc-list
```

即可见到目前可用的字体.

若不再需要已经安装的字体, 只需要直接删除, 例如

```
rm ~/.local/share/fonts/fontname
sudo rm /usr/local/share/fonts/fontname
```

随后刷新缓存

```
fc-cache -fv
```

查看字体是否已经删除

```
fc-list
```

B.2.3 使用 xfonts-utils

此处介绍 `xfonts-utils` 只为补充, 它主要用于处理非常旧的 X11 位图字体 (pcf, bdf) 或某些特定的 Type1 字体场景, 普通用户安装现代字体几乎不需要它.

在 Ubuntu 26.04 中, `xfonts-utils` 并没有默认安装. 可直接使用以下命令安装

```
sudo apt install xfonts-utils
```

获取字体后, 通过以下命令建立字体缓存

```
mkfontscale
mkfontdir
```

再次强调, 对于如今大部分字体文件而言, 已不再推荐使用 `xfonts-utils`.

B.2.4 直接安装源内字体

实际上有部分字体可直接使用源进行安装, 如安装字体 Noto CJK

```
sudo apt install fonts-noto-cjk
sudo apt install fonts-noto-cjk-extra
```

安装 Windows 系统中常见的字体 Andale Mono, Arial Black, Arial (Bold, Italic, Bold Italic), Comic Sans MS (Bold), Courier New (Bold, Italic, Bold Italic), Georgia (Bold, Italic, Bold Italic), Impact, Times New Roman (Bold, Italic, Bold Italic), Trebuchet (Bold, Italic, Bold Italic), Verdana (Bold, Italic, Bold Italic), Webdings

```
sudo apt install ttf-mscorefonts-installer
```

安装字体文泉驿正黑, 文泉驿微米黑, 文泉驿等宽微米黑, 文泉驿点阵宋体, 文泉驿点阵楷体

```
sudo apt install ttf-wqy-zenhei
```

B.3 反向代理 PPA

这里介绍中科大的反向代理, 将

```
/etc/apt/sources.list.d/
```

中文件的内容

```
http(s)://ppa.launchpad.net
```

改为

```
https://launchpad.proxy.ustclug.org
```

而后再

```
sudo apt update
```

即完成换源. 有时地址太多而逐个换太麻烦, 可以直接使用如下命令直接替换[3]

```
sudo find /etc/apt/sources.list.d/ -type f -name "*.list" -exec sed -i.bak -r 's#deb(-src)?\s*http(
    s)?://ppa.launchpad.net#deb\1 https\2://launchpad.proxy.ustclug.org#ig' {} \;
```

[3]参见 https://mogeko.me/zh-cn/posts/zh-cn/035/

B.4 用 VS Code 作为 WSL 中的编辑器

微软最近频繁推荐 WSL 作为开发工具, 详情阅读微软文档. 本手册在此基础上, 借助 VS Code 的 Remote Development 插件实现若干功能. 若非特别指明, 以下操作均在 WSL 中用 bash 执行.

VS Code 的插件权限有限, 当更改 ubuntu.sources 时, 先将 ubuntu.sources 备份

```
sudo cp /etc/apt/sources.list.d/ubuntu.sources /etc/apt/sources.list.d/ubuntu.sources.bak
```

再用 VS Code 打开文档

```
code /etc/apt/sources.list.d/ubuntu.sources
```

并对其中内容进行更改, 并且另存为

```
~/ubuntu.sources.sav
```

执行

```
sudo cp ~/ubuntu.sources.sav /etc/apt/sources.list.d/ubuntu.sources
```

替换文档. 若发现文档有错误, 执行

```
sudo cp /etc/apt/sources.list.d/ubuntu.sources.bak /etc/apt/sources.list.d/ubuntu.sources
```

恢复文档.

更改.bashrc 相对简单, 将.bashrc 备份

```
cp ~/.bashrc ~/.bashrc.bak
```

用 VS Code 打开文档

```
code ~/.bashrc
```

并对其中内容进行更改并保存. 若更改错误, 可以执行

```
cp ~/.bashrc.bak ~/.bashrc
```

恢复文档.

利用 LaTeX Workshop 插件可以将 VS Code 作为 LaTeX 编辑器, 难点在于如何设置. 这里还是建议用户阅读插件 Wiki 了解更多内容. 以下 VS Code 设置的 json 文件内容为个人自用的 LaTeX Workshop 设置, 仅供参考.

```
"latex-workshop.latex.tools": [
{
  "name": "pdflatexmk",
  "command": "latexmk",
  "args":
  [
    "-pdf",
    "-synctex=1",
    "-halt-on-error",
    "-file-line-error",
```

```
    "-interaction=nonstopmode",
    "%DOCFILE%",
  ]
},
{
  "name": "xelatexmk",
  "command": "latexmk",
  "args":
  [
    "-pdfxe",
    "-synctex=1",
    "-halt-on-error",
    "-file-line-error",
    "-interaction=nonstopmode",
    "%DOCFILE%"
  ]
},
],
"latex-workshop.latex.recipes": [
{
  "name": "pdflatexmk",
  "tools": [ "pdflatexmk"]
},
{
  "name": "xelatexmk",
  "tools": [ "xelatexmk"]
}
],
"latex-workshop.latex.autoBuild.run": "never",
"latex-workshop.view.pdf.viewer": "tab"
```

需要声明的是, LaTeX Workshop 插件并不局限在 WSL 当中, 它的配置是通用的. LaTeX Workshop 允许调用外部 PDF 阅读器. 这里提供一份在 Windows 系统内调用外部 SumatraPDF 阅读器的设置方法, 虽然 LaTeX Workshop 官方的 Wiki 已声明不再支持外部 PDF 阅读器, 但这套方法依然可以使用.

首先在 VS Code 设置的 `json` 文件内写入

```
"latex-workshop.view.pdf.viewer": "external",
"latex-workshop.view.pdf.ref.viewer": "external",
"latex-workshop.view.pdf.external.viewer.command": "<SumatraPDFPath>/SumatraPDF.exe",
"latex-workshop.view.pdf.external.viewer.args": [
  "-reuse-instance",
  "%PDF%"
],
"latex-workshop.view.pdf.external.synctex.command": "<SumatraPDFPath>/SumatraPDF.exe",
"latex-workshop.view.pdf.external.synctex.args": [
  "-reuse-instance",
  "-forward-search",
```

```
  "%TEX%",
  "%LINE%",
  "%PDF%"
]
```

实际使用时, 用户需要结合自身安装情况更换 <SumatraPDFPath>, 即 SumatraPDF 的安装路径.

接下来设置 SumatraPDF. 在菜单栏 Settings 》Advanced Options 中找到并设置其键值

```
EnableTeXEnhancements = true
ReuseInstance = true
ReloadModifiedDocuments = true
InverseSearchCmdLine = wscript.exe "<SumatraPDFPath>\sumatra-inverse.vbs" "%f" %l
```

并在 <SumatraPDFPath> 内写入 sumatra-inverse.vbs 文件, 内容是

```
Set WshShell = CreateObject("WScript.Shell")
If WScript.Arguments.Count < 2 Then WScript.Quit
filePath = WScript.Arguments(0)
lineNum = WScript.Arguments(1)
gotoArg = Chr(34) & filePath & ":" & lineNum & Chr(34)
cmdLine = Chr(34) & "<VSCodePath>\bin\code.cmd" & Chr(34) & " -r -g " & gotoArg
WshShell.Run cmdLine, 0, False
```

这里 <VSCodePath> 是 VS Code 的安装路径.

B.5 迁移 WSL 的安装位置

考虑到一部分用户的系统盘空间有限, 结合微软官方文档和 StackExchange, 可以考虑迁移 WSL 的安装位置到其他盘. 在操作前首先执行以下 cmd 命令, 保证 WSL 已经关闭

```
wsl --shutdown
```

第一步, 建立新的位置. 假设新位置在 X:\ 盘. 在 cmd 中执行

```
wsl -l
```

可以查到当前已经安装的 WSL 分发, 例如

```
适用于 Linux 的 Windows 子系统分发:
Ubuntu (默认)
```

在 bash 中执行

```
lsb_release -a
```

查到实际 WSL 内 Ubuntu 的版本, 例如

```
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 24.04.2 LTS
Release: 24.04
```

```
Codename: noble
```

然后在 cmd 中执行

```
mkdir X:\WSL\ubuntu2404
```

创建新的位置. 这里起名 ubuntu2404 是因为 Ubuntu 实际版本为 24.04, 也可以起其他名字.

第二步, 将现有的 Linux 发行版打包成一个虚拟硬盘 (VHDX) 文件. 在 cmd 中执行

```
wsl --export Ubuntu X:\WSL\ubuntu2404\ext4.vhdx --format vhd
```

这里的 Ubuntu 指的是系统默认的 WSL 分发, 这条命令将原本的分发打包为一个.vhdx 文件.

第三步, 删除旧分发. 需要注意的是, 此步骤是破坏性的, 它会删除原分发及所有数据. 执行这一步前, 务必确认已经生成的.vhdx 文件已成功保存. 在 cmd 中执行

```
wsl --unregister Ubuntu
```

就将旧分发 Ubuntu 删除了. 这时在 cmd 中执行

```
wsl -l
```

应该看不到任何 Ubuntu 的分发内容.

第四步, 导入新分发. 将保存的.vhdx 文件重新注册回 WSL 系统, 在 cmd 中执行

```
wsl --import-in-place Ubuntu X:\WSL\ubuntu2404\ext4.vhdx
```

这时在 cmd 中执行

```
wsl -l
```

将看到

```
适用于 Linux 的 Windows 子系统分发:
Ubuntu (默认)
```

至此迁移成功.

实际上, 将整个 WSL 的安装位置迁移还有一个好处. 根据微软官方文档, 将项目文件直接存储在 WSL 的驱动器上, 性能速度会提高.

附录 C　离线安装宏包

由于一些原因, 有些电脑不能联网, 这对升级宏包不利. 这里介绍离线升级标准 TEX Live 的包 (pkg) 的方法, 主要是用 `tlmgr install` 的`--file` 参数来实现.

首先在能够联网的电脑上访问 archive 页面, 下载 `<archiveName>.tar.xz` 文件. 文件列表很长, 加载需花费些时间. 注意 `<archiveName>` 未必是 pkg 上的 `<pkgName>`, 例如在 pkg 上的 lshort-zh-cn 对应着 archive 上的 `lshort-chinese.tar.xz` 和 `lshort-chinese.doc.tar.xz`.

这里多解释一点 `<archiveName>`. 前面已经看到一个 `<pkgName>` 可能会对应多个 `<archiveName>`, 但基本上 `<archiveName>` 的命名规则是 `<xxx>`, `<xxx.source>` 和 `<xxx.doc>`. `<xxx>` 是必装的宏包文件; `<xxx.source>` 是选择安装的源码, 如 `dtx` 文件; `<xxx.doc>` 是选择安装的文档, 如 `pdf` 文件等. 这三者未必同时存在, 例如前面提到的 lshort-chinese.

另外, 如果需要安装的是一个可执行文件, 例如 `pdftex` 或 `xetex`, 那么会涉及到根据操作系统进行下载的相关判断. 如果要升级本地已安装的可执行文件, 可以在命令行输入以下语句查询:

Windows 系统

```
tlmgr info --only-installed | findstr "pdftex"
```

Ubuntu 和 macOS

```
tlmgr info --only-installed | grep 'pdftex'
```

在输出结果中可看到带系统信息的名称. 如果是升级本地未安装的可执行文件, 那么用户就需要根据经验自行判断.

下载正确的 `<archiveName>.tar.xz` 后, 可先检查一下该压缩文件中是否包含了 `tlpobj` 文件. 之后将压缩文件拷贝到未联网电脑上, 在命令行执行

```
tlmgr install --file <archiveName>.tar.gz
```

系统便可自行安装.

对于部分提供了 `l3build` 脚本 `build.lua` 文件的宏包, 用户可在下载压缩文件后解压, 在解压后的目录中执行

```
l3build install
```

系统便可自行安装.

附录 D　版本信息

v2026.11.1

- 更新 Ubuntu 26.04 章
- 简化 WSL 章
- 设置 AppImage 为默认 tex 编辑器
- 调整 Windows 章
- 调整 SumatraPDF 设置

v2026.10.1

- 增加镜像网址
- 恢复代码环境不复制行号
- 微调 VS Code 设置
- 修正 SumatraPDF 设置
- 调整一些排版细节

v2026.9.1

- Overleaf 升级为 2026 版

v2026.7.1

- 添加仅限单位内部访问的源
- 更新 VS Code 外部调用 SumatraPDF 的设置

v2026.5.1

- 更新 WSL 迁移内容

v2026.4.1

- 更新 WSL 的 SumatraPDF 阅读器设置
- 更新 Ubuntu 换源配置
- 增加磁盘空间确认内容

v2026.3.3

- 更新 TeX Live 2026

v2026.3.1

- 调整国内在线平台介绍
- 修订 MacTeX 的安装

v2026.2.1

- 调整国内镜像网站
- 纠正笔误
- 更正 Ubuntu 系统卸载命令

v2026.1.1

- 增加了国内镜像网站
- 适应 Overleaf 新版界面调整了介绍

v2025.11.1

- 调整 VS Code 的配置说明
- 修订 Ubuntu 下 `tlmgr` 使用权限问题

v2025.10.1

- Overleaf 升级为 2025 版
- 更新 WSL 镜像挂载

v2025.9.1

- 修订 Ubuntu 系统下字体安装内容
- 修订 Ubuntu 下 `tlmgr` 使用权限内容

v2025.8.1

- 更正- `shell-escape` 范例
- 整合在线 LaTeX 平台内容

v2025.6.1

- 更正 macOS 中通过 Homebrew 安装 TeXstudio 的命令
- 订正拼写细节错误
- 更新一处失效的博客链接
- 更新 `build.lua`
- 增加联合高校镜像源说明以及地址

v2025.5.1

- 添加了 Homebrew 的快速安装脚本
- 更改了手册的代码高亮样式
- 订正拼写细节错误

v2025.4.1

- 更正 WSL 中的错误
- 添加山东大学镜像
- 解释 MacTeX 字体问题
- 增加 `Cygwin` 的提示
- 增加 MacTeX 命令行卸载相关内容
- 删除了示例中的 `UTF8` 选项
- 给出 `build.lua`
- 更新商用编辑器的最新价格
- 添加了 `l3build` 的离线安装方式
- 调整大陆镜像地址
- 新增 LoongTeX 在线平台
- 更新捐赠二维码

v2025.3.10

- 更新 TeX Live 2025

v2025.3.1

- 增加了开启 `cmd` 界面的介绍

v2025.1.1

- 强调了记录 Ubuntu 安装时用的 `x86_64`
- 将卸载和跨版本升级内容调整至章末
- 增加 TeXStudio PDF 阅读器内容

v2024.9.1

- Overleaf 升级为 2024 版

v2024.6.1

- 添加镜像
- 修改对于不安装第三方安全软件的建议

v2024.4.1

- 添加 TeXstudio 的 AppImage 内容
- 调整编辑器售价
- 调整大陆镜像地址
- 增加捐赠渠道

v2024.3.15

- 更新 TeX Live 2024

v2024.1.1

- 订正前言超链接

v2023.12.1

- 订正 Ubuntu 系统下环境变量生效的内容
- 修订 Windows 中文用户名的一处年份错误

v2023.11.1

- 更新 MacTeX 安装内容 (感谢 tauyoung 起草)

v2023.10.1

- Overleaf 升级为 2023 版

v2023.9.1

- Ubuntu 下 fontconfig 路径问题

v2023.7.1

- 补充在线平台

v2023.6.1

- 在 PDF 书签中显示章节序号

v2023.5.1

- 更新 Ubuntu 环境变量和字体安装
- 添加 TUG 的编辑器列表

v2023.4.1

- 重构有关 Ubuntu 和 WSL 的内容
- 补北京大学和华为云镜像
- 订正 Windows 系统下安装目录的错误

v2023.3.23

- 更新 TeX Live 2023 相关内容
- 更改有关编辑器的内容
- 添加一些可以直接用包管理器安装字体的信息

v2023.1.1

- 订正拼写细节错误
- 更新大陆可用源列表
- 更新有关 TeXstudio 部分设置

v2022.10.1

- Overleaf 添加 Stop on first error 功能

v2022.9.1

- Overleaf 升级至 TeX Live 2022

v2022.8.1

- 补充 MacTeX 卸载和跨版本升级的一些内容
- 添加了一个编译参数的例子
- 删除了失效的国内镜像

v2022.7.1

- 补充 TeXPage 相关内容

v2022.5.1

- Windows、Ubuntu 和 WSL 下安装 TEX Live 2022 的内容更新
- 统一挂载、加载为装载, 与现行 Windows 系统名称统一

v2022.4.1

- 根据 Dongsheng Deng 提供的内容, 修改 macOS 安装文本

v2022.3.1

- 更改 TEXworks 的编译命令设置

v2021.12.1

- TEXstudio Qt6 版本在 Windows 系统上存在问题

v2021.11.1

- Overleaf 升级为 TEX Live 2021

v2021.10.1

- 添加 ubuntu 和 WSL 在用户文件夹安装的相关内容

v2021.8.1

- 改动一些文本上的小问题
- 改动错误的链接
- 添加南方科技大学镜像
- 添加 Windows 10 系统内注册表位置

v2021.7.3

- 添加 Windows 中文用户名处理方法的提示
- 添加 TEXworks 拼写检查文件夹建立的提示

v2021.5.1

- 添加 Windows 中文用户名导致安装失败的解决方法
- 添加升级宏包的必要性
- 更正两处超链接

v2021.4.7

- 更新安装 TEX Live 2021 内容
- 添加东莞理工学院源
- 补充 TEXstudio 下载方法
- 补充 WSL 中使用 `texdoc` 的方法

v2021.2.1

- 更新 `l3backend-*` 解决方案
- 补充调整安装路径后的操作
- 调整大陆镜像地址

v2020.12.1

- 更新哈工大源
- 添加 CTAN 镜像列表和连接情况网址
- 调整 Ubuntu 和 WSL 字体相关内容

v2020.11.1

- Overleaf 后台更新为 TEX Live 2020
- 补充卸载方法
- 补充 PPA 反向代理安装 TEXStudio 的方法

v2020.10.1

- 增加北京外国语大学镜像链接
- 修正 Windows 10 中用 `sudo` 命令的错误
- 完善 Windows 下手动删除涉及的注册表位置

v2020.8.2

- 增加验证 sha512 的内容
- 将本文提交至 CTAN 而进行必要更改
- 利用 `texdoc` 调出本手册
- 给出直接下载 MacTEX 的链接

v2020.7.1

- TEXworks 拼写检查调整
- VS Code 是开源软件
- 改摘要为前言并增加内容
- 增加调出宏包手册内容

v2020.6.1

- 重述大陆地区的源的相关内容
- 为适应 Ubuntu 20.04 调整某些内容
- 调整有关管理员权限的内容
- 调整有关编辑器的内容
- 借助网站加速下载 TeXstudio
- 提示卸载时的权限
- 给出 `make.bat` 与 `makefile`
- 增加 MD5 的说明
- 更正设置主文档的错误

v2020.5.1

- 更新 MiKTEX 相关网址
- 更新至 TEX Live 2020
- 更新 Windows 10 手动卸载 TEX Live 的内容
- 调整 TEXstudio 图标大小
- 更新镜像网址
- 更改 OpenOffice 网址
- 更新 Windows 10 升级宏包描述
- 更新 Overleaf 的介绍

v2020.4.1

- 简要介绍 WSL 辅助程序
- 添加 Overleaf 学习与帮助
- 添加两处与 WSL 有关的内容
- 增加几处软件菜单栏的讲解
- 更清晰查看 Windows 10 环境变量的代码
- Windows 10 直接以管理员身份打开命令行安装
- 调整标点

v2020.3.1

- 摘要中明确提到 CTEX 套装和发行版、编辑器
- 更改 Windows 10 安装 TEX Live 的描述, 如路径名, 文件扩展名等

- 将文件名要求写入注释
- 添加 TeXstudio 数学符号表和插入、改写调整
- 解决 LaTeX 2$_\varepsilon$ 版本不匹配导致 `xelatex` 失败的问题
- 添加打开 cmd 的方法
- 改写 TeXstudio 显示行号的设置
- 更新 MiKTeX 相关网址
- 添加调整 TeXworks 字体的内容
- 添加 Windows 中手动卸载 TeX Live 的方法
- 添加 Overleaf 中文网址

v2020.2.1

- 添加版本信息列表
- 改变 Windows 10 的 TeXstudio 的下载网址
- 添加 macOS 中 `xelatex` 用字体名调用字体的方法
- 详述 WSL 中不懂 vim 的处理方法和阅读宏包手册的方法
- 强调路径为不带空格的英文
- 订正若干拼写错误
- 给出 `jdk` 环境变量的处理
- 添加北京交通大学镜像
- 注明安装需卸载国产安全软件
- 更正 TeXworks 自动补全的教程网址
- 给出更换 2345 好压的建议
- 补充 Overleaf 升级内容