Latex 入门

陈昊天

2019 年 4 月 26 日

目录

1 环境配置

1.1 texlive 的安装

texlive 是进行 Tex 排版最关键的软件之一，其作用相当于 sdk，是编译和运行 Latex 必须的。你可以转到 **texlive** 官网 (http://tug.org/texlive/) 进行下载，但这种方式并不是很推荐。推荐使用国内的镜像源下载其开源的 iso 镜像。例如清华开源镜像站 (https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/) 下载 texlive.iso 文件。下载完成 texlive.iso 文件后，使用虚拟光驱软件（或者系统自带的打开），用管理员运行里面的文件install-tl-advanced.bat，之后可以打开N. of collections 选项，去掉不经常使用的语言包（建议保留中文和英文）和 texworks 编辑器，具体操作方法请参考教程。[1]

1.2 vscode 环境配置

网上有很多 Latex 编辑器可以选择，但是本文主要讲解在 vscode 上进行环境配置的方法。打开 vscode，在扩展商店中搜索Latex Workshop，进行安装，基本上只要这么一个扩展就足够使用了。接下来最关键的一步是修改 vscode 的环境配置。首先按下快捷键 Ctrl + Shift + P，选择Open Settings(Json) 选项，用附带的文件 settings.json 覆盖原来的配置。覆盖方式为将第一个括号内的内容复制到设置的第一个括号内结尾处，然后删掉之前重复的设置项，然后保存[2]。

1.3 开始第一个 Latex 项目

由于.tex 文件编译和运行时需要生成一系列文件，建议专门创建一个文件夹进行文档编写。首先创建一个文件夹 Hello，然后新建一个文件 hello.tex，在文件内输入下列文本。

```
\documentclass{article}
\begin{document}
    Hello world!
\end{document}
```

[1] 参考自知乎教程**最新** TeXLive **环境的安装与配置** (https://zhuanlan.zhihu.com/p/64555335)

[2] 参考自知乎问题回答**有哪些好的** LaTeX **编辑器**？(https://www.zhihu.com/question/19954023/answer/676316395)

代码文件见*./texs/hello1.tex*

之后选择选项卡 Latex->commands->Build Latex project->Recipe:xelatex，进行编译。然后点击上方 View Latex PDF File，稍等便可将会在右侧显示编译好的 Latex 文件。

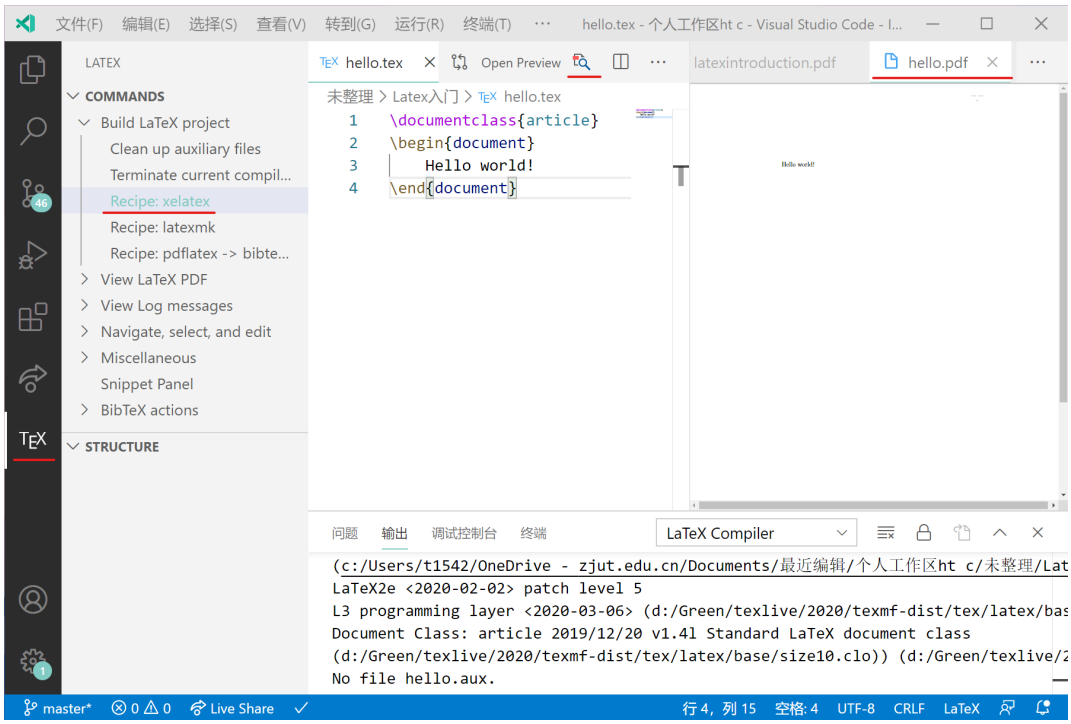

图 1: 编写 Hello.tex 文件并编译

学习一门新的技能，最麻烦的地方在开头的环境配置，只要你能够在 vscode 正确编译并运行，那么你之后遇到的麻烦将会少很多，因为有啥问题查看错误报告。下面将会在此基础上讲解 Latex 的各种特性。

2 Latex 宏

2.1 Latex 宏命令

Latex 到处充满了宏命令，宏也是 Latex 中最核心的内容，一般宏分为命令宏，单行宏，段落宏[3]，命令分别如下。

```
% 单行宏，其中option是可以省略的，称为选项
\command[option]{content}
% 例如，当你需要对某一段内容加粗时，可以这么写。
这是没有加粗的内容\textbf{这是加粗的内容}
% 例如，此宏将会读取当前文本的宽度，常用来设置参数时使用
\textwidth

% 段落宏，其中option时可以省略的，称为选项
```

[3]这里的宏分类并不是严谨的，这里按照宏实现的功能进行分类，是为了便于理解。

```
\begin{command}[option]
    ...content
\end{command}
% 例如，当你需要插入一张图片时，可以这么写
% 其中\label命令将会在引用章节说明
\begin{figure}[H]
    \centering
    \includegraphics[option]{图片的路径}
    \caption{标签的内容}
    \label{Label的内容}
\end{figure}
```

代码文件见*./texs/tutoral1.tex*

熟练的使用宏命令可以让 Latex 文档的编写更加轻松，其中最基本的宏命令是关于排版的命令，将会在排版章节讲解。

2.2 Latex 环境

有时候我们会提到环境这个词，这个其实我们在写文章的时候就会不自觉的用到。有时候，基于内容的不同便会使用不同的环境。

```
\documentclass{article}
\begin{document}
    Hello world!
\end{document}
```

代码文件见*./texs/hello1.tex*

例如上面最简单的一个文档（应该没有比这更简单的了）第 2-4 行之间的内容就属于 document 环境。这个时候代码的执行和渲染按照 document 来执行。之后使用到的插入表格和图片均会使用到相应的 Latex 环境。

2.3 Latex 宏包

原生的 Latex 支持的内容并不是很多，最简单的一点，就是 Latex 不支持中文，因此，要编写排版优雅，具有一定美观性的文档肯定离不开**宏包**，有些包是官方给出，也有些是第三方开发的，这些宏包组合起来，可以实现各种各样的功能。

2.3.1 导入宏

有时候你在写一句宏指令时会发现没有自动提示或者编译时出错，往往是没有导入正确的宏的结果，导入宏用到命令是 \usepackage{packagename}，此命令理论上可以放置在任意的位置，但推荐放在 document 区域之前，例如下文的代码。如果把中文排版的删掉，编译后并不会出错，但其中的中文字符将不会显示。

```
\documentclass{article}
% 中文排版
\usepackage{ctex}
```

```
\begin{document}
    Hello world!
    你好，世界。
\end{document}
```

代码文件见*./texs/hello2.tex*

2.3.2 常用的宏包

其中，在中文编辑中，ctex 宏包是最重要的一个宏包[4]，而且其已经形成了一个完整的生态。可以在 CTAN 相关的网站进行查看。例如 CTEX 在线文档 (http://www.ctex.org/documents/packages/)。

下面将列出经常使用到的宏包，某些常用功能的使用将会在后续的教程中编写。

宏包	功能	功能说明
ctex	tex 的中文发行版	如果文档是中文的，则必须使用此宏包。
inputenc	输入的编码	用于控制输入的编码，在 ctex 中一般不会使用，用于解决西文特殊符号导致编码错误的问题。
geometry	页面布局	用于设置页边距，页眉和页脚的位置等。(一般默认的页边距太大了。)
multicol	分栏	用于对文章的特殊区域进行分栏的操作。
float	设置浮动元素	可以自定义浮动样式，使部分元素脱离正确的文档流位置。
caption	标签	用于设置表格、图片和其他内容的标签
graphicx	插入图像扩展版本	用于插入图像
subfigure	插入子图	用于在一个区域内插入多张图片，常常和 graphicx 一起使用。
amsmath amssymb	数学公式	扩用于扩展 Latex 的公式系统，以提供更多中公式的显示。
natbib	参考文献	扩展 Latex 的参考文献功能。
hyperref	超链接，内部连接	在文档中插入超链接
bookmark	pdf 书签系统	用于在 pdf 中加入书签系统，以便于浏览与翻阅，包含了 hyperref 的功能。请在 hyperref 后声明。
soul	文本修饰	提供了多种文本修饰的效果
listings	代码块	支持在 Latex 中插入代码和高亮代码
longtable	长表格	用于在 Latex 中制作跨页的表格（一般 Latex 的表格不支持跨页）

表 1: 常用的宏包

[4]ctex 不仅仅使 latex 支持中文，而且加了很多的功能，让 Latex 排版符合中国人的习惯。

3 Latex 基本语法

3.1 内容与排版

3.1.1 主标题

Latex 默认提供了快速编写标题的命令，标题样式是默认的，如果需要自定义标题，需要使用其他的宏包，但是一般默认的已经够了。

```
\documentclass{article}
% 中文排版
\usepackage{ctex}

% 下面三行代码将定义元数据
\title{Hello}
\author{Author}
\date{2020年4月26日}

\begin{document}
    % 将标题放在文档的开头位置
    \maketitle
    Hello world!

    你好，世界。
\end{document}
```

代码文件见*./texs/hello3.tex*

3.1.2 标题与内容

Latex 默认支持多级标题与内容，其中 \section{section-title} 将会插入一级标题。\subsection{subsection-title} 将会插入二级标题。\subsubsection{subsubsection-title} 将会插入三级标题。例如下列代码。

```
\documentclass{article}
% 中文排版
\usepackage{ctex}

% 下面三行代码将定义元数据
\title{Hello}
\author{Author}
\date{2020年4月26日}

\begin{document}
    % 将标题放在文档的开头位置
    \maketitle
    Hello world!

    你好，世界。
```

```
    \section{一级标题1}
        \subsection{二级标题}
            \subsubsection{三级标题}
                位于三级标题内部的内容
    \section{一级标题2}
        位于一级标题内部的内容
\end{document}
```

代码文件见*./texs/hello4.tex*

图 2: 标题与内容效果图

其他关于段落有一篇知乎文章 **LaTeX 入门 (五)——段落** (https://zhuanlan.zhihu.com/p/56127244) 写得很好。

3.1.3 目录

Latex 支持自动导出目录，只需要加入 \tableofcontents 语句然后编译即可[5]。

```
\documentclass{article}
% 中文排版
\usepackage{ctex}

% 下面三行代码将定义元数据
```

[5]由于 Latex 的相关原因，编译一次可能并没有效果，需要编译两次及以上，这个问题将在后面讨论

```
\title{Hello}
\author{Author}
\date{2020年4月26日}

\begin{document}
    % 将标题放在文档的开头位置
    \maketitle
    \tableofcontents
    Hello world!

    你好，世界。
    \section{一级标题1}
        \subsection{二级标题}
            \subsubsection{三级标题}
                位于三级标题内部的内容
    \section{一级标题2}
        位于一级标题内部的内容
\end{document}
```

Hello

Author

2020 年 4 月 26 日

目录

Hello world!

你好，世界。

1 一级标题 1

1.1 二级标题

1.1.1 三级标题

位于三级标题内部的内容

2 一级标题 2

位于一级标题内部的内容

图 3: 目录效果图

当然，此时点击目录并没有什么效果，如果要在目录上生成跳转链接，只需要导入 hyperref 宏包即可，具体步骤请看在文档内部创建锚点与跳转。

当然，当你改变文档的目录结构（例如添加一个 section, 删除一个 section）时，第一次编译时目录并不会跟着刷新，关于此问题请转至常见问题更新后目录没有刷新部分。

3.1.4 文档类型和样式

Latex 为不同的文档类型预置了不同的基础样式，因为文章的内容，排版方式有所不同。有时候我们需要的是一篇短小的文章，有时候我们会用 Latex 编写（当然一般不会这么做。）

Latex 排版系统虽然非常优秀，让编写者可以关注于内容，而不需要费心思关心与排版，然而这是基于 Latex 内置或者在其他样式的基础之上的。因此，要定制 Latex 还是比较麻烦的，这部分属于 Latex 高级语法，一般对于日常使用而言，可以直接套用网上流传的模板，可以减轻很多的工作。

当然，在编写文档时，我们可能还会遇到另外几个问题。这些问题将会在常见问题有所说明。

一般来说，文档类型很多，而且用户可以自定义文档类型。比如一般我们再写论文时第一句就会写上 \documentclass{article}，这个便是声明该文章的样式了。一般来说，常见的有 article,report,book 三种，分别对应短篇幅，中篇幅，长篇幅。

3.1.5 include 命令

include 命令，顾名思义，和 html 文档一样，在一个 Latex 文档内，可以使用 include 命令导入另一个 Latex 文档，这就相当于文档替换的作用。体现了部分的模块设计思想。只要使用 \include{url} 即可使用此功能。

例如，当你需要写一篇很长的书籍时（只有这个合适的例子了），总不可能在一个.tex 文件内写所有的内容，这时候，一般 main.tex 文件就作为主.tex 文件，起着排版和设计样式的作用。

3.1.6 页面布局

这部分在知乎专栏文章 **LaTeX 入门 (七)——页面设置** (https://zhuanlan.zhihu.com/p/56405574) 有详细介绍。

3.1.7 小技巧：快速定位元素

Latex 预览状态下的 pdf 不想 markdown 文档一样，会和源代码同步对齐内容。两者滚动条是分离的。这里有个小技巧。当使用 ctrl+ 鼠标左键点击 pdf 中的文字时，会直接跳转到源码中的相应内容。这对于篇幅过长的文档的元素定义有奇效。

3.2 文字效果

这部分内容需要进一步地更新

3.2.1 常见修饰效果

常见的修改字体修饰效果，例如斜体、粗体、设置字体族可以直接查看知乎文章教程 **LaTeX 入门 (四)——文字** (https://zhuanlan.zhihu.com/p/56069161) 其中常用的命令有设置粗体 \textbf{content}，设置

斜体 \textit{content} 等。

3.2.2 着重号

3.2.3 首字下沉

3.2.4 强调色

3.3 表格与图片

这部分内容需要进一步地更新

3.3.1 传统表格

3.3.2 长表格，多页表格

3.3.3 图片

3.4 书签、引用、链接

3.4.1 创建书签

[6] 当我们导出 pdf 文件再在浏览器上打开后，我们会发现并没有生成目前，这是怎么回事呢。要解决这个问题，我们只要导入 bookmark 包就可以了。此时，你会发现打开 pdf 后有书签了，而且点击目录的相应位置会跳转到文档的相应的位置。当然，有时候我们希望这种链接显眼一些。相关内容请看改变链接的颜色。

```
\documentclass{article}
% 中文排版
\usepackage{ctex}
\usepackage{bookmark}

% 下面三行代码将定义元数据
% \title{Hello}
% \author{Author}
% \date{2020年4月26日}

\begin{document}
    % 将标题放在文档的开头位置
    % \maketitle
    \tableofcontents
    \section{一级标题}
        \subsection{二级标题}
            \subsubsection{三级标题}
                Hello world!

                你好，世界。
\end{document}
```

[6] 参考自 csdn 教程 Latex 下关于 bookmark 包乱码修正的注记https://blog.csdn.net/newdriver2783/article/details/79658056

代码文件见*./texs/hello6.tex*

3.4.2 链接系统

必须导入宏包 \hyperref。

```
\documentclass{article}
\usepackage{ctex}
\usepackage[colorlinks]{hyperref}
\usepackage{bookmark}
\begin{document}
    % 直接编写url，可跳转
    \url{http://www.ctex.org/documents/packages/}

    % 不可以跳转的url
    \nolinkurl{http://www.ctex.org/documents/packages/}

    % 直接编写url，可跳转
    \url{about:blank}

    % 内容指向一个url
    % 只有url是一个合法的链接才可以跳转
    % 格式为 \href{网址}{内容}
    \href{http://www.ctex.org/documents/packages/}{author1}

    % 创建指向锚点的链接
    \hyperlink{author2}{跳转至内容1}

    中间的内容

    中间的内容

    中间的内容

    中间的内容

    中间的内容

    中间的内容

    中间的内容

    中间的内容

    % 创建锚点，用于文档的定位，当后面的内容为空时指向下一个元素
    \hypertarget{author2}{内容1}

```

```
    内容2
\end{document}
```

代码文件见*./texs/link1.tex*

3.4.3 改变链接的颜色

需要改变的链接，必须要导入宏包 \hyperref。然后可以使用 \hypersetup 进行颜色的定制。

```
\usepackage{hyperref}
\hypersetup{
    colorlinks=true,
    linkcolor=blue,
    filecolor=blue,
    urlcolor=blue,
    citecolor=cyan,
}
```

代码文件见*./texs/linkcolor.tex*

http://www.ctex.org/documents/packages/
http://www.ctex.org/documents/packages/
about:blank
author1
跳转至内容 1
中间的内容
中间的内容
中间的内容
中间的内容
中间的内容
中间的内容
中间的内容
中间的内容
内容 1
内容 2

图 4: 链接效果图

3.5 页面其他元素

这部分代码需要进一步更新

3.5.1 标签 caption

3.5.2 注释

3.5.3 参考文献

3.6 常见问题

3.6.1 更新后目录没有更新

目录（其实就是文章的结构）其实是记录在当前路径与本文件同名的.toc 文件中的，打开就会看到很多看起来向概要的 latex 代码。这便是文章的结构。改变文章的目录结构后第一次编译时，会发生.toc 更新后无法及时呈现在 latex 文档上的问题，这时候多编译几次这个问题就可以得到有效解决。

3.6.2 文档没有显示问题但无法通过编译

关于这一点，我无法给出究竟是什么原因。但在我自己的电脑上，也出现过几次问题。

第一种是使用了某些软件打开了 pdf 文件，导致文件被锁定，无法通过编译。第二种是电脑的内存占用过高，而显示 Memory Exhasted 的警告（看来这个软件确实内存占用挺高的，特别是编译的时候。）

4 代码块

待完善的内容

5 Latex 公式系统

待完善的内容

6 扩展

6.1 颜色

7 自定义样式

太难了，暂时没有打算写。

8 代码系统

其实也没有必要写。

9 测试效果

9.1 代码高亮

下面将测试 python 代码的高亮情况。

```python
def fibonacci_loop(x):
    if x < 0:
        return -1
    if x == 1 or x == 2:
        return 1
    else:
        x1 = 1
        x2 = 1
        x3 = 0
        for i in range(3, x + 1): # iter [2..x]
            x3 = x2 + x1
            x1 = x2
            x2 = x3
        return x3


if __name__ == '__main__':
    print(fibonacci_loop(10000))
```