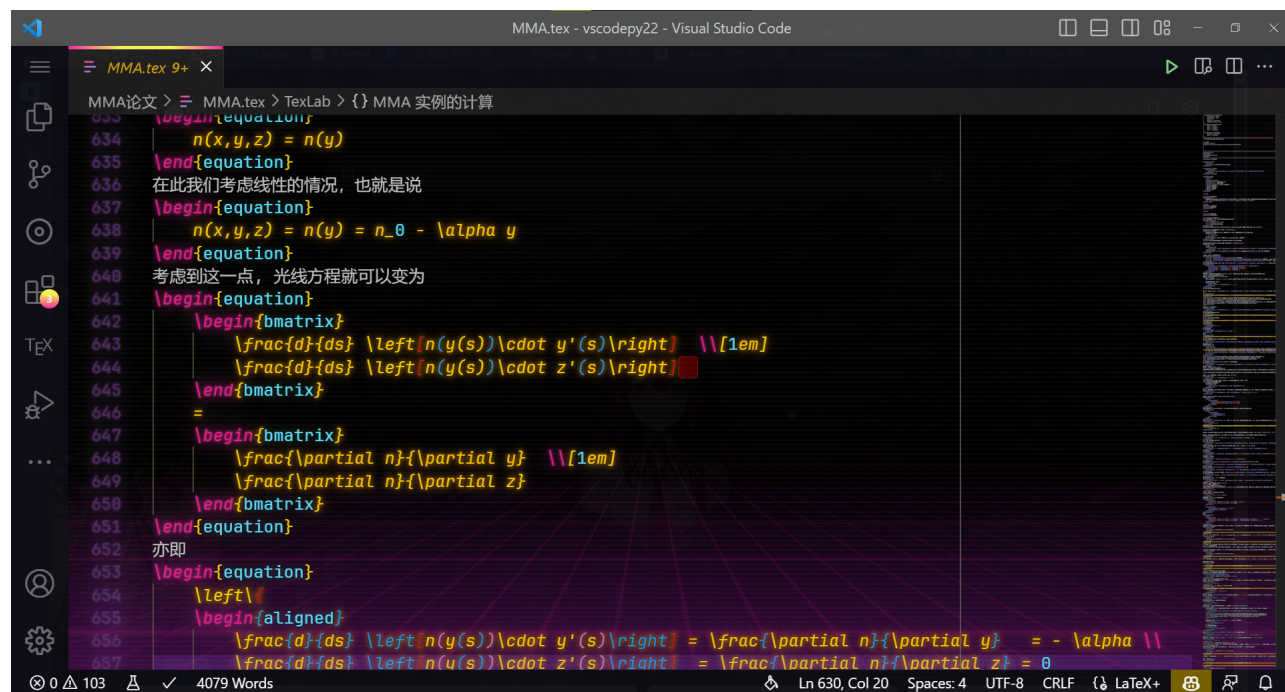


Neovim

一、Neovim 与 vscode

我之前写 latex 或者是其他的一些代码使用的都是 vscode，vscode 使用的门槛低只需要点击即可下载对应的扩展来获得所需要的功能



```
MMA.tex - vscodepy22 - Visual Studio Code
MMA论文 > MMA.tex > TexLab > {} MMA 实例的计算
633 \begin{equation}
634 n(x,y,z) = n(y)
635 \end{equation}
636 在此我们考虑线性的情况，也就是说
637 \begin{equation}
638 n(x,y,z) = n(y) = n_0 - \alpha y
639 \end{equation}
640 考虑到这一点，光线方程就可以变为
641 \begin{equation}
642 \begin{bmatrix}
643 \frac{d}{ds} \left[ n(y(s)) \cdot y'(s) \right] \\
644 \frac{d}{ds} \left[ n(y(s)) \cdot z'(s) \right]
645 \end{bmatrix}
646 =
647 \begin{bmatrix}
648 \frac{\partial n}{\partial y} \\
649 \frac{\partial n}{\partial z}
650 \end{bmatrix}
651 \end{equation}
652 亦即
653 \begin{equation}
654 \left\{
655 \begin{aligned}
656 \frac{d}{ds} \left[ n(y(s)) \cdot y'(s) \right] &= \frac{\partial n}{\partial y} = -\alpha \\
657 \frac{d}{ds} \left[ n(y(s)) \cdot z'(s) \right] &= \frac{\partial n}{\partial z} = 0
\end{aligned}
\right.
\end{equation}
103 4079 Words Ln 630, Col 20 Spaces: 4 UTF-8 CRLF LaTeX+
```

但是在使用的过程中我越来越发现vscode的打开时间很慢，一些补全的生成也会比较慢，从而一定程度上影响了敲 latex 的效率。所以我开始尝试使用 Neovim。

<https://neovim.io/>

Neovim 即使装了 90 个最有名的插件启动时间也就大约 70ms 左右，而且 Neovim 的自定义程度更好，支持更多形式的 snippets，以下是使用 Neovim 写 latex 的一些介绍

[UltiSnips Plugin Guide for LaTeX | Vim and LaTeX Series Part 2 | ejmastnak](#)

[使用 Neovim 和 vimtex 高效撰写 LaTeX 学术论文 - 知乎 \(zhihu.com\)](#)

除此之外 linux 系统中的 latex 编译速度大约是 windows 系统中编译速度的三倍，这也是转而去选择 linux 系统中 Neovim 的一个原因。

二、vim 与 Neovim

vim 是 linux 系统默认的文本编辑器，而 Neovim 可以视为其升级，Neovim 的优势体现在：

1. Neovim 的配置文件采用的是 lua，lua 代码支持模块化管理，同时 lua 写的代码运行起来更快
2. 许多插件现在都更支持 Neovim
3. Neovim 更加开源和现代化

三、Linux 系统的配置

我使用的是 windows 系统，我采用的是 wsl

安装 [WSL | Microsoft Learn](#)

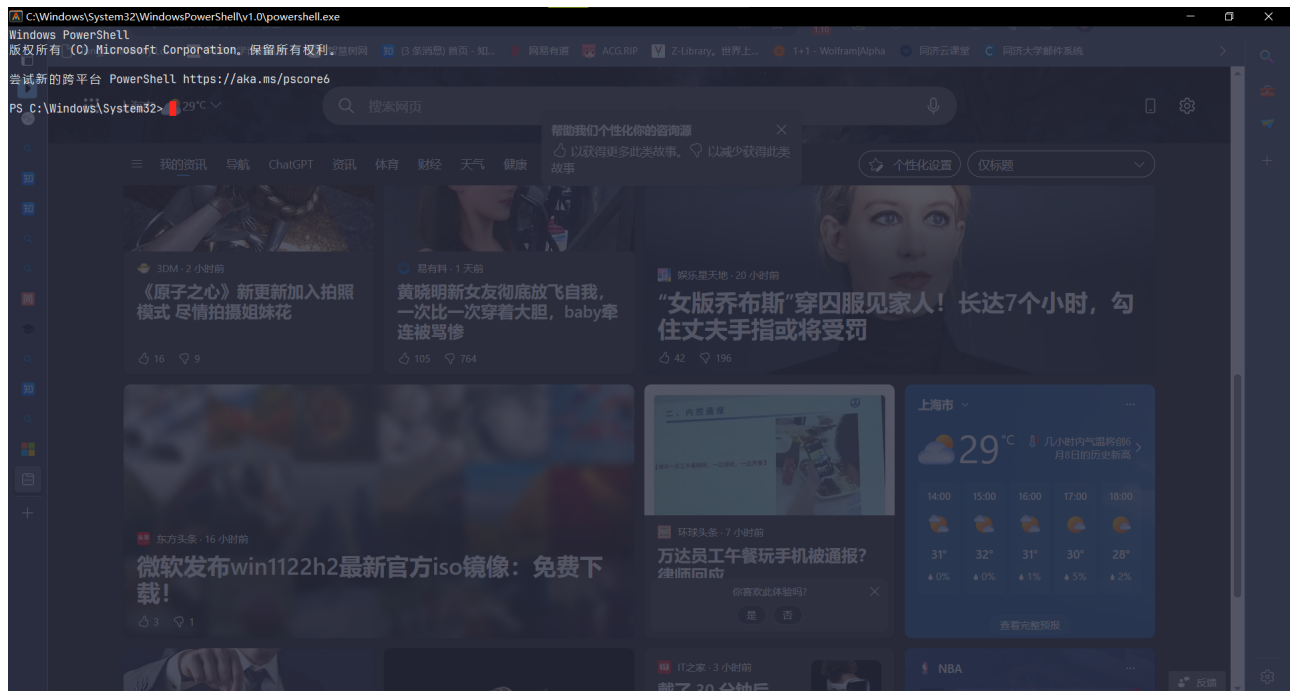
来使用 linux 系统。

四、终端的选择

我一开始使用的是 Windows Terminal 终端，但是我后来使用了 alacritty

<https://github.com/alacritty/alacritty>

这是一款使用 GPU 加速的终端，其号称是最快的终端，通过配置文件能够设置一些按键的映射，字体的选择，背景的透明度，字体的大小等等



这里我设置的背景透明度是 0.8，字体设置是 JetBrainsMono Nerd Font，可以在这里下载到喜欢的字体

[Nerd Fonts - Iconic font aggregator, glyphs/icons collection, & fonts patcher](#)

Nerd Fonts 就是支持各种图标的字体。

五、终端的美化

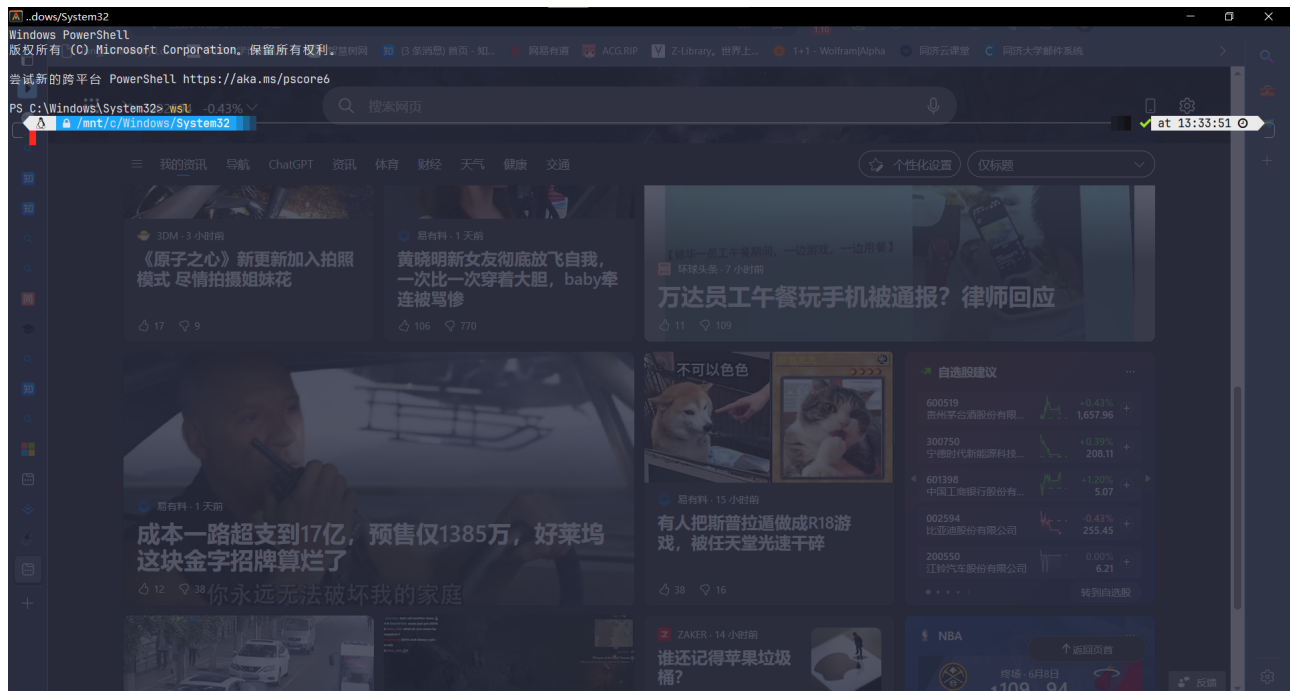
首先选择 zsh 作为解释器, 然后安装 oh-my-zsh

<https://github.com/ohmyzsh/ohmyzsh>

之后选择主题为 powerlevel10k

<https://github.com/romkatv/powerlevel10k>

即可达成这样的效果



值得注意的是 `oh-my-zsh` 兼容了许多插件，可以通过插件来增加使用终端的体验（比如自动检查命令是否正确的插件）

六、Neovim 的配置学习

我学习是通过 Nshen 大佬写的

<https://link.zhihu.com/?target=https%3A//juejin.cn/book/7051157342770954277>

进行的学习，通过学习可以大致了解如何搭建 Neovim。我这里只作一些文件结构的简单介绍

```
├─ LICENSE
├─ README.md
├─ init.lua
├─ lua
│   └─ autocmds.lua
│   └─ basic.lua
│   └─ colorscheme.lua
│   └─ keybindings.lua
│   └─ lsp
│       └─ cmp.lua
│       └─ config
│           └─ bash.lua
```

```
| | └─ emmet.lua
| | └─ html.lua
| | └─ json.lua
| | └─ lua.lua
| | └─ markdown.lua
| | └─ pyright.lua
| | └─ rust.lua
| | └─ ts.lua
| └─ formatter.lua
| └─ null-ls.lua
| └─ setup.lua
| └─ ui.lua
└─ plugin-config
| └─ bufferline.lua
| └─ comment.lua
| └─ dashboard.lua
| └─ gitsigns.lua
| └─ indent-blankline.lua
| └─ lua-line.lua
| └─ nvim-autopairs.lua
| └─ nvim-tree.lua
| └─ nvim-treesitter.lua
| └─ project.lua
| └─ surround.lua
| └─ telescope.lua
| └─ toggleterm.lua
| └─ vimspector.lua
| └─ which-key.lua
└─ plugins.lua
└─ utils
    └─ fix-yank.lua
    └─ global.lua
    └─ im-select.lua
```

上述文件都是放置于 `nvim` 文件夹之中，一般来说可以在 `.config` 中找到 `nvim` 文件夹。可以看到每个 `lua` 文件都是一个模块，这里展示的模块很多是因为每个插件都写了对应的模块（对没错，需要自己写！）以下介绍一些重要的模块

1. `init.lua` 最重要的模块，Neovim 启动的时候会先在 `nvim` 文件夹中找这个 `init.lua`，然后 `init.lua` 中如果调用了别的模块的话，才会进一步去找别的模块的 `lua` 文件，一般来说可以通过

```
require("")
```

的指令来调用别的模块。

2. `lua` 文件夹就是用来放置其余 `lua` 文件的地方
3. `basic.lua` 是用于设置一些基本的指令，比如是否显示行数，是否高亮光标所在行等等
4. `colorscheme.lua` 是用于调用主题的
5. `keybindings.lua` 是用于设置键盘映射的
6. `lsp` (Language Server Protocol) 是用于提供补全的一套系统在 Nshen 写的教程中有详细介绍
7. `plugin-config` 中放了插件的配置模块.
8. `plugins.lua` 是用于下载插件的模块，这里用的是 `packer` 进行的下载.

七、最终的 Neovim

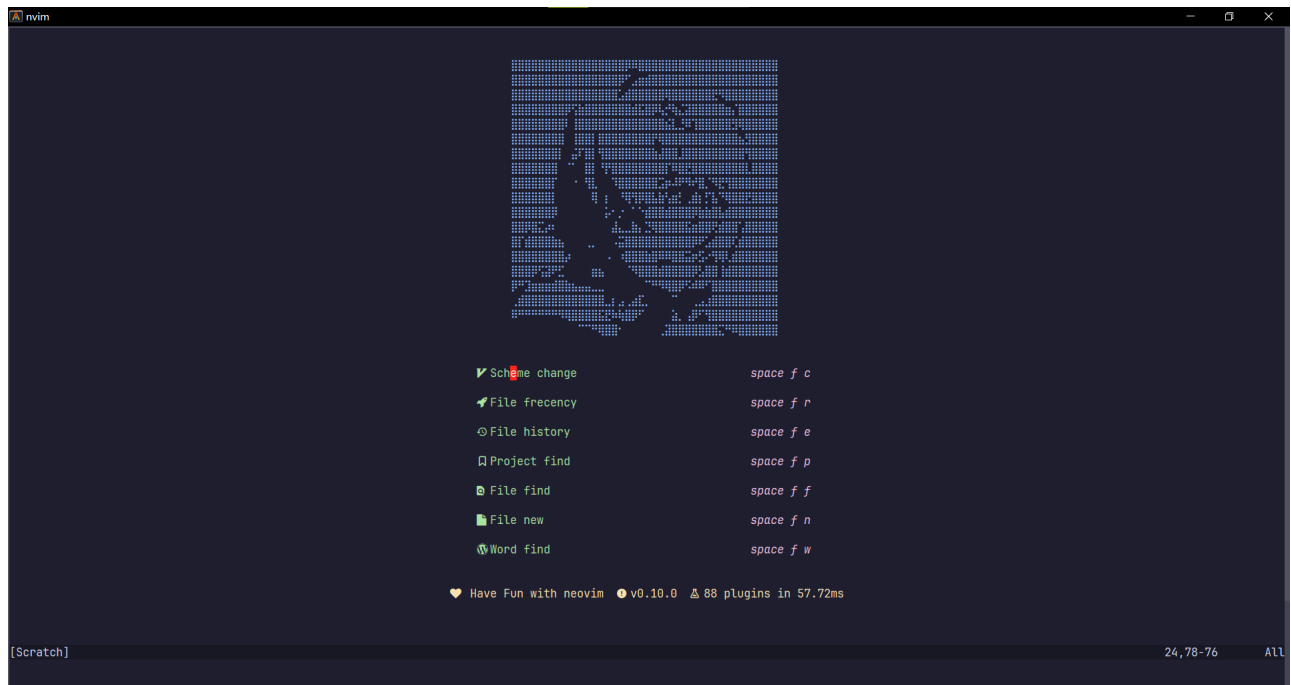
事实上自己写一整套 Neovim 非常的累！而且不是专业的人，很容易报错！我的建议以及我最终采取的选择是：在学习完 Neovim 的配置之后，使用别人的配置，这里强推 `ayamir` 的配置

<https://github.com/ayamir/nvimdots>

网页里有对应的 `wiki`，介绍的非常详细。

八、最终的成果

实际上一些细节可以在上述 `wiki` 中找到，以下是我目前 Neovim 主页的截图



九、一些补充

上述主页中的图片是可以更改的，我这里使用的是 **ayamir** 的默认配置，更换的方法是在对应的 **lua** 文件中将点图输入即可，将一般的图片导出为点图的方式可以参考：

<https://github.com/TheZoraiz/ascii-image-converter>

安装后使用

```
ascii-image-converter [file] -b -w 40 > a.txt
```

即可

十、Copilot

Neovim 支持 copilot 参考

<https://github.com/github/copilot.vim>

即可。