

实验报告二 1

24 计算机科学与技术吴虹霖 学号: 24020007135

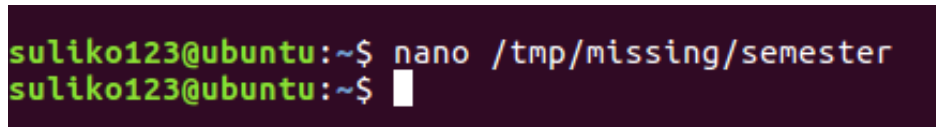
2025 年 9 月 16 日

1 一、Shell 命令

实例 1.1 nano 文本编辑器写入内容

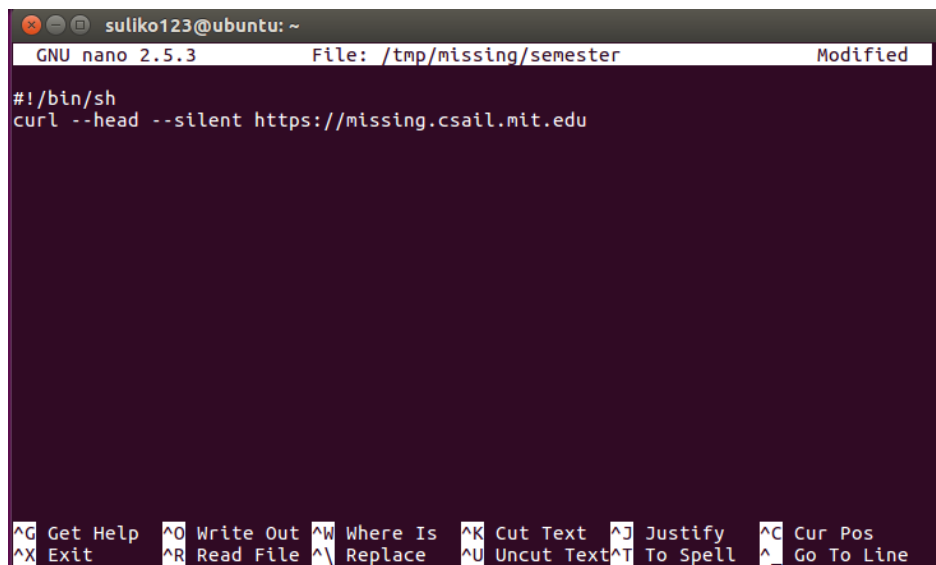
Q: 使用 nano 文本编辑器创建并编辑一个文件，写入指定内容。

- A:**
1. 打开终端，输入 nano example.txt 打开 nano 编辑器。
 2. 在编辑器中输入文本内容。
 3. 按下 Ctrl + O 保存文件，确认文件名后按回车。
 4. 按下 Ctrl + X 退出 nano 编辑器。



```
suliko123@ubuntu:~$ nano /tmp/missing/semester
suliko123@ubuntu:~$
```

图 1: nano 文本编辑器写入内容 1



```

suliko123@ubuntu: ~
GNU nano 2.5.3      File: /tmp/missing/semester      Modified
#!/bin/sh
curl --head --silent https://missing.csail.mit.edu

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

图 2: nano 文本编辑器写入内容 2

实例 1.2 touch 新建文件

Q: 使用 touch 命令创建一个名为”semester” 的空文件。

- A:**
1. 在终端中输入以下命令： touch semester
 2. 使用 ls 命令确认文件已创建。

```
suliko123@ubuntu:~$ touch /tmp/missing/semester
suliko123@ubuntu:~$
```

图 3: touch 新建 semester 文件

实例 1.3 查看 touch 程序使用手册

Q: 查看 touch 命令的使用手册。

A: 1. 在终端中输入: man touch
2. 使用方向键浏览, 按 q 退出。

```

suliko123@ubuntu: ~
TOUCH(1)                                User Commands                                TOUCH(1)
NAME
    touch - change file timestamps
SYNOPSIS
    touch [OPTION]... FILE...
DESCRIPTION
    Update the access and modification times of each FILE to the current
    time.

    A FILE argument that does not exist is created empty, unless -c or -h
    is supplied.

    A FILE argument string of - is handled specially and causes touch to
    change the times of the file associated with standard output.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a      change only the access time

Manual page touch(1) line 1 (press h for help or q to quit)
```

图 4: 查看 touch 程序使用手册

实例 1.4 查看当前路径

Q: 查看当前所在目录的绝对路径。

A: 在终端中输入: pwd

```

suliko123@ubuntu:~$ pwd
/home/suliko123
suliko123@ubuntu:~$ S
```

图 5: 查看当前路径

实例 1.5 查看权限

Q: 查看当前目录下文件的详细权限信息。

A: 在终端中输入: ls -l

```
bash: ./semester: Permission denied
suliko123@ubuntu:/tmp/missing$ ls -l semester
-rw-rw-r-- 1 suliko123 suliko123 61 Sep  7 05:25 semester
suliko123@ubuntu:/tmp/missing$
```

图 6: 查看权限

实例 1.6 导航文件夹

Q: 切换至目标目录。

A: 在终端中输入: `cd ..`

```
suliko123@ubuntu:~$ cd /tmp/missing
```

图 7: 导航文件夹

实例 1.7 执行文件

Q: 尝试执行一个文件。

A: 输入文件路径尝试执行, `./`

```
suliko123@ubuntu:/tmp/missing$ ./semester
bash: ./semester: Permission denied
```

图 8: 执行文件

实例 1.8 查看指定目录文件

Q: 查看指定目录下的文件。

A: 使用 `ls`

```
suliko123@ubuntu:~$ ls
Desktop    Downloads  Music      Public     Videos
Documents  examples.desktop  Pictures   Templates
```

图 9: 查看指定目录文件

实例 1.9 添加权限

Q: 为文件添加执行权限。

A: 使用 `chmod` 命令

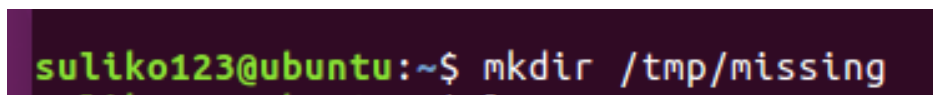
```
suliko123@ubuntu:/tmp/missing$ chmod +x semester
suliko123@ubuntu:/tmp/missing$
```

图 10: 添加权限

实例 1.10 新建文件夹

Q: 创建一个名为"test_dir" 的目录。

A: 使用 mkdir 命令: mkdir test_dir



```
suliko123@ubuntu:~$ mkdir /tmp/missing
```

图 11: 新建文件夹

2 二、Vim 命令

实例 2.1 创建 test 文件

Q: 使用 Vim 创建并编辑一个名为"test" 的文件。

A: 输入 vim test.txt

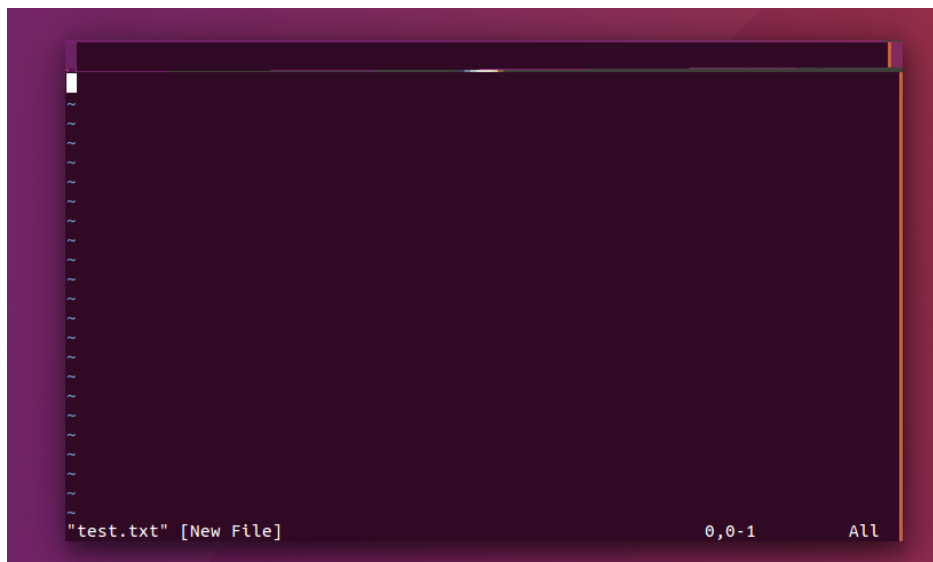


图 12: 创建 test 文件

实例 2.2 查看 vim 版本

Q: 查看当前系统中安装的 Vim 版本。

A: 在终端中输入: vim -version

```
o mode
suliko123@ubuntu:~$ vim --version
VIM - Vi IMproved 7.4 (2013 Aug 10, compiled Oct 13 2020 16:04:38)
Included patches: 1-1689
Extra patches: 8.0.0056
Modified by pkg-vim-maintainers@lists.aliases.debian.org
Compiled by pkg-vim-maintainers@lists.aliases.debian.org
Huge version without GUI.  Features included (+) or not (-):
+acl                +farsi             +mouse_netterm     +tag_binary
+arabic             +file_in_path      +mouse_sgr         +tag_old_static
+autocmd            +find_in_path      -mouse_sysmouse    -tag_any_white
-balloon_eval       +float             +mouse_urxvt       -tcl
-browse             +folding           +mouse_xterm       +termio
++builtin_terms     -footer            +multi_byte        +termresponse
+byte_offset        +fork()            +multi_lang        +textobjects
+channel            +gettext           -mzscheme          +timers
+cindent            -hangul_input      +netbeans_intg     +title
-clientserver       +iconv             +packages          -toolbar
+clipboard          +insert_expand     +path_extra        +user_commands
+cmdline_compl     +job              -perl              +vertsplit
```

图 13: 查看 vim 版本

实例 2.3 保存文档

Q: 在 Vim 中编辑文件后保存并退出。

A: 在命令模式下输入: :wq

[illegible]

图 14: 保存文档

实例 2.4 部分替换

Q: 使用 Vim 替换文件中部分文本内容。

A: 在命令模式下输入: :s/old/new/g

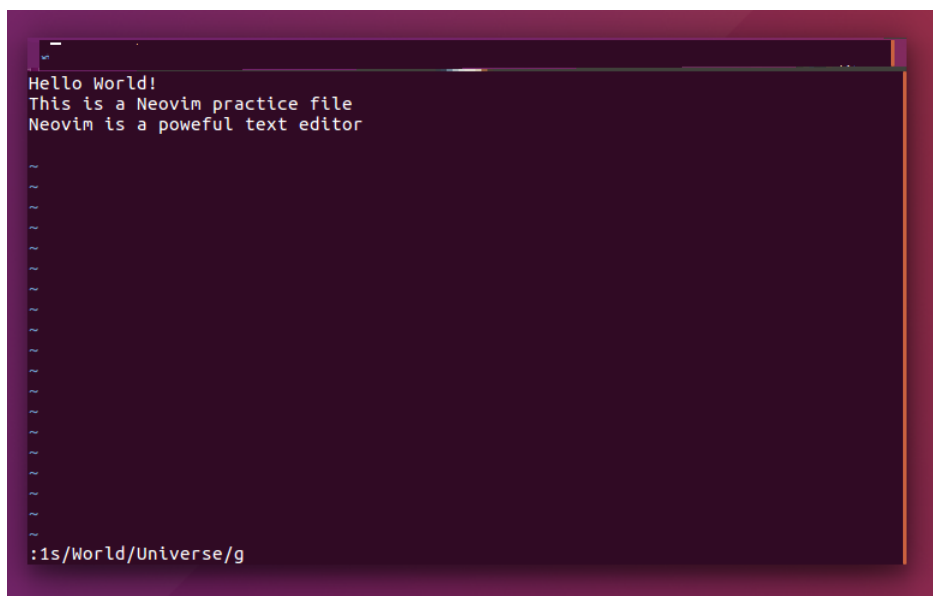


图 15: 部分替换

实例 2.5 查找

Q: 在 Vim 中查找指定字符串。

A: 在命令模式下输入: /要查找的字符串

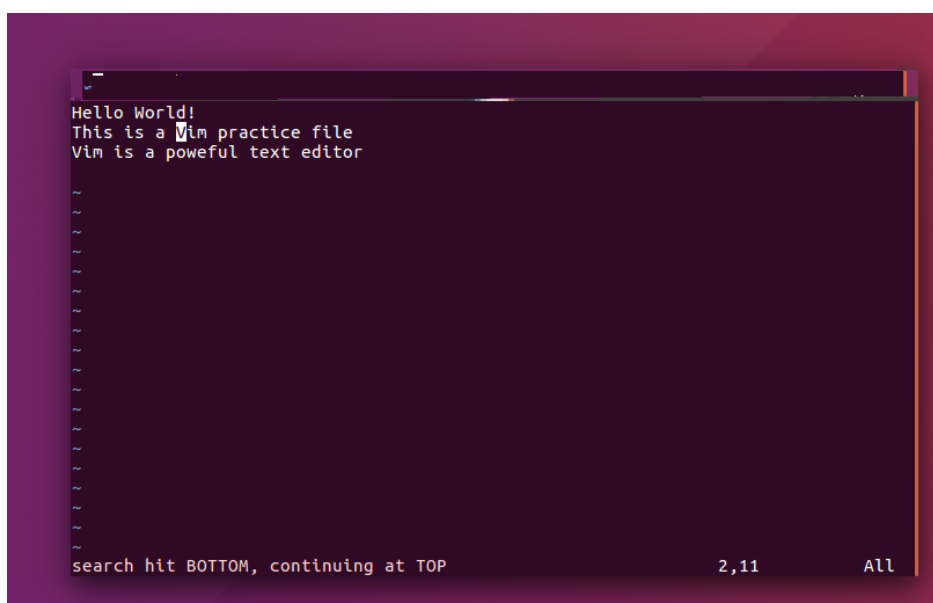


图 16: 查找

实例 2.6 撤销上一步操作

Q: 在 Vim 中撤销上一步编辑操作。

A: 在命令模式下按 `u`。

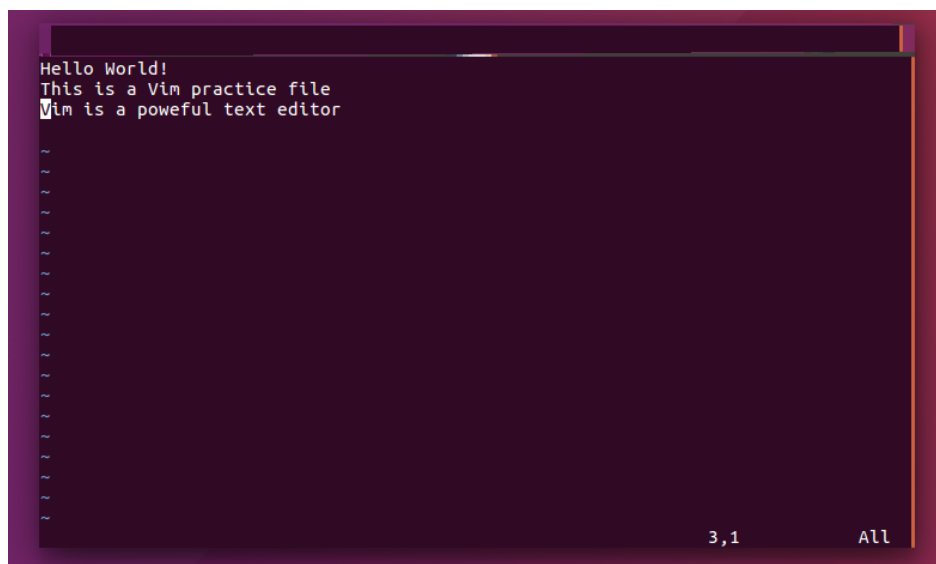


图 17: 撤销上一步操作

实例 2.7 分割窗口

Q: 在 Vim 中水平分割窗口。

A: 在命令模式下输入: `:split`

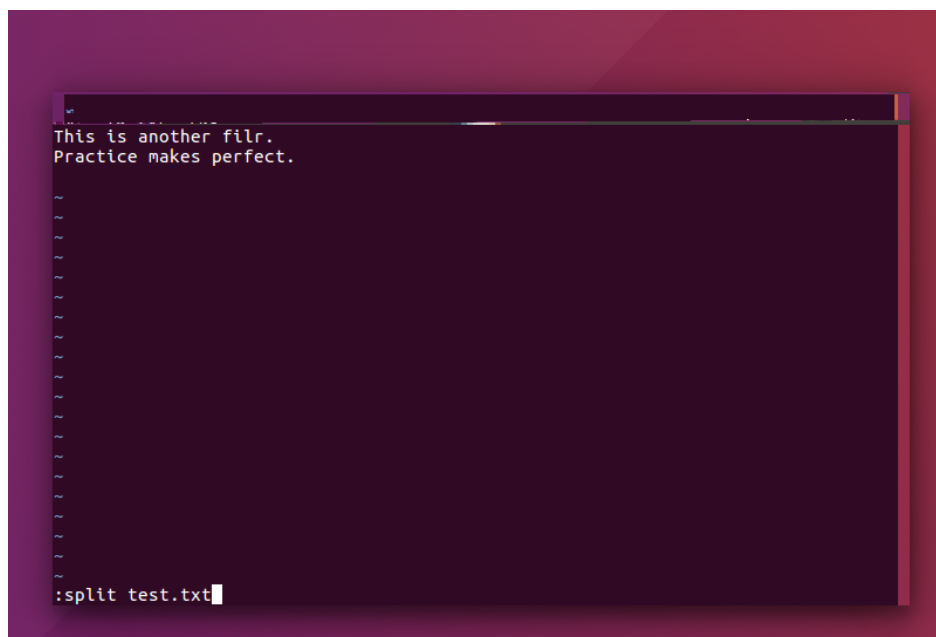


图 18: 分割窗口

实例 2.8 复制粘贴

Q: 在 Vim 中复制一行并粘贴。

A: 将光标移至要复制的行。按 `yy` 复制该行。移动光标至目标位置，按 `p` 粘贴。

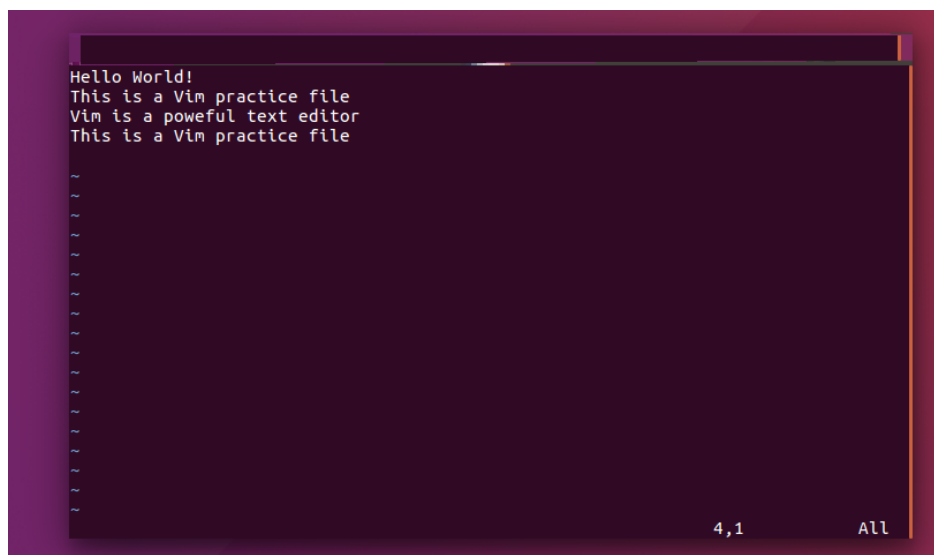


图 19: 复制与粘贴

实例 2.9 录制并应用宏

Q: 在 Vim 中录制宏并重复执行。

A: 按 q 加一个寄存器（如 q）开始录制。执行一系列操作。按 q 结束录制。按 @q 执行宏。

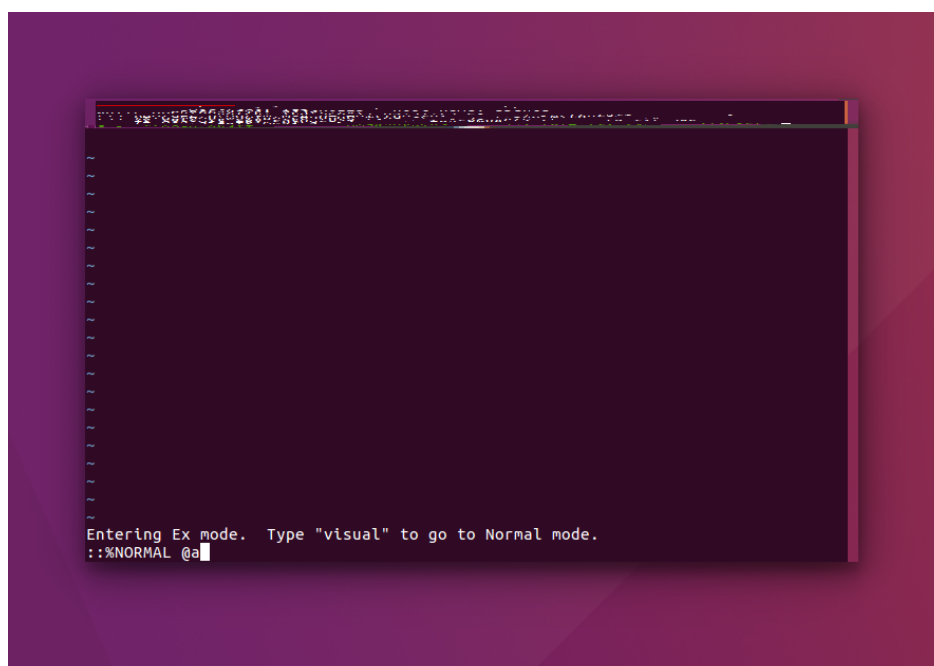


图 20: 录制并应用宏

实例 2.10 全局替换

Q: 在 Vim 中对整个文件进行字符串替换。

A: 在命令模式下输入: :%s/old/new/g

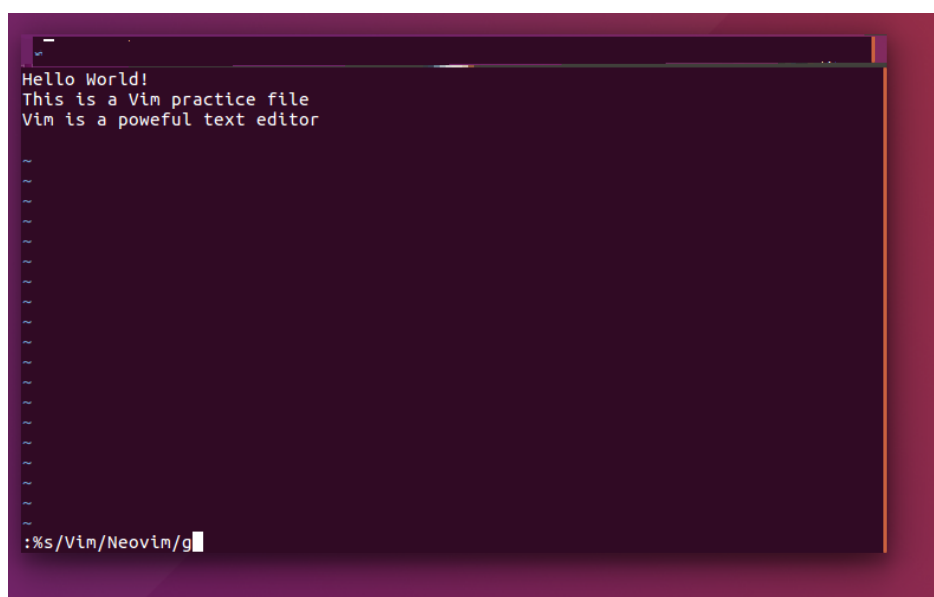


图 21: 全局替换

个人心得

通过本次实验，我系统性地学习了 Shell 和 Vim 两大核心工具的基本操作与高级功能。在 Shell 部分，掌握了文件管理、权限设置、进程执行等关键命令，理解了命令行环境下的高效工作流。Vim 部分则从基础编辑入手，逐步练习了文本替换、宏录制、窗口分割等进阶操作，显著提升了代码和文档的编辑效率。

实验过程中，我尤其体会到理论与实践结合的重要性：仅阅读手册不足以熟练掌握工具，唯有反复实践才能内化技能。此外，遇到权限问题、命令失败等情况时，通过查阅文档和调试解决了问题，增强了自主解决问题的能力。

本次实验不仅巩固了系统开发工具的基础知识，也为后续的编程和系统管理工作奠定了坚实基础。未来我将进一步探索 Shell 脚本编写和 Vim 定制化配置，以提升自动化处理与个性化开发环境的构建能力。