



实验报告二

课程：系统开发工具基础

姓名：陈培诺

学号：23160001003

时间：2024年8月30日

目录

1	Shell 工具和脚本	3
1.1	实例1: find命令	3
1.2	实例2: man命令	3
1.3	实例3: 脚本计算1-100奇数和	5
1.4	实例4: shell脚本判断所输整数是否为质数	5
1.5	实例5: touch命令创建文本	6
1.6	实例6: ls命令	7
1.7	实例7: pwd命令	7
1.8	实例8: cat命令	7
1.9	cp命令拷贝文件	7
1.10	实例10: shell脚本输出九九乘法表	8

2	vim文本编辑器	8
2.1	实例11 vi/vim	8
2.2	实例12: 插入模式	9
2.3	实例13: wq指令	9
2.4	实例14: 删除字符	10
2.5	实例15: 撤销上一次操作	10
2.6	实例16: 进入底线命令模式	10
2.7	实例17: 从编辑模式切换命令模式	10
2.8	实例18: 统计文件字符数	11
2.9	实例19: 粘贴	11
2.10	实例20: 四个退出指令的区别	11
3	git仓库链接	12
4	学习心得	12

1 Shell 工具和脚本

1.1 实例1: find命令

find命令的一般形式为:find pathname -options [-print -exec -ok ...]

find命令的参数:

pathname: find命令所查找的目录路径。例如用.来表示当前目录, 用/来表示系统根目录

-print: find命令将匹配的文件输出到标准输出

-ok: 和-exec的作用相同,只不过以一种更为安全的模式来执行该参数所给出的shell命令,在执行每一个命令之前,都会给出提示,让用户来确定是否执行

```
cpn@cpn-virtual-machine:~/桌面$ cd test1
cpn@cpn-virtual-machine:~/桌面/test1$ ls
helloworld.cpp  pe.doc  test1  test2
cpn@cpn-virtual-machine:~/桌面/test1$ find . -name "test2"
./test2
cpn@cpn-virtual-machine:~/桌面/test1$ find . -name ".doc"
cpn@cpn-virtual-machine:~/桌面/test1$ find . -type f
./test1
./helloworld.cpp
./pe.doc
./test2
```

图 1: 实例1

1.2 实例2: man命令

通过man指令可以查看shell中的指令帮助、配置文件帮助和编程帮助等信息

```

GEDIT(1)                                General Commands Manual                                GEDIT(1)

NAME
    gedit - text editor for the GNOME Desktop

SYNOPSIS
    gedit [OPTION...] [FILE...] [+LINE[:COLUMN]]
    gedit [OPTION...] -

DESCRIPTION
    gedit is the official text editor of the GNOME desktop environment.

    While aiming at simplicity and ease of use, gedit is a powerful general
    purpose text editor. It can be used to create and edit all kinds of
    text files.

    gedit features a flexible plugin system which can be used to dynami-
    cally add new advanced features to gedit itself.

OPTIONS
    --encoding
        Set the character encoding to be used for opening the files
        listed on the command line.

    --list-encodings
        Display list of possible values for the encoding option and
        exit.

    --new-window
        Create a new toplevel window in an existing instance of gedit.

    --new-document
        Create a new document in an existing instance of gedit.

    -s, --standalone
        Run gedit in standalone mode.

    -w, --wait
        Open files and block the gedit process.

    --help
        Prints the command line options.

    --version
        Output version information and exit.

FILE
    Specifies the file to open when gedit starts. If this is not
    specified, gedit will load a blank file with an "Unsaved Docu-
    ment" label. Multiple files can be loaded if they are separated
    by spaces. gedit also supports handling of remote files. For
    example, you can pass the location of a webpage to gedit, like
    "http://www.gnome.org", or load a file from a FTP server, like
    "ftp://ftp.gnome.org/robots.txt".

-
    gedit will read from stdin

+LINE
    For the first file, go to the line specified by LINE (do not in-
    sert a space between the "+" sign and the number). If LINE is
    missing, go to the last line.

COLUMN
    For the first file, go to the column specified by COLUMN. If
    COLUMN is missing, go to the first column.

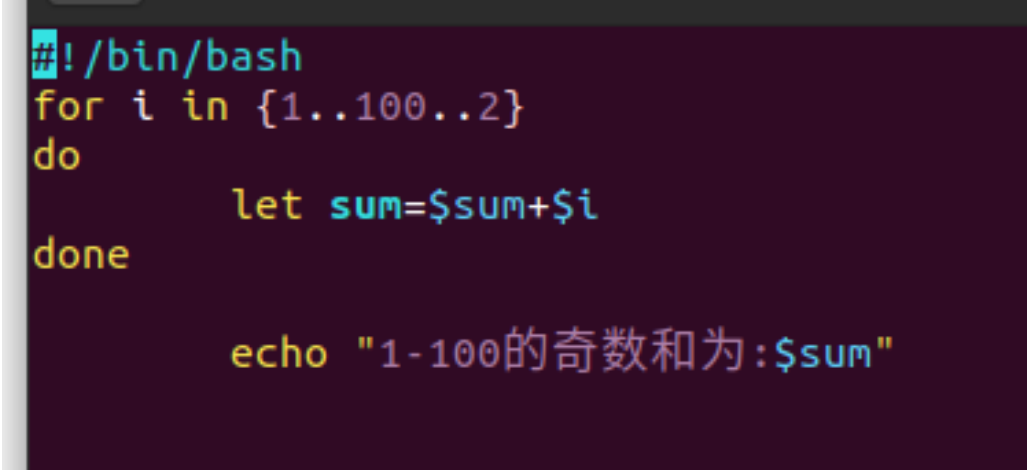
```

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

图 2: 实例2 man命令获取gedit的信息

1.3 实例3: 脚本计算1-100奇数和

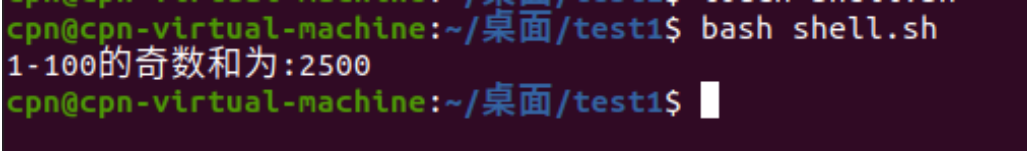
思路: 定义一个变量来保存奇数的和 (sum=0) 找出1-100的奇数, 保存到另一个变量里 (i=遍历出来的奇数) 从1-100中找出奇数后,再相加,然后将和赋值给变量 (循环变量 for) 遍历完毕后,将sum的值打印出来



```
#!/bin/bash
for i in {1..100..2}
do
    let sum=$sum+$i
done

echo "1-100的奇数和为:$sum"
```

图 3: 实例3 shell脚本内容



```
cpn@cpn-virtual-machine:~/桌面/test1$ bash shell.sh
1-100的奇数和为:2500
cpn@cpn-virtual-machine:~/桌面/test1$
```

图 4: 实例3 shell脚本执行:结果为2500

1.4 实例4: shell脚本判断所输整数是否为质数

思路: 让用户输入一个数,保存到一个变量里,如果能被其他数整除就不是质数如果输入的数是1或者2取模根据上面判断又不符合,所以先排除1和2

```
#!/bin/bash
read -p "请输入一个正整数:" number

# 先排除用户输入的数字1和2
if [ $number -eq 1 ]; then
    echo "$number不是质数"
    exit
elif [ $number -eq 2 ]; then
    echo "$number是质数"
    exit
fi

# 循环判断用户所输入的数字是否是质数
for (( i=2; i<=$((number/2)); i++ ))
do
    if [ $((number%i)) -eq 0 ]; then
        echo "$number不是质数"
        exit
    fi
done
echo "$number是质数"
```

图 5: 实例4 shell脚本内容

```
cpn@cpn-virtual-machine:~/桌面/test$ bash aaa.sh
请输入一个正整数:9
9不是质数
cpn@cpn-virtual-machine:~/桌面/test$ bash aaa.sh
请输入一个正整数:11
11是质数
```

图 6: 实例4 shell脚本执行结果

1.5 实例5: touch命令创建文本

形式为touch xxx.后缀如 touch abc.txt 创建一个文件名为abc的文本
touch abc.sh 创建一个名为abc的shell脚本文件

1.6 实例6: ls命令

ls命令可以查看指定目录下包含哪些文件

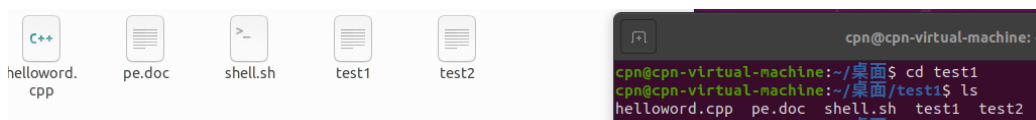


图 7: 实例6 查看test1这一目录下有哪些文件

1.7 实例7: pwd命令

pwd命令可以查看当前所处文件位置

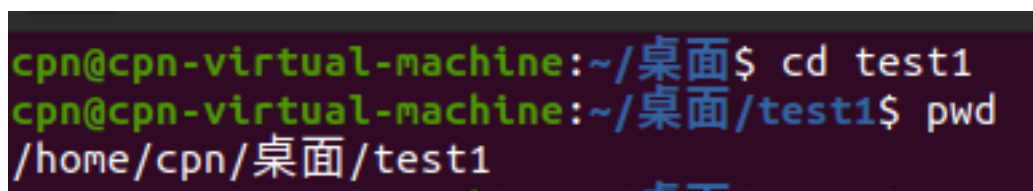


图 8: 实例7 pwd路径查询结果

1.8 实例8: cat命令

cat命令可以显示文本内容



图 9: 实例9 cat显示test1文本内容

1.9 cp命令拷贝文件

复制文件：将一个或多个文件复制到指定的目标位置，例如：cp file1.txt file2.txt将file1.txt复制为file2.txt

复制目录：使用-r选项可以递归复制整个目录及其子目录，例如：cp -r dir1 dir2将dir1目录及其所有子目录和文件复制到dir2目录

1.10 实例10: shell脚本输出九九乘法表

```
1 #!/bin/bash
2 for ((i=1;i<=9;i++))
3 do
4     for ((j=1;j<=i;j++))
5     do
6         echo -n " $i*$j=$[i*j]"
7     done
8 echo
9 done
10
```

图 10: 实例10 shell脚本内容

```
cpn@cpn-virtual-machine:~/桌面/zuoye$ bash bb.sh
1*1=1
2*1=2 2*2=4
3*1=3 3*2=6 3*3=9
4*1=4 4*2=8 4*3=12 4*4=16
5*1=5 5*2=10 5*3=15 5*4=20 5*5=25
6*1=6 6*2=12 6*3=18 6*4=24 6*5=30 6*6=36
7*1=7 7*2=14 7*3=21 7*4=28 7*5=35 7*6=42 7*7=49
8*1=8 8*2=16 8*3=24 8*4=32 8*5=40 8*6=48 8*7=56 8*8=64
9*1=9 9*2=18 9*3=27 9*4=36 9*5=45 9*6=54 9*7=63 9*8=72 9*9=81
```

图 11: 实例10 shell脚本执行结果

2 vim文本编辑器

2.1 实例11 vi/vim

通过vi/vim + 文件路径即可进入编辑文件状态

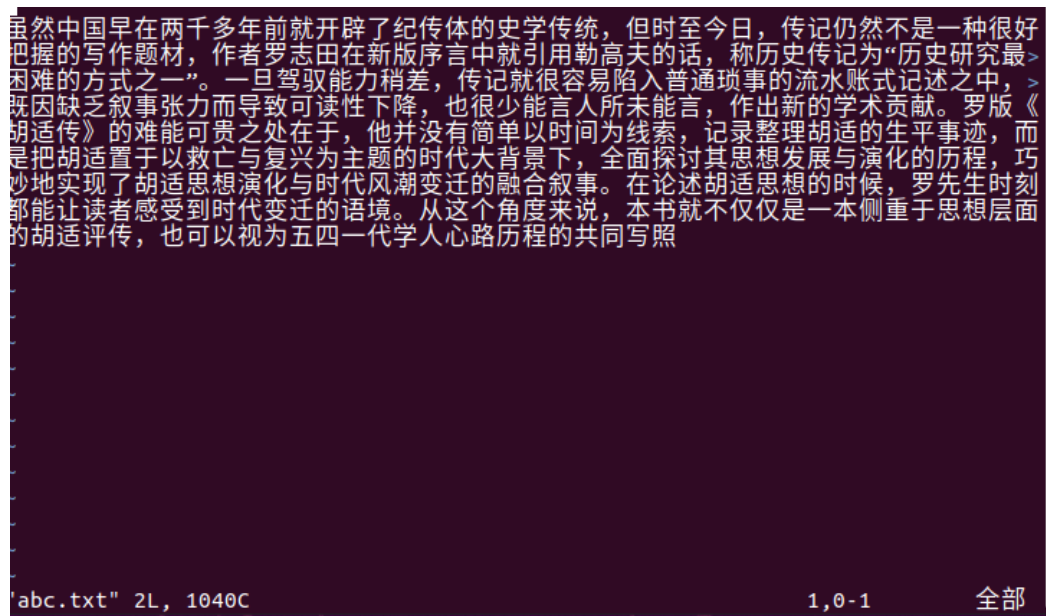


图 12: 实例11 vi/vim进入编辑

2.2 实例12: 插入模式

输入i即可在光标位置开始输入文本

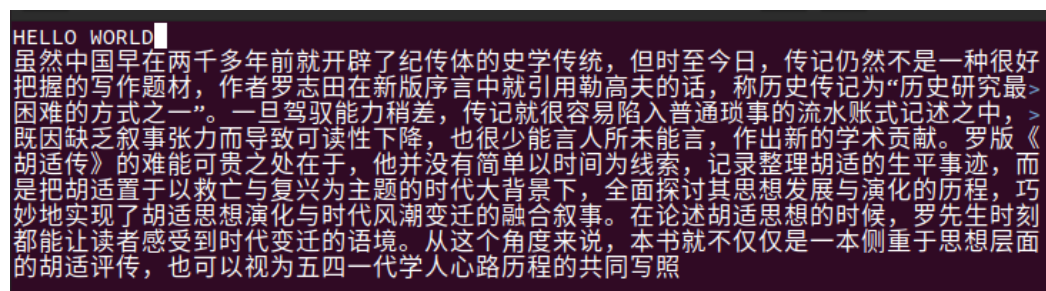


图 13: 实例12 输入HELLO WORLD

2.3 实例13: wq指令

使用wq指令即可实现保存文件并退出

2.4 实例14: 删除字符

键盘输入x即可删除当前光标的字符

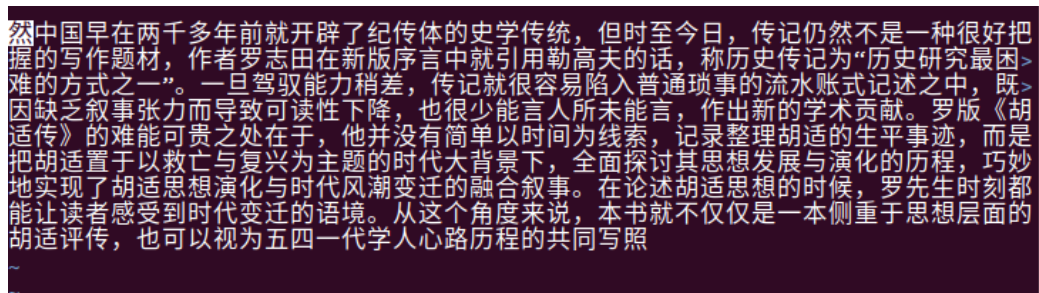


图 14: 实例14 删除”虽”这一字符

2.5 实例15: 撤销上一次操作

键盘输入u即可撤销上一次操作

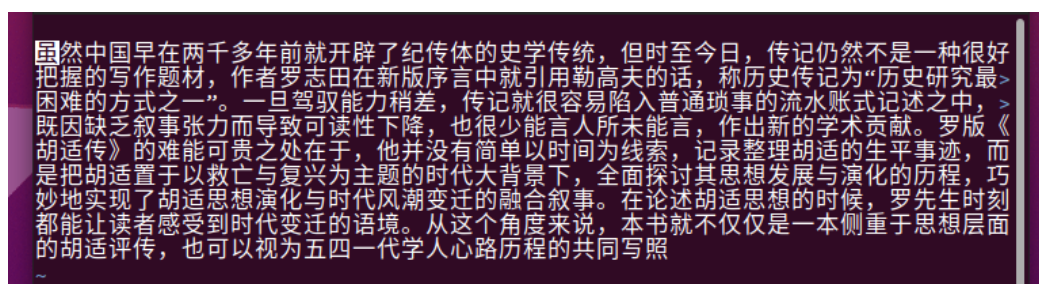


图 15: 实例15 撤销删除”虽”的操作

2.6 实例16: 进入底线命令模式

键盘输入”-”即可进入底线命令模式

2.7 实例17: 从编辑模式切换命令模式

若输入了i进入编辑模式,想切换回命令模式,输入ESC即可

2.8 实例18: 统计文件字符数

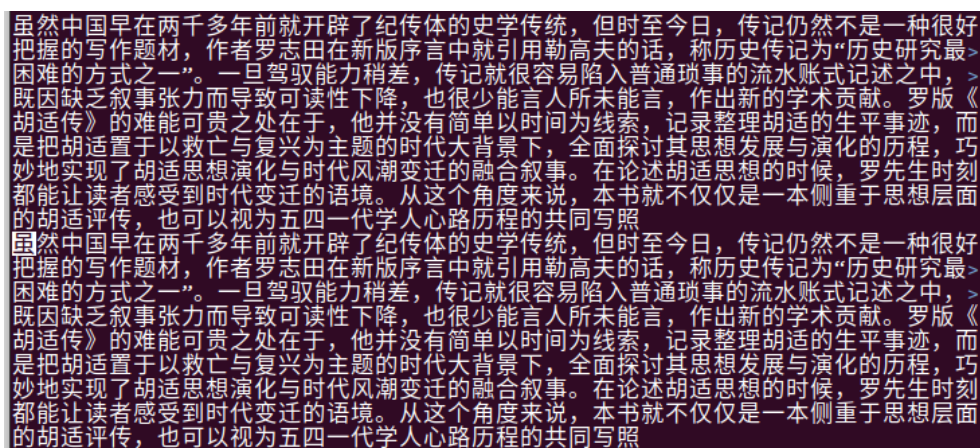
键盘输入 `wc -m + 文件名` 即可统计文件字符数

```
cpn@cpn-virtual-machine:~/桌面/zuoye$ wc -m abc.txt
348 abc.txt
```

图 16: 实例18 统计abc.txt文件字符数为 348

2.9 实例19: 粘贴

在命令模式下输入P即可实现粘贴



虽然中国早在两千多年前就开辟了纪传体的史学传统，但时至今日，传记仍然不是一种很好把握的写作题材，作者罗志田在新版序言中就引用勒高夫的话，称历史传记为“历史研究最困难的方式之一”。一旦驾驭能力稍差，传记就很容易陷入普通琐事的流水账式记述之中，既因缺乏叙事张力而导致可读性下降，也很少能言人所未能言，作出新的学术贡献。罗版《胡适传》的难能可贵之处在于，他并没有简单以时间为线索，记录整理胡适的生平事迹，而是把胡适置于以救亡与复兴为主题的时代大背景下，全面探讨其思想发展与演化的历程，巧妙地实现了胡适思想演化与时代风潮变迁的融合叙事。在论述胡适思想的时候，罗先生时刻都能让读者感受到时代变迁的语境。从这个角度来说，本书就不仅仅是一本侧重于思想层面的胡适评传，也可以视为五四一代学人心路历程的共同写照

虽然中国早在两千多年前就开辟了纪传体的史学传统，但时至今日，传记仍然不是一种很好把握的写作题材，作者罗志田在新版序言中就引用勒高夫的话，称历史传记为“历史研究最困难的方式之一”。一旦驾驭能力稍差，传记就很容易陷入普通琐事的流水账式记述之中，既因缺乏叙事张力而导致可读性下降，也很少能言人所未能言，作出新的学术贡献。罗版《胡适传》的难能可贵之处在于，他并没有简单以时间为线索，记录整理胡适的生平事迹，而是把胡适置于以救亡与复兴为主题的时代大背景下，全面探讨其思想发展与演化的历程，巧妙地实现了胡适思想演化与时代风潮变迁的融合叙事。在论述胡适思想的时候，罗先生时刻都能让读者感受到时代变迁的语境。从这个角度来说，本书就不仅仅是一本侧重于思想层面的胡适评传，也可以视为五四一代学人心路历程的共同写照

图 17: 实例19 将原来的内容粘贴了一次

2.10 实例20: 四个退出指令的区别

:q 当vim进入文件没有对文件内容做任何操作可以按”q”退出

:q! 当vim进入文件对文件内容有操作但不想保存退出

:wq 正常保存退出

:wq! 强行保存退出,只针对与root用户或文件所有人生效

3 git仓库链接

<https://github.com/cpn-cyber/-.git>

4 学习心得

经过这次实验，我系统学习了shell工具及其脚本,vim编辑器并了解了数据整理方法，通过以上二十个实例，也有助于我巩固对这些知识的掌握

在shell中我认为我对for循环这块知识还不够熟练，还需继续练习其编程题。而在vim编辑中我需要要牢牢记住那些重要的命令

这节课只是初步学习这些知识，在今后我还得继续深入学习shell工具以及脚本，为我以后做相关开发奠定基础