

《Linux 实训》实验报告

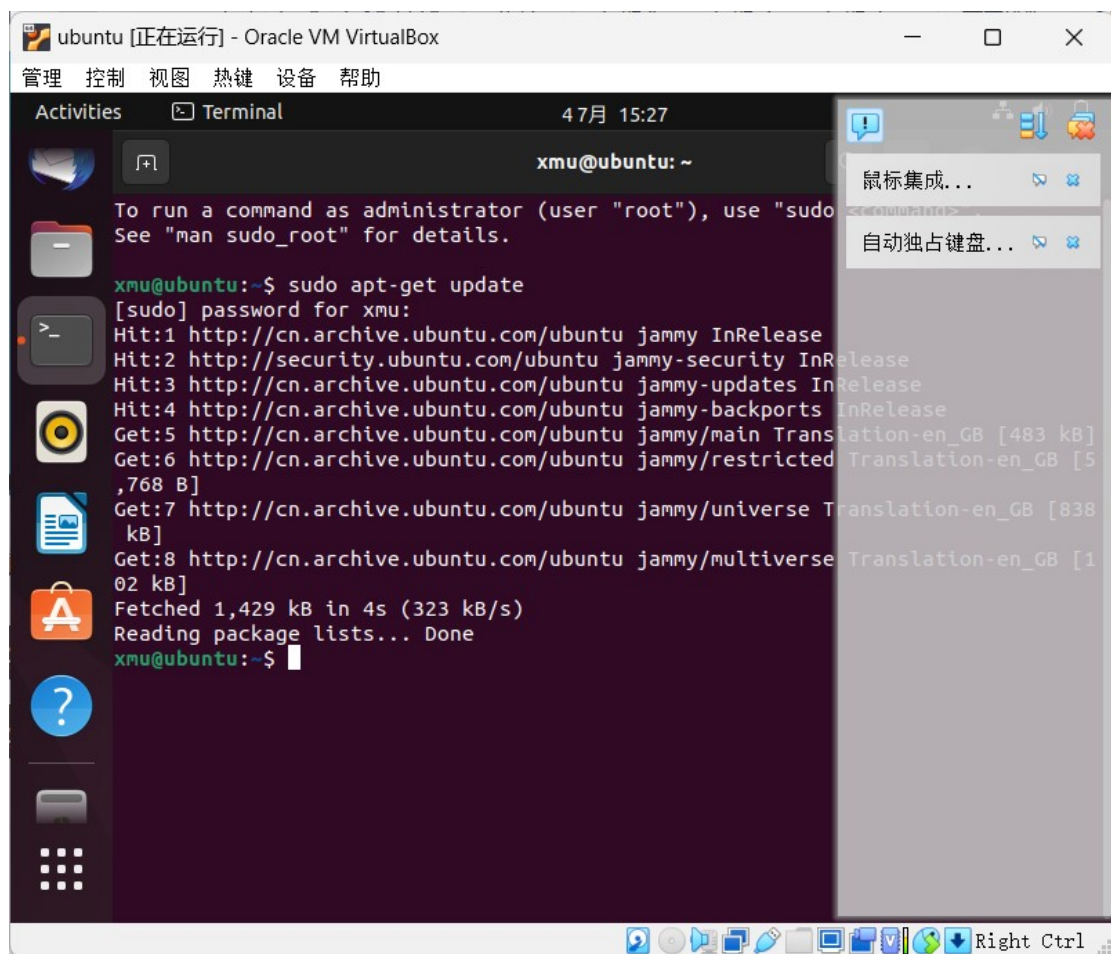
实验 2

1.作业要求概述

可以输出图形字符的命令 banner

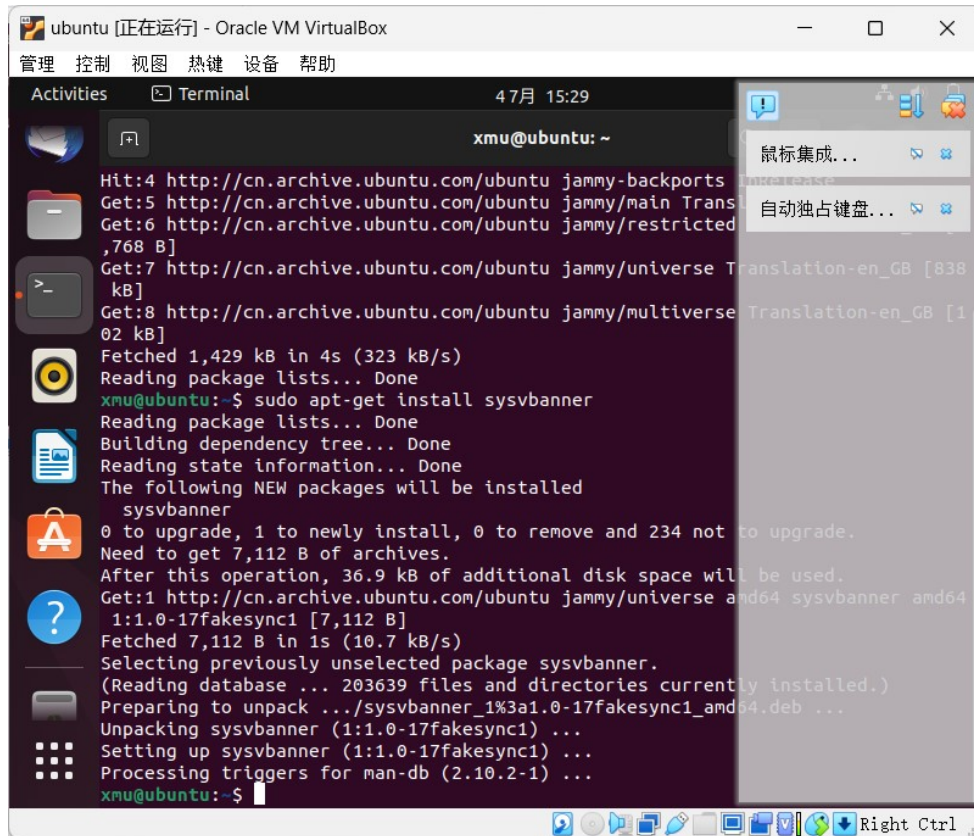
2.操作截图和简要说明

(1)输入 `sudo apt-get update`

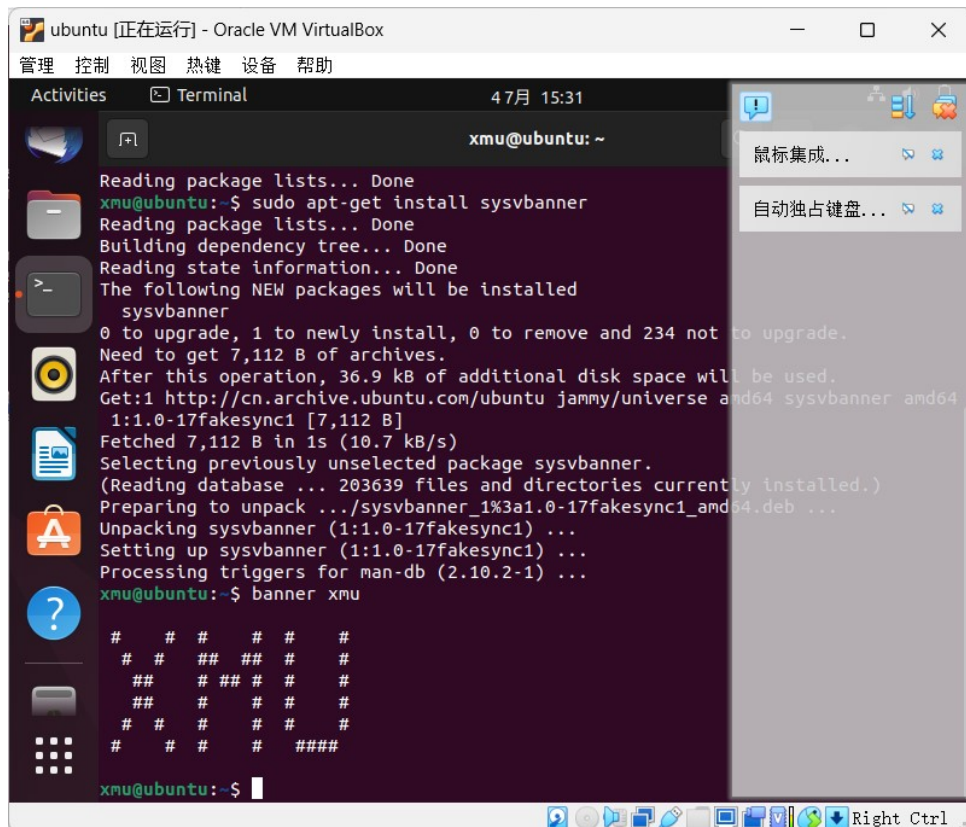


```
ubuntu [正在运行] - Oracle VM VirtualBox
管理 控制 视图 热键 设备 帮助
Activities Terminal 47月 15:27
xmu@ubuntu: ~
To run a command as administrator (user "root"), use "sudo"
See "man sudo_root" for details.
xmu@ubuntu:~$ sudo apt-get update
[sudo] password for xmu:
Hit:1 http://cn.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://cn.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://cn.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:5 http://cn.archive.ubuntu.com/ubuntu jammy/main Translation-en_GB [483 kB]
Get:6 http://cn.archive.ubuntu.com/ubuntu jammy/restricted Translation-en_GB [5,768 B]
Get:7 http://cn.archive.ubuntu.com/ubuntu jammy/universe Translation-en_GB [838 kB]
Get:8 http://cn.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en_GB [102 kB]
Fetched 1,429 kB in 4s (323 kB/s)
Reading package lists... Done
xmu@ubuntu:~$
```

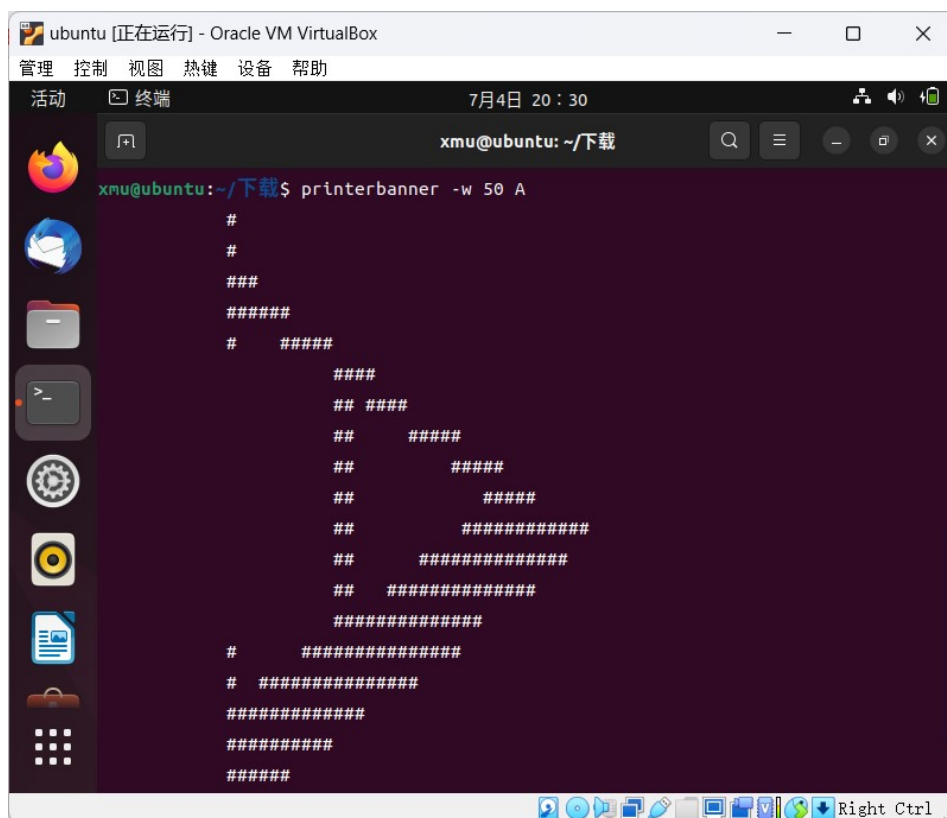
(2)输入 `sudo apt-get install sysvbanner`



(3) 运行 banner xmu

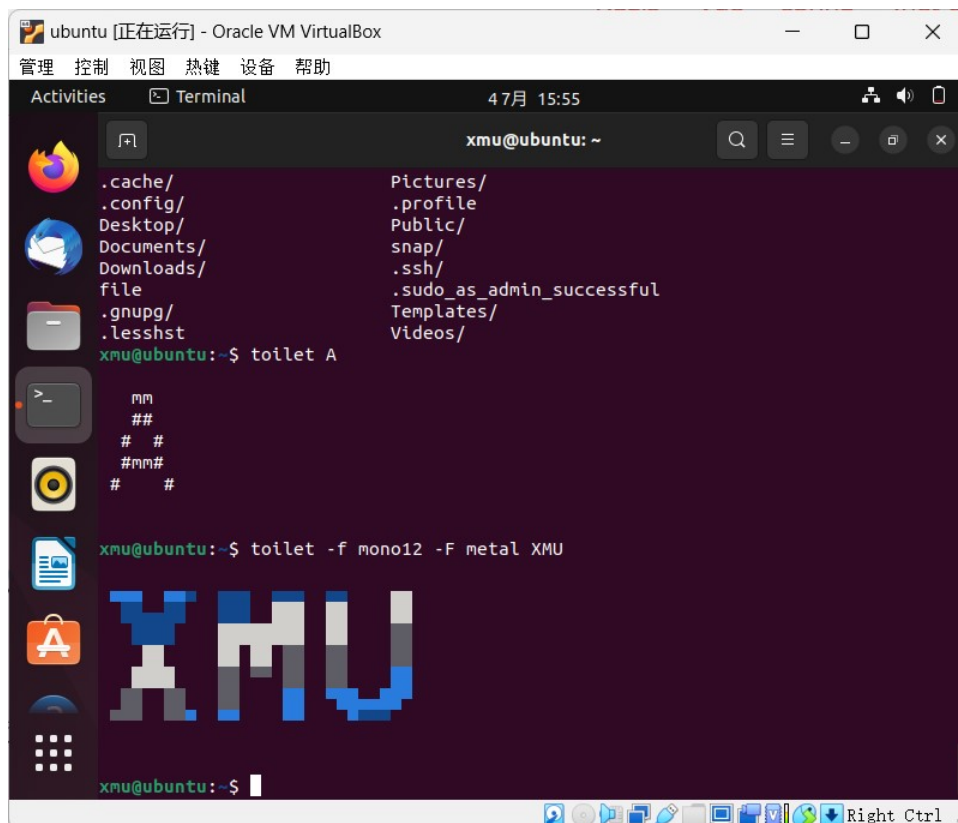


(4) 运行 `printerbanner -w 50 A`



The screenshot shows a terminal window titled "ubuntu [正在运行] - Oracle VM VirtualBox". The terminal prompt is `xmu@ubuntu: ~/下载`. The command `printerbanner -w 50 A` has been executed, resulting in a large, complex ASCII art banner. The banner consists of multiple lines of hash symbols (`#`) and spaces, forming a wide, rectangular shape. The terminal window includes a sidebar with application icons and a top bar with system information and window controls.

(5) `toilet` 命令



The screenshot shows a terminal window titled "ubuntu [正在运行] - Oracle VM VirtualBox". The terminal prompt is `xmu@ubuntu: ~`. The command `toilet A` has been executed, resulting in a small ASCII art banner. The banner consists of a few lines of hash symbols (`#`) and spaces, forming a small, rectangular shape. The terminal window includes a sidebar with application icons and a top bar with system information and window controls.

(6) `figlet` 命令


```
xmu@ubuntu:~$ sudo adduser loutest
正在添加用户"loutest"...
正在添加新组"loutest" (1002)...
正在添加新用户"loutest" (1002) 到组"loutest"...
创建主目录"/home/loutest"...
正在从"/etc/skel"复制文件...
新的 密码:
无效的密码: 密码少于 8 个字符
重新输入新的 密码:
passwd: 已成功更新密码
正在改变 loutest 的用户信息
请输入新值, 或直接敲回车键以使用默认值
    全名 []:
    房间号码 []:
    工作电话 []:
    家庭电话 []:
    其它 []:
这些信息是否正确? [Y/n] y
```

```
xmu@ubuntu:~$ ls /home
cc  loutest  xmu
```

(2) 给用户 loutest 赋予 root 权限 (加入 sudo 组)

```
xmu@ubuntu:~$ sudo usermod -G sudo loutest
xmu@ubuntu:~$ groups loutest
loutest : loutest sudo
```

(3) 使用 sudo 创建文件/opt/forloutest

```
xmu@ubuntu:~$ sudo touch /opt/forloutest
xmu@ubuntu:~$ cd /opt
xmu@ubuntu:/opt$ /ls
-bash: /ls: 没有那个文件或目录
xmu@ubuntu:/opt$ ls
forloutest
```

(4) 更改目录的所有者为 "loutest" 用户

```
xmu@ubuntu:/opt$ ls -alh forloutest
-rw-r--r-- 1 root root 0  7月  5 14:39 forloutest
xmu@ubuntu:/opt$ sudo chown loutest forloutest
xmu@ubuntu:/opt$ ls -alh forloutest
-rw-r--r-- 1 loutest root 0  7月  5 14:39 forloutest
```

(5) 授予 "loutest" 用户对目录的读写权限

```
xmu@ubuntu:/opt$ sudo chmod 600 forloutest
xmu@ubuntu:/opt$ ls -alh forloutest
-rw----- 1 loutest root 0 7月 5 14:39 forloutest
```

3.其他

实验 4

1.作业要求概述

创建一个 homework 目录，建立名为 1.txt ~ 10.txt 文件，并删除 1.txt ~ 5.txt

Linux 的日志文件在哪个目录？

2.操作截图和简要说明

(1) 创建一个 homework 目录

```
xmu@ubuntu:~$ mkdir homework
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap
模板    图片  下载  桌面  father        homework
```

(2) 建立名为 1.txt ~ 10.txt 文件

```
xmu@ubuntu:~$ cd homework
xmu@ubuntu:~/homework$ touch {1..10}.txt
xmu@ubuntu:~/homework$ ls
10.txt  1.txt  2.txt  3.txt  4.txt  5.txt  6.txt  7.txt  8.txt  9.txt
```

(3) 删除 1.txt-5.txt

```
xmu@ubuntu:~/homework$ rm {1..5}.txt
xmu@ubuntu:~/homework$ ls
10.txt  6.txt  7.txt  8.txt  9.txt
```

(4) Linux 的日志文件在哪个目录？

位于 “/var/log” 目录

```
xmu@ubuntu:~$ cd /var/log
xmu@ubuntu:/var/log$ ls
alternatives.log  dmesg.2.gz      kern.log
apt               dmesg.3.gz      lastlog
auth.log          dmesg.4.gz      openvpn
boot.log          dpkg.log         private
bootstrap.log     faillog          speech-dispatcher
btmpt             fontconfig.log   syslog
cups              gdm3             ubuntu-advantage.log
dist-upgrade      gpu-manager.log  ubuntu-advantage-timer.log
dmesg             hp               unattended-upgrades
dmesg.0           installer        vboxpostinstall.log
dmesg.1.gz        journal          wtmp
```

3.其他

实验 5

1.作业要求概述

找出 /etc/ 目录下的所有以 .list 结尾的文件。

2.操作截图和简要说明

(1)sudo 表示使用 root 权限，find 后先接-name 表示搜索名字，后接一个正则表达式匹配，*表示一个或者多个字符。

```
xmu@ubuntu:~$ sudo find /etc/ -name "*.list"
[sudo] xmu 的密码:
/etc/apt/sources.list
/etc/gnome/defaults.list
```

3.其他

挑战：寻找文件

介绍

有一个非常重要的文件（`sources.list`）但是你忘了它在哪了，你依稀记得它在 `/etc/` 目录下，现在要你把这个文件找出来，然后设置成自己（`shiyamlou` 用户）可以访问，但是其他用户并不能访问。

目标

1. 找到 `sources.list` 文件
2. 把文件所有者改为自己（`shiyamlou`）
3. 把权限修改为仅仅只有自己可读可写

1. 找到 `sources.list` 文件

```
xmu@ubuntu:~/homework$ sudo find /etc/ -name "sources.list"
[sudo] xmu 的密码:
/etc/apt/sources.list
```

2. 把文件所有者改为自己(xmu)

```
xmu@ubuntu:/etc/apt$ sudo chown xmu sources.list
xmu@ubuntu:/etc/apt$ ls -alh sources.list
-rw-rw-r-- 1 xmu root 2.8K 7月 4 14:50 sources.list
```

3. 把权限修改为仅仅只有自己可以读写

```
xmu@ubuntu:/etc/apt$ ls -alh sources.list
-rw----- 1 xmu root 2.8K 7月 4 14:50 sources.list
```

实验 6

1. 作业要求概述

创建一个名为 `test` 的文件，分别用 `zip` 和 `tar` 打包成压缩包，再解压到 `/home/shiyamlou` 目录。

2. 操作截图和简要说明

(1) 创建一个名为 `test` 的文件


```
xmu@ubuntu:~$ touch test
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap
模板    图片  下载  桌面  father        homework  test
```

(2)用 zip 打包成压缩包

```
xmu@ubuntu:~$ zip test.zip test
  adding: test (stored 0%)
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap  test.zip
模板    图片  下载  桌面  father        homework  test
```

(3)用 tar 打包成压缩包

```
xmu@ubuntu:~$ tar -cf test.tar test
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap  test.tar
模板    图片  下载  桌面  father        homework  test  test.zip
```

(4)解压 test.zip 到/home/xmu 目录

```
xmu@ubuntu:~$ unzip test.zip -d /home/xmu
Archive:  test.zip
replace /home/xmu/test? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  extracting: /home/xmu/test
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap  test.tar
模板    图片  下载  桌面  father        homework  test  test.zip
```

(5)解压 test.tar 到/home/xmu 目录

```
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap  test.zip
模板    图片  下载  桌面  father        homework  test.tar
xmu@ubuntu:~$ tar -xf test.tar -C /home/xmu
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  file  snap  test.tar
模板    图片  下载  桌面  father        homework  test  test.zip
```

3.其他

实验 7

1.作业要求概述

找出当前目录下占用最大的前十个文件。

2.操作截图和简要说明

du 为输出文件容量的指令，-a 为显示所有文件大小，-h 为易读方式，结合为-ah，

“.”表示当前目录 sort -rh 为结果最大到最小排序，head -n 10 为显示前 10 个。

```
xmu@ubuntu:~$ du -ah . | sort -rh | head -n 10
66M      .
32M      ./cache
24M      ./cache/tracker3/files
24M      ./cache/tracker3
17M      ./commands.log
15M      ./thunderbird/razawqxo.default-release
15M      ./thunderbird
6.5M     ./cache/thunderbird/razawqxo.default-release/startupCache
6.5M     ./cache/thunderbird/razawqxo.default-release
6.5M     ./cache/thunderbird
```

3.其他

挑战：备份日志

小明是一个服务器管理员，他需要每天备份论坛数据（这里我们用 alternatives.log 日志替代），备份当天的日志并删除之前的日志。而且备份之后文件名是 年-月-日 的格式。alternatives.log 在 /var/log/ 下面。

目标

1. 为 shiyanlou 用户添加计划任务
2. 每天凌晨 3 点的时候定时备份 alternatives.log 到 /home/shiyanlou/tmp/ 目录
3. 命名格式为 年-月-日，比如今天是 2017 年 4 月 1 日，那么文件名为 2017-04-01

提示语

- date
- crontab
- cp 命令
- 用一条命令写在 crontab 里面即可，不用写脚本

(1) 用 crontab -e 为用户添加计划任务

```
xmu@ubuntu:~$ crontab -e
no crontab for xmu - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/vim.tiny
 3. /bin/ed

Choose 1-3 [1]: 1
crontab: installing new crontab
```

(2) 编辑缓冲区文档

白字为新增的一行：

0 3 * * *: cron 表达式的时间部分，表示在每天凌晨 3 点执行任务。其中，0 表示分钟部分为 0，3 表示小时部分为 3。

cp: 复制文件。

/var/log/alternatives.log: 这是要备份的源文件路径，即 /var/log/ 目录下的 alternatives.log 日志文件。

/home/shiyanlou/tmp/\$(date +%Y-%m-%d): 这是目标文件的路径，使用了命令替换 \$(date +%Y-%m-%d) 来获取当前的日期，并将其作为文件名。+%Y-%m-%d 表示日期格式为年-月-日。

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 3 * * * cp /var/log/alternatives.log /home/xmu/tmp/$(date +%Y-%m-%d)
保存已修改的缓冲区?
Y 是
N 否      ^C 取消
```

实验 10

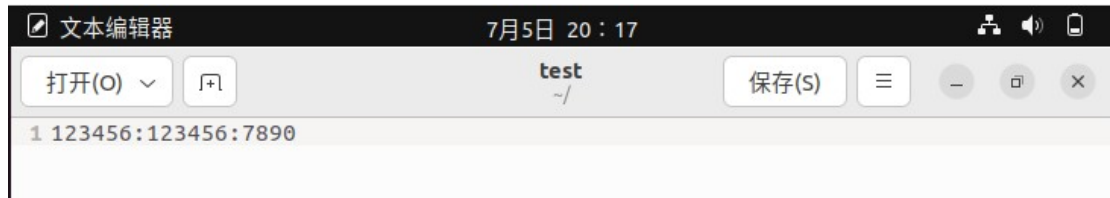
1.作业要求概述

熟悉 `cut`, `sort`, `uniq` 命令以及参数。

2.操作截图和简要说明

(1) `cut`

test 内容如下：



`-c` 打印 test 文件中的第 1-7 个字符

```
xmu@ubuntu:~$ cut -c 1-7 test
123456:
```

`-d` 按指定分隔符提取指定字段，此处按照 “:” 分隔分别打印了第 2 个和第 3 个字段

```
xmu@ubuntu:~$ cut -d ':' -f 2 test
123456
xmu@ubuntu:~$ cut -d ':' -f 3 test
7890
```

此外 `-f` 可以提取每行的指定字段

```
xmu@ubuntu:~$ cut -f 1 test
123456:123456:7890
```

(2) `sort`

将命令得到的结果排序输出

与命令之间用 “|” 隔开，无参数时按照字典排序。

```
xmu@ubuntu:~$ cat /etc/passwd | sort
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
avahi:x:114:121:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
```

`-r` 倒序排序

```
xmu@ubuntu:~$ cat /etc/passwd | sort -r
xmu:x:1000:1000:xmu,,,:/home/xmu:/bin/bash
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
whoopsie:x:117:124::/nonexistent:/bin/false
uidd:x:107:115::/run/uidd:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
usbmux:x:111:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false
```


-n 以数值方式排序

```
xmu@ubuntu:~$ cat /etc/passwd | sort -n
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
avahi:x:114:121:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
cc:x:1001:1001:,,,:/home/cc:/bin/bash
```

(3)uniq

去除重复行（仅相邻）

test 内容:



```
1 1
2 2
3 1
4 3
5 2
6 5
7 5
```

```
xmu@ubuntu:~$ cat test | uniq
1
2
1
3
2
5
```

-c 显示每行重复次数

```
xmu@ubuntu:~$ cat test | sort | uniq -c
2 1
2 2
1 3
2 5
```


3.其他

实验 11

1.作业要求概述

在《文件打包与解压缩》一节实验中提到 Windows/dos 与 Linux/UNIX 文本文件一些特殊字符不一致。

如断行符 Windows 为 CR+LF(\r\n)，Linux/UNIX 为 LF(\n)。使用 `cat -A` 文本 可以看到文本中包含的不可见特殊字符。Linux 的 `\n` 表现出来就是一个 `$`，而 Windows/dos 的表现 为 `^M$`，可以直接使用 `dos2unix` 和 `unix2dos` 工具在两种格式之间进行转换，使用 `file` 命令可以查看文件的具体类型。

不过现在希望你在不使用上述两个转换工具的情况下，使用前面学过的命令手动完成 dos 文本格式到 UNIX 文本格式的转换。

2.操作截图和简要说明

(1)先获得一个 dos 的文件

```
xmu@ubuntu:~$ unix2dos test
unix2dos: 正在转换文件 test 为DOS格式...
xmu@ubuntu:~$ cat -A test
1^M$
2^M$
1^M$
3^M$
2^M$
5^M$
5^M$
```

(2)使用 `sed` 命令将 `\r` 替换为空字符串，即将 dos 文件的 “`\r\n`” 变为 linux 文件的 “`\n`”

使用到的正则表达式 `s/\r//g`，其中 `s/` 表示字符串开始，`g` 表示结束，`\r//` 表示用空字符串替换 `\r`

```
xmu@ubuntu:~$ sed 's/\r//g' test > test_linux
xmu@ubuntu:~$ cat -A test_linux
1$
2$
1$
3$
2$
5$
5$
```

3.其他

实验 12

1.作业要求概述

理解下面这段代码的作用，实际这段代码不会正常工作，请结合这一小节的知识分析这段代码没有正确工作的原因，并设法解决这个问题。

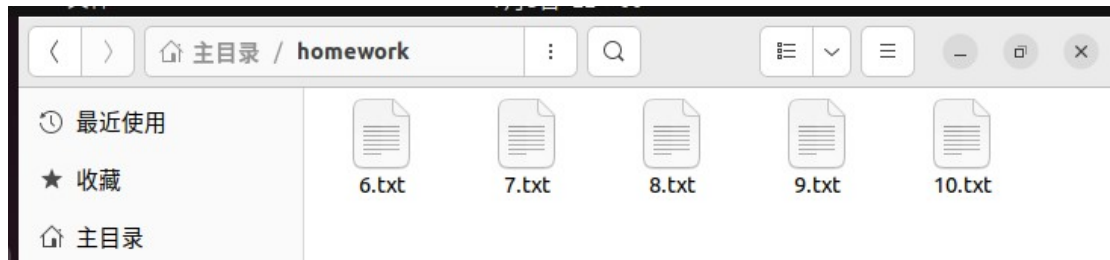
```
while read filename; do
    rm -iv $filename
done <<(ls)
```

2.操作截图和简要说明

作用：这是想把 `ls` 命令的输出也就是当前目录的文件列表逐行读取出来，然后使用 `rm -iv` 在获得你许可的情况下删除他们。

没有正常工作的原因：因为 `rm -iv` 期待用户从标准输入给出一个 `y` 或 `n` 的答案以确认是否删除，但标准输入被 `<<(ls)` 重定向了。于是 `rm -iv` 开始在 `<<(ls)` 里寻找答案。如果找不到 `y` 或者 `n` 就一直寻找下去，直到把 `<<(ls)` 的内容消耗完。这时在下一轮的循环中由于数据没有了，`read` 读不出数据，程序也就退出了。

```
xmu@ubuntu:~/homework$ while read filename; do
>  rm -iv $filename
> done < <(ls)
rm: 是否删除普通空文件 '10.txt'? xmu@ubuntu:~/homework$
```



homework 内的内容没有被删除

解决问题: 把标准输入复制一份出来, 然后让 `rm -iv` 使用复制出来的标准输入。

再把原来的标准输入重定向给 `<<(ls)`

```
exec 3<&0
```

```
while read filename; do
```

```
    rm -iv $filename <&3
```

```
done <<(ls)
```

```
xmu@ubuntu:~/homework$ exec 3<&0
xmu@ubuntu:~/homework$ while read filename; do
>  rm -iv $filename <&3
> done <<(ls)
rm: 是否删除普通空文件 '10.txt'? y
已删除 '10.txt'
rm: 是否删除普通空文件 '6.txt'? y
已删除 '6.txt'
rm: 是否删除普通空文件 '7.txt'? y
已删除 '7.txt'
rm: 是否删除普通空文件 '8.txt'? y
已删除 '8.txt'
rm: 是否删除普通空文件 '9.txt'? y
已删除 '9.txt'
```



homework 内容被全部删除

3.其他

挑战：历史命令

介绍

在 Linux 中，对于文本的处理和分析是极为重要的，现在有一个文件叫做 `data1`，可以使用下面的命令下载：

```
cd /home/shiyanlou
```

```
wget https://labfile.oss-internal.aliyuncs.com/courses/1/data1
```

`data1` 文件里记录是一些命令的操作记录，现在需要你从里面找出出现频率次数前 3 的命令并保存在 `/home/shiyanlou/result`。

目标

1. 处理文本文件 `/home/shiyanlou/data1`
2. 将结果写入 `/home/shiyanlou/result`
3. 结果包含三行内容，每行内容都是出现的次数和命令名称，如“100 ls”

(1) 下载 `data1` 文件

```
xmu@ubuntu:~$ history > data1
xmu@ubuntu:~$ ls
公共的  视频  文档  音乐  commands.log  father  homework  test  test.tar
模板    图片  下载  桌面  data1         file    snap      test_linux  test.zip
xmu@ubuntu:~$ cat data1
 1 touch file
 2 cd/etc/
 3 pwd
 4 cd /etc/
 5 pwd
 6 a
 7 y
 8 touch
 9 touch
10 tail
```

(2) 找出频率前 3 使用频率的指令写入 result

cat /home/xmu/data1 取出文件

awk '{print \$2}' 取出每行第二个字段字符

(为什么不是第一个字段? 因为 data1 是从历史命令中获得的, 其第一个字段是使用序号, 第二个字段才是命令, 如图)



```
1 1 touch file
2 2 cd/etc/
3 3 pwd
4 4 cd /etc/
5 5 pwd
6 6 a
7 7 y
8 8 touch
9 9 touch
10 10 tail
11 11 find/
12 12 cd /home/
13 13 man man
```

sort 进行排序 (为什么提前排序? 因为 uniq 只计算相邻的重复数, 需要进行一次排序使相同的命令相邻进行计数)

第二次 sort -rn 进行倒序排序

Head -3 显示前三行

> /home/xmu/result 将输出重定向到 result 文件

之后用 cat 打开了 result 可以看到结果正确


```
xmu@ubuntu:~$ cat /home/xmu/data1 | awk '{print $2}' | sort | uniq -c | sort -rn | head -3 > /home/xmu/result
xmu@ubuntu:~$ cat result
    30 sudo
    27 cd
    26 ls
xmu@ubuntu:~$
```

实验 13

1.作业要求概述

练习其他几个命令动作的使用。

练习 1：结合正则表达式做更多练习。

练习 2：参考下面的链接，掌握 sed 处理文本的基本原理，理解 pattern space 和 hold space 概念。 [sed 简明教程](#) [sed 单行脚本快速参考](#) [sed 完全手册](#)

练习 3：基于 pattern space 和 hold space 实现将一个文本倒序输出和交换奇数行和偶数行。

2.操作截图和简要说明

(1) sed 处理文本的基本原理：

读取输入：sed 从标准输入或指定的文件中逐行读取文本内容。

模式匹配：对于每一行文本，sed 使用模式来匹配该行是否符合特定的条件。模式可以是正则表达式或行号等。

执行指令：一旦找到匹配的行，sed 将执行与之关联的指令。指令可以是修改文本、删除行、打印行等操作。

输出结果：经过指令处理后的行将被输出到标准输出或者根据需要保存到文件中。

(2) pattern space 和 hold space 概念：

模式空间（Pattern Space）：

模式空间是 sed 中用来存储当前处理的文本行的缓冲区。开始时，模式空间会被初始化为空。每次读取一行文本后，该行的内容就会存储在模式空间中，然后可以对其进行各种操作，如模式匹配、替换、删除、打印等。当处理完当前行后，sed 会自动将模式空间的内容输出到标准输出。

保持空间（Hold Space）：

保持空间是 sed 中的另一个缓冲区，用于存储临时数据。与模式空间不同，保持空间不会随着输入的每一行而变化，它的内容可以在 sed 脚本执行过程中持续保存，并可以在需要时引用。保持空间通常用于保存中间结果或者在处理多行文本时进行操作，例如跨行替换。

(3) 基于 pattern space 和 hold space 实现将一个文本倒序输出和交换奇数行和偶数行。

倒序输出：

还是以刚才的 data 为例

1!G：对于除第一行外的每一行，在模式空间中的内容后追加保持空间的内容。

h：将模式空间的内容复制到保持空间。

\$_d：对于除最后一行外的每一行，删除模式空间的内容。

```
xmu@ubuntu:~$ sed '1!G;h;$_d' data1
226 history > data1
225 cat history > data1
224 cat histfile > data1
223 ls /tmp
222 ls /home/xmu/tmp
221 crontab -e
220 sudo service rsgslog start
219 rsgslog
218 sudo apt-get install -y rsgslog
217 ls -alh sources.list
216 sudo chmod 600 sources.list
```

交换奇数偶数行：

h：将模式空间的内容复制到保持空间，也就是奇数行。

n：读取下一行，也就是偶数行。

G：将保持空间的内容追加到模式空间，交换奇数行和偶数行的内容。

p：打印模式空间的内容。

```
xmu@ubuntu:~$ sed -n 'h;n;G;p' data1
2 cd/etc/
1 touch file
4 cd /etc/
3 pwd
6 a
5 pwd
8 touch
7 y
10 tail
9 touch
```

3.其他