
计算机科学与技术学院

课程报告

课 程 名 称	<u>Linux 实践应用</u>
学 生 姓 名	<u>Banban</u>
学 号	<u></u>
专 业 班 级	<u></u>
设 计 时 间	<u>2020-2021-2</u>
地 点	<u></u>

xx 大学计算机学院

《Linux 应用实践》评分表

专业班级：_____ 学号：_____ 姓名：_____ banban

支撑指标	评定项目	课程目标	评定标准	考核等级	分值范围(分)	得分
能够将数学、自然科学、工程基础和专业知用于解决软件工程应用领域复杂工程问题。	工程知识	课程目标 1	能将数学、自然科学、工程基础及软件工程专业知识与技术应用于解决软件工程应用领域复杂工程问题，包括判定系统的复杂性（工程解决能力）	优/ 良/ 合格/ 不合格	20	
能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析软件工程应用领域复杂工程问题，以获得有效结论。	问题分析	课程目标 2	能够通过文献调研、方案推理等方法给出软件工程应用领域复杂工程问题的多种解决方案并能确定适合具体问题的解决方案（方案分析）	优/ 良/ 合格/ 不合格	40	
能够设计针对软件工程应用领域的复杂工程问题的解决方案，设计满足特定需求的软硬件系统，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。	设计/开发解决方案	课程目标 3	能够根据目标和解决方案，设计或开发满足特定需求的软件工程软硬件系统、组件及产品（设计开发）	优/ 良/ 合格/ 不合格	40	
课程设计评分老师签名：				总分		
问题及回答：						

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

目 录

实验一 Linux 基本命令的使用	5
1、实验目的	5
2、实验内容和步骤	5
3、实验结论	13
实验二 简单程序设计	20
1、实验目的	20
2、实验内容和步骤	20
3、实验结论	25

实验一 Linux 基本命令的使用

1、实验目的

学习和掌握 Linux 的基本命令。

2、实验内容和步骤（写出命令、结果或抓图）

步骤 1：以 root 登录 Linux，创建一个普通账户和口令。（useradd...）

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

xulong@ubuntu:~$ sudo passwd
[sudo] password for xulong:
New password:
Retype new password:
passwd: password updated successfully
xulong@ubuntu:~$ su -root
su: invalid option -- 'r'
Try 'su --help' for more information.
xulong@ubuntu:~$ su - root
Password:
root@ubuntu:~# useradd ahut617
root@ubuntu:~# passwd ahut617
新的 密码:
重新输入新的 密码:
passwd: 已成功更新密码
root@ubuntu:~#
```

步骤 2：使用新创建的用户账户和口令登录 Linux 系统，察看登录后的界面。

```
xulong@ubuntu:~$ su ahut617
密码:
$
```

步骤 3：使用 pwd 命令察看当前的工作目录，然后用 ls 命令查看当前目录下的内容，尝试使用 -a, -l, -F, -A, -lF 等不同选项并比较不同之处。

```
xulong@ubuntu:~$ pwd
/home/xulong
xulong@ubuntu:~$ ls -a
. .bash_logout .cache Desktop Downloads .local Music Pictures Public .thunderbird
.. .bashrc .config Documents .gnupg .mozilla .pam_environment .profile Templates Videos
xulong@ubuntu:~$ ls -l
total 32
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Desktop
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Documents
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Downloads
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Music
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Pictures
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Public
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Templates
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Videos
xulong@ubuntu:~$ ls -F
Desktop/ Documents/ Downloads/ Music/ Pictures/ Public/ Templates/ Videos/
xulong@ubuntu:~$ ls -A
.bash_logout .cache Desktop Downloads .local Music Pictures Public .thunderbird
.bashrc .config Documents .gnupg .mozilla .pam_environment .profile Templates Videos
xulong@ubuntu:~$ ls -lF
total 32
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Desktop/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Documents/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Downloads/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Music/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Pictures/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Public/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Templates/
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Videos/
xulong@ubuntu:~$
```

<http://gitudo.com/dream7787/Computer-learning-Resources.git>

步骤 4: 在当前目录下建立一个名为 test 的新目录，然后将工作目录切换到 test 下，尝试将/etc 目录下的文件 passwd 拷贝到该目录下（cp 源文件 目的目录）。察看当前目录下的 passwd 文件的属主和文件权限。

```
xulong@ubuntu:~$ mkdir test
xulong@ubuntu:~$ cd test
xulong@ubuntu:~/test$ ls
xulong@ubuntu:~/test$ cd
xulong@ubuntu:~$ cp /etc/passwd test
xulong@ubuntu:~$ cd test
xulong@ubuntu:~/test$ ls
passwd
xulong@ubuntu:~/test$ ls -l
total 4
-rw-r--r-- 1 xulong xulong 2741 Jun 19 00:31 passwd
xulong@ubuntu:~/test$ cd
xulong@ubuntu:~$ ls
Desktop Documents Downloads Music Pictures Public Templates test Videos
xulong@ubuntu:~$
```

步骤 5: 尝试向当前目录下的 passwd 文件和/etc/passwd 文件分别写入一些新内容（可使用 echo “字符串” >>文件的命令），看看操作能否成功，如果不能成功，请说明原因。用 cat 命令浏览文件 password 的内容，用 more 命令进行浏览翻页操作，再用 less 命令浏览文件的内容。比较这几个命令的不同之处

```
xulong@ubuntu:~$ cd test
xulong@ubuntu:~/test$ echo '1122334455'>>passwd
xulong@ubuntu:~/test$ cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
saned:x:117:123:./var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
hplip:x:119:7:HPLIP system user,,,:/run/hplip:/bin/false
whoopsie:x:120:125:./nonexistent:/bin/false
colord:x:121:126:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:127:./var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534:./run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
xulong:x:1000:1000:徐龙,,,:/home/xulong:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
1122334455
xulong@ubuntu:~/test$
```

```
geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534:./run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
xulong:x:1000:1000:徐龙,,,:/home/xulong:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
1122334455
xulong@ubuntu:~/test$ more passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
```



```

xulong@ubuntu:~/test$ cd ..
xulong@ubuntu:~$ cd /etc
xulong@ubuntu:/etc$ echo '112233'>>passwd
bash: passwd: Permission denied
xulong@ubuntu:/etc$ cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:nonexistent:/usr/sbin/nologin
syslog:x:104:110:/:home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uidd:x:107:114:/:run/uidd:/usr/sbin/nologin
tcpdump:x:108:115:/:nonexistent:/usr/sbin/nologin

dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:113:120:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
avahi:x:115:121:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/usr/sbin/nologin
saned:x:117:123:/:var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
hplip:x:119:7:HPLIP system user,,,:/run/hplip:/bin/false
whoopsie:x:120:125:/:nonexistent:/bin/false
colord:x:121:126:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:127:/:var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534:/:run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
xulong:x:1000:1000:徐龙,,,:/home/xulong:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:usr/sbin/nologin
1122334455
xulong@ubuntu:~/test$ less passwd

```



```

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mail List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:nonexistent:/usr/sbin/nologin
syslog:x:104:110:/:home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,:/var/lib/tpm:/bin/false
uuidd:x:107:114:/:run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115:/:nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:110:46:usbmux daemon,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:117:RealtimeKit,,:/proc:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:113:120:user for cups-pk-helper service,,:/home/cups-pk-helper:/usr/sbin/nologin
speech-dispatcher:x:114:29:Speech Dispatcher,,:/run/speech-dispatcher:/bin/false
avahi:x:115:121:Avahi mDNS daemon,,:/var/run/avahi-daemon:/usr/sbin/nologin
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,:/usr/sbin/nologin
saned:x:117:123:/:var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager OpenVPN,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
hplip:x:119:7:HPLIP system user,,:/run/hplip:/bin/false
whoopsie:x:120:125:/:nonexistent:/bin/false
colord:x:121:126:colord colour management daemon,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:127:/:var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534:/:run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
:

```

原因：对当前目录中成功，因为该目录下的 passwd 文件对当前用户具有写的权限。对 /etc/passwd 文件的操作被拒绝因为对当前用户不具有写的权限。

cat 命令将文件的内容全部显示，more 命令将文件内容分屏显示，less 命令进行刷新的全部显示

步骤 6: 用 ls 命令查看 test 下文件的权限，用 mv 命令更改文件 password 的文件名为 test.txt，尝试用 chown 和 chgrp 更改文件的属主为 root、组为 root，看看能否成功，不成功，请说明原因。尝试用 chmod 将文件权限为 “-rw-----”。看看能否成功，不成功，请说明原因。

```

xulong@ubuntu:~/test$ ls -l
total 4
-rw-r--r-- 1 xulong xulong 2752 Jun 19 00:49 test.txt
xulong@ubuntu:~/test$ chown root test.txt
chown: changing ownership of 'test.txt': Operation not permitted
xulong@ubuntu:~/test$ chgrp root test.txt
chgrp: cannot access 'test.txt': No such file or directory
xulong@ubuntu:~/test$ ls -l
ls-l: command not found
xulong@ubuntu:~/test$ ls -l
total 4
-rw-r--r-- 1 xulong xulong 2752 Jun 19 00:49 test.txt
xulong@ubuntu:~/test$

```

步骤 7: 用 rm 命令删除 test 目录下的所有文件, 再用 rmdir 命令删除 test 目录。 (想一想有没有一条命令将目录及目录下的所有文件删除, 写出这条命令)

```
xulong@ubuntu:~/test$ rm -f test.txt
xulong@ubuntu:~/test$ ls
xulong@ubuntu:~/test$ cd ..
xulong@ubuntu:~$ rmdir test
xulong@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
xulong@ubuntu:~$ mkdir test
xulong@ubuntu:~$ >test.txt
xulong@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  test  test.txt
xulong@ubuntu:~$ cd test
xulong@ubuntu:~/test$ ls
xulong@ubuntu:~/test$ >test.txt
xulong@ubuntu:~/test$ ls
test.txt
xulong@ubuntu:~/test$ cd ..
xulong@ubuntu:~$ rm -r test
xulong@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  test.txt
xulong@ubuntu:~$
```

命令: rm -rf test

步骤 8: 使用 ps 命令查看当前系统内的进程, 并利用 man 命令获取 ps 命令的参数, 写出获取当前终端进程执行情况的 ps 命令。

```
xulong@ubuntu:~$ ps
  PID TTY          TIME CMD
 50292 pts/10    00:00:00 bash
 50298 pts/10    00:00:00 ps
xulong@ubuntu:~$ ps -man
  PID TTY          MAJFLT MINFLT   TRS   DRS   SIZE  SWAP   RSS   SHRD   LIB   DT COMMAND
 6657 tty2          33    1040    40 163963 41001   - 4948   -   -   - /usr/lib/gdm3/gdm-x-session --run-script env GNOME_SHE
 6659 tty2          80  36329 1602 278653 70064   - 61160  -   -   - /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/user/10
 6667 tty2          0   3340   168 190683 47713   - 10148  -   -   - /usr/libexec/gnome-session-binary --systemd --systemd
32341 pts/3          0   3724   705  9918  2656   - 4984   -   -   - bash
47221 pts/1          0  1193   705  9918  2656   - 5040   -   -   - bash
47271 pts/0          0  1582   705  9918  2656   - 5032   -   -   - bash
47314 pts/2          0   865   705  9918  2656   - 4964   -   -   - bash
47412 pts/4          0  1179   705  9918  2656   - 4844   -   -   - bash
47425 pts/5          0   699   705  9918  2656   - 4820   -   -   - bash
50189 pts/6          0   771   705 12998 3426   - 4852   -   -   - bash
50201 pts/6          0   371    0 14484 3621   - 3900   -   -   - su - root
50202 pts/6          0   863    0 13712 3428   - 4824   -   -   - bash
50235 pts/7          0   695   705 12998 3426   - 5024   -   -   - bash
50247 pts/8          0   760   705 12998 3426   - 4956   -   -   - bash
50282 pts/9          0   585   705 12994 3425   - 4776   -   -   - bash
50292 pts/10         0   667   705 12994 3425   - 4840   -   -   - bash
50300 pts/10         0   171   46 14421 3617   - 2924   -   -   - ps -man
xulong@ubuntu:~$
```

步骤 9: 使用 df 命令查看当前系统已安装的文件系统的空间使用情况, 记录结果。

```
xulong@ubuntu:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            1971028         0   1971028   0% /dev
tmpfs           400072      1848   398224  1% /run
/dev/sda5       19992176 8090648  10862936 43% /
tmpfs           2000352         0   2000352   0% /dev/shm
tmpfs           5120         4     5116   1% /run/lock
tmpfs           2000352         0   2000352   0% /sys/fs/cgroup
/dev/sda1       523248         4   523244   1% /boot/efi
/dev/loop0      30720     30720         0 100% /snap/snapd/8542
/dev/loop1      56320     56320         0 100% /snap/core18/1880
/dev/loop2     261760    261760         0 100% /snap/gnome-3-34-1804/36
/dev/loop3      63616     63616         0 100% /snap/gtk-common-themes/1506
/dev/loop4      51072     51072         0 100% /snap/snap-store/467
tmpfs           400068      124   399944   1% /run/user/1000
xulong@ubuntu:~$
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

步骤 10: 使用 du 命令查看用户的工作目录占用了多少空间，记录结果。

```
xulong@ubuntu:~$ du
4      ./Templates
4      ./gnupg/private-keys-v1.d
16     ./gnupg
4      ./Music
4      ./Public
4      ./local/share/gnome-settings-daemon
112    ./local/share/gvfs-metadata
4      ./local/share/nautilus/scripts
8      ./local/share/nautilus
4      ./local/share/evolution/memos/trash
8      ./local/share/evolution/memos
4      ./local/share/evolution/addressbook/system/photos
92     ./local/share/evolution/addressbook/system
4      ./local/share/evolution/addressbook/trash
100    ./local/share/evolution/addressbook
4      ./local/share/evolution/calendar/system
4      ./local/share/evolution/calendar/trash
12     ./local/share/evolution/calendar
4      ./local/share/evolution/mail/trash
8      ./local/share/evolution/mail
8      ./local/share/evolution/tasks/system
4      ./local/share/evolution/tasks/trash
16     ./local/share/evolution/tasks
148    ./local/share/evolution
4      ./local/share/webkitgtk/deviceidhashsalts/1
8      ./local/share/webkitgtk/deviceidhashsalts
4      ./local/share/webkitgtk/localstorage
4      ./local/share/webkitgtk/databases/indexeddb/v1
8      ./local/share/webkitgtk/databases/indexeddb
12     ./local/share/webkitgtk/databases
40     ./local/share/webkitgtk
4      ./local/share/flatpak/db
8      ./local/share/flatpak
4      ./local/share/icc
12     ./local/share/keyrings
4      ./local/share/ibus-table
4      ./local/share/applications
8      ./local/share/gnome-shell
4      ./local/share/sounds
52     ./local/share/xorg
```

步骤 11: 使用 free 命令查看内存资源的使用情况，记录结果。

```
xulong@ubuntu:~$ free
              total        used        free      shared  buff/cache   available
Mem:           4000704       931332       383040         3448       2686332       2783660
Swap:          945416         11104       934312
```

步骤 12: 使用 man 获取 tar 和 gzip 的帮助信息，尝试将 test 目录下的文件打包并压缩，然后到另外一目录 tmp 下解包，写出这几条命令。

```
xulong@ubuntu:~$ man tar
xulong@ubuntu:~$ man gzip
xulong@ubuntu:~$ tar -zcvf test.tar.gz text
```

man tar

man gzip

tar -zcvf test.tar.gz text

tar -zcvf test.tar.gz -c/home/tmp/

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

步骤 13: 尝试执行 “ls -l > tmp”，看看这条命令的执行会出现什么结果，解释一下这条命令。

```
xulong@ubuntu:~$ ls -l > tmp
xulong@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  tmp  test.txt
xulong@ubuntu:~$ cat tmp
total 32
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Desktop
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Documents
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Downloads
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Music
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Pictures
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Public
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Templates
drwxr-xr-x 2 xulong xulong 4096 Jun 18 23:43 Videos
-rw-rw-r-- 1 xulong xulong 0 Jun 19 01:49 tmp
-rw-rw-r-- 1 xulong xulong 0 Jun 19 01:38 test.txt
xulong@ubuntu:~$
```

步骤 14: 尝试执行 find /usr/src -name *.c -print | xargs grep “#include”，看看这条命令的执行会出现什么结果，解释一下这条命令。

```
xulong@ubuntu:~$ find /usr/src -name *.c -print|xargs grep "#include"
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <stdio.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <stdlib.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <unistd.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <string.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <errno.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include <ctype.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include "classmap.h"
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/genheaders/genheaders.c:#include "initial_sid_to_string.h"
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include <stdio.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include <stdlib.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include <unistd.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include <string.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include <linux/kconfig.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include "classmap.h"
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/selinux/mdp/mdp.c:#include "initial_sid_to_string.h"
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <stdio.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <string.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <stdlib.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <unistd.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <assert.h>
/usr/src/linux-hwe-5.8-headers-5.8.0-55/scripts/genksyms/genksyms.c:#include <stdarg.h>
```

查询 /usr/src 下到所有.c 文件并筛选出带#include 的行并打印

步骤 15: 执行 cal 和 date 命令，说说这两条指令的用途。

```
xulong@ubuntu:~$ cal
cal: setlocale: No such file or directory
      June 2021
Su Mo Tu We Th Fr Sa
                1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30

xulong@ubuntu:~$ date
Sat Jun 19 01:57:16 PDT 2021
xulong@ubuntu:~$
```

cal 显示当前月份的日历，date 显示当前所在的日期和时间。

步骤 16: 执行命令 clear 和 logout，退出系统。（想一想有没有其他的方法，写出这种方法）

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```
root@ubuntu:~# logout
$ █
```

exit

步骤 17: 执行命令 shutdown，关闭系统。（想一想有没有更简单的命令，写出这条命令）

```
xulong@ubuntu:~$ shutdown
Shutdown scheduled for Sat 2021-06-19 02:02:47 PDT, use 'shutdown -c' to cancel.
xulong@ubuntu:~$
xulong@ubuntu:~$ █
```

- 1、halt 立刻关机
- 2、poweroff 立刻关机
- 3、shutdown-h now 立刻关机 (root 用户使用)
- 4、shutdown-h 10 10 分钟后自动关机

3、实验结论

掌握 Linux 的基本命令很重要，对于基本命令知道越多使用 Linux 就越方便越顺手。

1.Find

find [PATH] [option] [action]

与时间有关的参数：

-mtime n : n 为数字，意思为在 n 天之前的“一天内”被更改过的文件；

-mtime +n : 列出在 n 天之前（不含 n 天本身）被更改过的文件名；

-mtime -n : 列出在 n 天之内（含 n 天本身）被更改过的文件名；

-newer file : 列出比 file 还要新的文件名

例如：

find /root -mtime 0 # 在当前目录下查找今天之内有改动的文件

与用户或用户组名有关的参数：

`-user name` : 列出文件所有者为 `name` 的文件

`-group name` : 列出文件所属用户组为 `name` 的文件

`-uid n` : 列出文件所有者为用户 ID 为 `n` 的文件

`-gid n` : 列出文件所属用户组为用户组 ID 为 `n` 的文件

例如:

`find /home/hadoop -user hadoop` # 在目录 `/home/hadoop` 中找出所有者为 `hadoop` 的文件

与文件权限及名称有关的参数:

`-name filename` : 找出文件名为 `filename` 的文件

`-size [+ -]SIZE` : 找出比 `SIZE` 还要大 (+) 或小 (-) 的文件

`-type TYPE` : 查找文件的类型为 `TYPE` 的文件, `TYPE` 的值主要有: 一般文件 (`f`)、设备文件 (`b`、`c`)、目录 (`d`)、连接文件 (`l`)、socket (`s`)、FIFO 管道文件 (`p`);

`-perm mode` : 查找文件权限刚好等于 `mode` 的文件, `mode` 用数字表示, 如 `0755`;

`-perm -mode` : 查找文件权限必须要全部包括 `mode` 权限的文件, `mode` 用数字表示

`-perm +mode` : 查找文件权限包含任一 `mode` 的权限的文件, `mode` 用数字表示

例如:

2.ls

`-a` : 全部的档案, 连同隐藏档 (开头为 `.` 的档案) 一起列出来~

`-A` : 全部的档案, 连同隐藏档, 但不包括 `.` 与 `..` 这两个目录, 一起列出来~

`-d` : 仅列出目录本身, 而不是列出目录内的档案数据

`-f` : 直接列出结果, 而不进行排序 (`ls` 预设会以档名排序!)

`-F` : 根据档案、目录等信息, 给予附加数据结构, 例如:

*: 代表可执行档; /: 代表目录; =: 代表 socket 档案; |: 代表 FIFO 档案;

-h : 将档案容量以人类较易读的方式(例如 GB, KB 等等)列出来;

-i : 列出 inode 位置, 而非列出档案属性;

-l : 长数据串行出, 包含档案的属性等等数据;

-n : 列出 UID 与 GID 而非使用者与群组的名称 (UID 与 GID 会在账号管理提到!)

-r : 将排序结果反向输出, 例如: 原本档名由小到大, 反向则为由大到小;

-R : 连同子目录内容一起列出来;

-S : 以档案容量大小排序!

-t : 依时间排序

--color=never : 不要依据档案特性给予颜色显示;

--color=always : 显示颜色

--color=auto : 让系统自行依据设定来判断是否给予颜色

--full-time : 以完整时间模式(包含年、月、日、时、分)输出

--time={atime,ctime} : 输出 access 时间或 改变权限属性时间 (ctime) 而非内容变更时间 (modification time)

3.rm

-f : 就是 force 的意思, 忽略不存在的文件, 不会出现警告消息

-i : 互动模式, 在删除前会询问用户是否操作

-r : 递归删除, 最常用于目录删除, 它是一个非常危险的参数

4.mv

-f : force 强制的意思, 如果目标文件已经存在, 不会询问而直接覆盖

-
- i : 若目标文件已经存在, 就会询问是否覆盖
 - u : 若目标文件已经存在, 且比目标文件新, 才会更新

5.tar

- c : 新建打包文件
- t : 查看打包文件的内容含有哪些文件名
- x : 解打包或解压缩的功能, 可以搭配-C(大写)指定解压的目录, 注意-c, -t, -x 不能同时出现在同一条命令中
- j : 通过 bzip2 的支持进行压缩/解压缩
- z : 通过 gzip 的支持进行压缩/解压缩
- v : 在压缩/解压缩过程中, 将正在处理的文件名显示出来
- f filename : filename 为要处理的文件
- C dir : 指定压缩/解压缩的目录 dir

6.gzip

gzip[参数][文件或者目录]

- a or --ascii 使用 ASCII 文字模式。
- c or --stdout or --to-stdout 把压缩后的文件输出到标准输出设备, 不去更动原始文件。
- d or --decompress or --uncompress 解开压缩文件。
- f or --force 强行压缩文件。不理睬文件名称 or 硬连接是否存在以及该文件是否为符号连接。
- h or --help 在线帮助。
- l or --list 列出压缩文件的相关信息。
- L or --license 显示版本与版权信息。

-n or --no-name 压缩文件时，不保存原来的文件名称及时间戳记。

-N or --name 压缩文件时，保存原来的文件名称及时间戳记。

-q or --quiet 不显示警告信息。

-r or --recursive 递归处理，将指定目录下的所有文件及子目录一并处理。

-S<压缩字尾字符串> or ----suffix<压缩字尾字符串> 更改压缩字尾字符串。

-t or --test 测试压缩文件是否正确无误。

-v or --verbose 显示指令执行过程。

-V or --version 显示版本信息。

-num 用指定的数字 num 调整压缩的速度，-1 or --fast 表示最快压缩方法（低压缩比），-9 or --best 表示最慢压缩方法（高压比）。系统缺省值为 6。

7. kill -signal PID

- 1: SIGHUP, 启动被终止的进程
- 2: SIGINT, 相当于输入 ctrl+c, 中断一个程序的进行
- 9: SIGKILL, 强制中断一个进程的进程
- 15: SIGTERM, 以正常的结束进程方式来终止进程
- 17: SIGSTOP, 相当于输入 ctrl+z, 暂停一个进程的进程

8.free

free [参数]

- b 以 Byte 为单位显示内存使用情况。
- k 以 KB 为单位显示内存使用情况。
- m 以 MB 为单位显示内存使用情况。
- g 以 GB 为单位显示内存使用情况。
- o 不显示缓冲区调节列。

-s<间隔秒数> 持续观察内存使用状况。

-t 显示内存总和列。

-V 显示版本信息。

9.chgrp

-c 当发生改变时输出调试信息

-f 不显示错误信息

-R 处理指定目录以及其子目录下的所有文件

-v 运行时显示详细的处理信息

--dereference 作用于符号链接的指向，而不是符号链接本身

--no-dereference 作用于符号链接本身

10.useradd

useradd [-mMnr][-c <备注>][-d <登入目录>][-e <有效期限>][-f <缓冲天数>][-g <群组>][-G <群组>][-s][-u][用户帐号]

-c #<备注> 加上备注文字。备注文字会保存在 passwd 的备注栏位中。

-d #<登入目录> 指定用户登入时的启始目录。

-D # 变更预设值。

-e # <有效期限> 指定帐号的有效期限。

-f # <缓冲天数> 指定在密码过期后多少天即关闭该帐号。

-g # <群组> 指定用户所属的群组。

-G # <群组> 指定用户所属的附加群组。

-m # 自动建立用户的登入目录。

-M # 不要自动建立用户的登入目录。

-n # 取消建立以用户名称为名的群组。

-
- r # 建立系统帐号。
 - s # 指定用户登入后所使用的 shell。
 - u # 指定用户 ID。

```
it is an exercise!  
we use vi to edit it.  
Left,down,up,right,  
_ _ _ _ _
```

实验二 简单程序设计

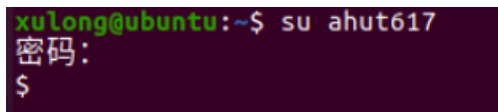
1、实验目的

- (1) 学习和掌握 vi 编辑器的基本使用方法。
- (2) 学习和掌握编写简单的 shell 程序。

2、实验内容和步骤

2.1 vi 的使用

步骤 1: 以 root 用户身份登录进入 Linux 系统，要求给出抓图。



```
xulong@ubuntu:~$ su ahut617  
密码:  
$
```

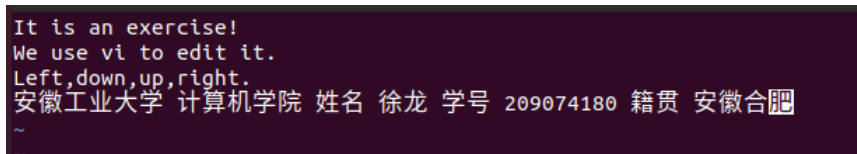
步骤 2: 启动 Vi，切换到输入模式，输入一段英文：，要求给出抓图。

It is an exercise!

We use vi to edit it.

Left,down,up,right.

安徽工业大学 计算机学院 姓名 徐龙 学号 209074180 籍贯 安徽合肥



```
It is an exercise!  
We use vi to edit it.  
Left,down,up,right.  
安徽工业大学 计算机学院 姓名 徐龙 学号 209074180 籍贯 安徽合肥  
~  
~
```

步骤 3: 尝试匹配其中的一串字符，如 it，写出这命令及执行结果或抓图。

1: 键入 Esc 进入命令模式; '/' + '要匹配的串'

结果: 如

请另外测试抓图


```
we use Vi to edit it.
abct,down,up,right,
it is an exercise!
we use Vi to edit it.
abct,down,up,right,
we use Vi to edit it.
```

1. 命令模式: yy p
2. 结果: 如

请另外测试抓图

```
It is an exercise!
We abc edit it.
We abc edit it.
Left,down,up,right.
安徽工业大学 计算机学院 姓名 徐龙 学号 209074180 籍贯 安徽合肥
~
~
~
```

步骤 6: 尝试复制/删除其中的一个单词或几个字符, 写出命令及执行结果或抓图。

1. 命令模式: 5yw, 移动光标至下一行
2. 结果: 如

请另外测试抓图

```
It is an exercise!
We use vi to edit it.
Left,down,up,right.
C
安徽工业大学 计算机学院 姓 iuj名 徐龙 学号 209074180 籍贯 安徽合肥
~
~
~
```

步骤 7: 尝试使用鼠标复制文本, 写出命令及执行结果或抓图。

1. 直接鼠标点中需要复制的内容, Ctrl+Alt+c 复制, 移动鼠标到粘贴位置, Ctrl+Alt+v 粘贴。请另外测试抓图

```
is an exercise!
wn,up,right.

ft,down,up,right.

We use vi to edit it.
Left,down,up,right.
C
安徽工业大学 计算机学院 姓 iuj名 徐龙 学号 209074180 籍贯 安徽合肥
~
~
~
```



步骤 8: 尝试存盘退出操作, 写出命令 `wq` enter, 抓图。



2.2 简单的 shell 程序编写, 要求给出程序和运行结果

第一题: 请详细查看如下几个数字的规律, 并使用 shell 脚本输出后面的 20 个数字。

10 31 53 77 105 141 结果:

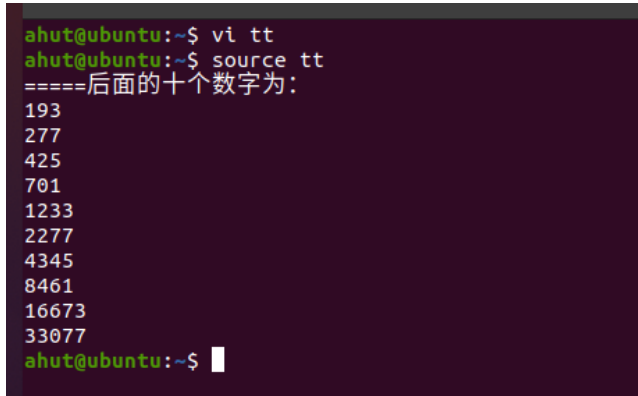
```
#!/bin/bash
##
echo "====后面的十个数字为: "
i=10
for n in `seq 0 15`
do
```

注意不要雷同 banban
<https://github.com/dream4789/Computer-learning-resources.git>

```

        a=$((2**$n))
        b=$((20+$a))
        let "i+= $b"
        if [$n -lt 15]&&[$n -ge 5]
        then
            echo $i
        fi
    done

```



```

ahut@ubuntu:~$ vi tt
ahut@ubuntu:~$ source tt
====后面的十个数字为:
193
277
425
701
1233
2277
4345
8461
16673
33077
ahut@ubuntu:~$

```

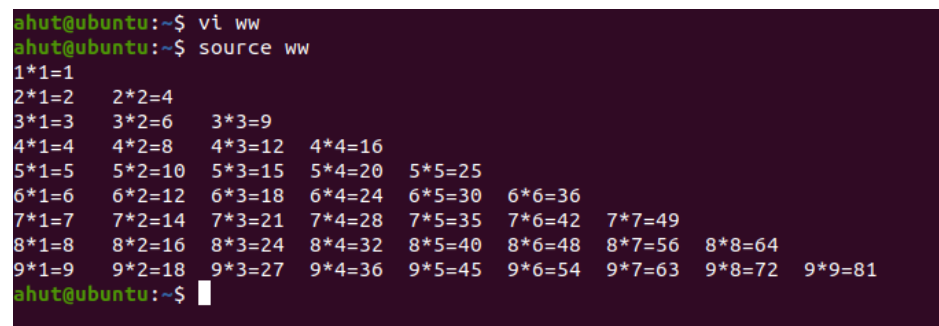
第二题：打印 99 乘法表，要求对齐。

结果：

```

#!/bin/bash
for ((i=1; i<=9; i++)); do
    for ((j=1; j<=i; j++))
    do
        printf "%i*$j=$((i*$j))\t"
    done
    echo ""
done

```



```

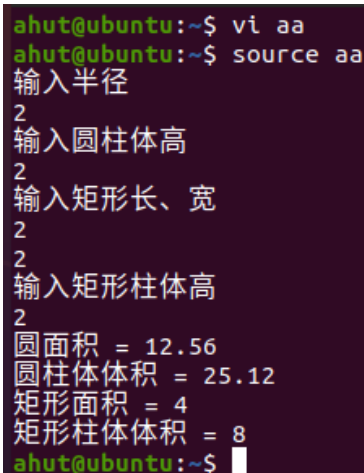
ahut@ubuntu:~$ vi ww
ahut@ubuntu:~$ source ww
1*1=1
2*1=2   2*2=4
3*1=3   3*2=6   3*3=9
4*1=4   4*2=8   4*3=12  4*4=16
5*1=5   5*2=10  5*3=15  5*4=20  5*5=25
6*1=6   6*2=12  6*3=18  6*4=24  6*5=30  6*6=36
7*1=7   7*2=14  7*3=21  7*4=28  7*5=35  7*6=42  7*7=49
8*1=8   8*2=16  8*3=24  8*4=32  8*5=40  8*6=48  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
ahut@ubuntu:~$

```

第三题: 编写 calCircleArea, calRectArea, calCircleCylinderVolume (计算圆柱体体积)、calRectCylinderVolume (计算矩形柱体体积), 体积函数要求调用面积函数。

结果:

```
#!/bin/bash
echo "输入半径"
read r
echo "输入圆柱体高"
read h
echo "输入矩形长、宽"
read a
read b
echo "输入矩形柱体高"
read c
calCircleArea()
{
    areal=`echo 3.14*$1*$1 | bc`
}
calCircleCylinderVolume()
{
    v1=`echo $1*$2 | bc`
}
calRectArea()
{
    area2=`echo $1*$2 | bc`
}
calRectCylinderVolume()
{
    v2=`echo $1*$2 | bc`
}
calCircleArea $r
calCircleCylinderVolume $areal $h
calRectArea $a $b
calRectCylinderVolume $area2 $c
echo "圆面积 = $areal"
echo "圆柱体体积 = $v1"
echo "矩形面积 = $area2"
echo "矩形柱体体积 = $v2"
```



```
ahut@ubuntu:~$ vi aa
ahut@ubuntu:~$ source aa
输入半径
2
输入圆柱体高
2
输入矩形长、宽
2
2
输入矩形柱体高
2
圆面积 = 12.56
圆柱体体积 = 25.12
矩形面积 = 4
矩形柱体体积 = 8
ahut@ubuntu:~$
```

3、实验结论

vi 编辑器使用很方便, 掌握 vi 的基本命令是灵活使用 vi 的基础, shell 可以作为 Linux 的高级语言来编写我们需要的代码, 是我们进一步了解使用 Linux 的有力工具。系统管理员的重要工作就是修改与设定某些重要软件的配置文件, 因此至少得要学会一种以上的命令界面文本编辑器。在所有 Linux distributions 上都会有的一套文字编辑器就是 vi, 而且很多软件默认的也是使用 vi 为他们编辑接口。实验分析通过这次实验, 我懂得了定义 SHELL 程序设计中的变量, 掌握了 SHELL 程序中 LINUX 基本命令的使用, 理解了 Shell 程序的设计方法, 并

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

且熟悉了 Shell 程序编辑、运行、调试方法与过程。这为我日后继续深入学习 Linux 课程打下基础。