



中山大学南方学院

电气与计算机工程学院

课程名称: Ubuntu Linux 操作系统使用教程

实验题目: Shell 编程

2019 年 04 月 11 日

附：实验报告

专业： 计算机科学与技术 年级： 2016 完成日期： 2019-04-11

学号： 162011802 姓名： 林宏健 成 绩： _____

一、实验目的

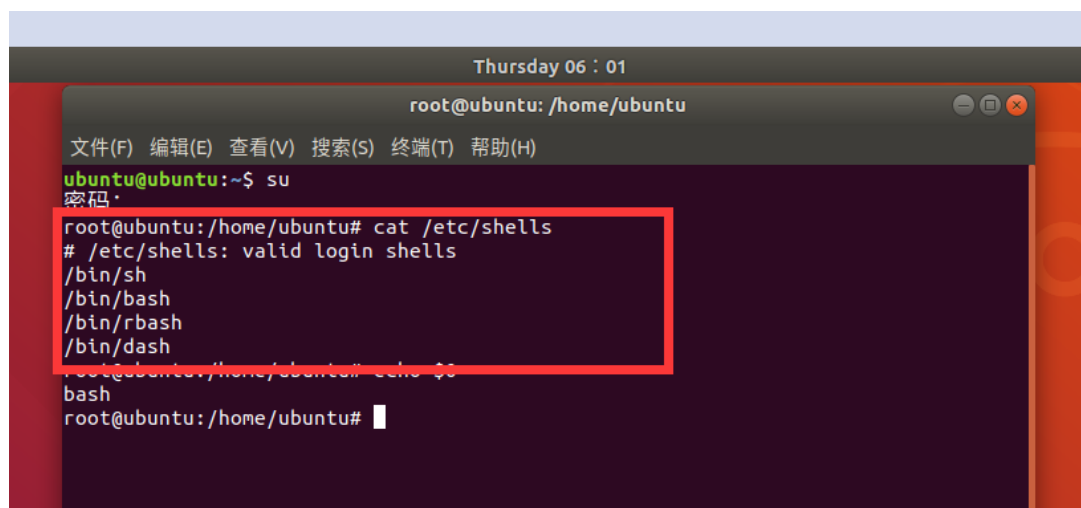
- 1、 掌握文本编辑器 vi 和 gedit 的使用方法
- 2、 熟悉环境变量和用户自定义变量的内容和应用
- 3、 掌握 Shell 脚本的基本语法
- 4、 能熟练应用分支结构和循环结构编写程序

二、实验过程

1. Shell 的基本使用（课本 174 页实训任务 1）

截图：

1、 查看目前系统支持的 Shell 版本



```
Thursday 06 : 01
root@ubuntu: /home/ubuntu
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ubuntu@ubuntu:~$ su
密码:
root@ubuntu:/home/ubuntu# cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/bin/rbash
/bin/dash
root@ubuntu:/home/ubuntu# echo $SHELL
bash
root@ubuntu:/home/ubuntu#
```

2、 查看目前使用的 Shell 版本

```

root@ubuntu: /home/ubuntu
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ubuntu@ubuntu:~$ su
密码:
root@ubuntu:/home/ubuntu# cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/bin/rbash
/bin/dash
root@ubuntu:/home/ubuntu# echo $0
bash
root@ubuntu:/home/ubuntu#
root@ubuntu:/home/ubuntu# echo $0
bash
root@ubuntu:/home/ubuntu# bash --version
GNU bash, 版本 4.4.19(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2016 Free Software Foundation, Inc.
许可证 GPLv3+: GNU GPL 许可证第三版或者更新版本 <http://gnu.org/licenses/gpl.htm
l>
本软件是自由软件，您可以自由地更改和重新发布。
在法律许可的情况下特此明示，本软件不提供任何担保。
root@ubuntu:/home/ubuntu#

```

3、显示目前系统主要的环境变量

```

Thursday 06.16
root@ubuntu: /home/ubuntu
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
root@ubuntu:/home/ubuntu# set
BASH=/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extquote:force_
ignore:histappend:hostcomplete:interactive_comments:progcomp:promptvars:sourcepa
th
BASH_ALIASES=()
BASH_ARGC=()
BASH_ARGV=()
BASH_CMDS=()
BASH_LINENO=()
BASH_SOURCE=()
BASH_VERSIONINFO=([0]="4" [1]="4" [2]="19" [3]="1" [4]="release" [5]="x86_64-pc-lin
ux-gnu")
BASH_VERSION='4.4.19(1)-release'
CLUTTER_IM_MODULE=xim
COLORTERM=truecolor
COLUMNS=80
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
DESKTOP_SESSION=ubuntu
DIRSTACK=()
DISPLAY=:0
EUID=0
GDMSESSION=ubuntu
GNOME_DESKTOP_SESSION_ID=this-is-deprecated

```

4、临时改变 Shell 版本为 csh

```
XDG_SESSION_TYPE=x11
XDG_VTNR=2
XMODIFIERS=@im=ibus
_=env
command_not_found_handle ()
{
    if [ -x /usr/lib/command-not-found ]; then
        /usr/lib/command-not-found -- "$1";
        return $?;
    else
        if [ -x /usr/share/command-not-found/command-not-found ]; then
            /usr/share/command-not-found/command-not-found -- "$1";
            return $?;
        else
            printf "%s: command not found\n" "$1" 1>&2;
            return 127;
        fi;
    fi;
}
root@ubuntu:/home/ubuntu# csh
#
```

2. 编写一个 Shell 程序判断/etc/shadow 是否为文件
代码:

```
#!/bin/bash
if [ -f /etc/shadow ]; then
    echo "File exists!"
else
    echo "File not exist!"
fi
#end
```

运行截图:

```
(4级) ** (gedit:33261): WARNING **: 07:04:30.093: Set document metadata failed: 不支持
设置属性 metadata::gedit-spell-language


** (gedit:33261): WARNING **: 07:04:30.093: Set document metadata failed: 不支持
设置属性 metadata::gedit-encoding

** (gedit:33261): WARNING **: 07:04:33.326: Set document metadata failed: 不支持
设置属性 metadata::gedit-position
root@ubuntu:/# bash test.sh
File exists!
root@ubuntu:/# sudo gedit test.sh

** (gedit:33276): WARNING **: 07:05:13.961: Set document metadata failed: 不支持
设置属性 metadata::gedit-position
root@ubuntu:/# bash test.sh
File exists!
root@ubuntu:/#
```

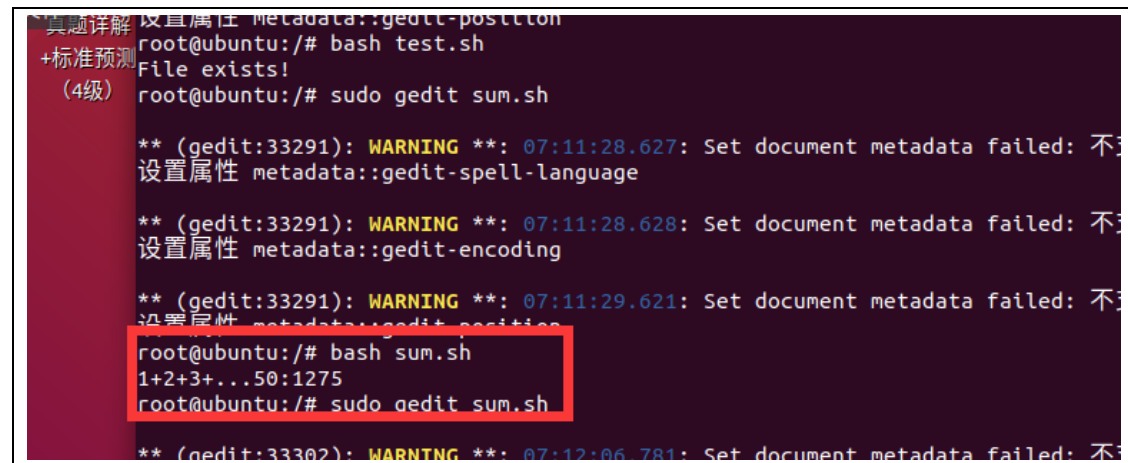
3. 编写一个 Shell 程序计算 1+2+...+50 的和

代码:



```
#!/bin/bash
sum=0
i=1
for i in `seq 1 50`
do
    sum=$((sum+i))
    i=$((i+1))
done
echo 1+2+3+...50:$sum
#end
```

截图:



```
root@ubuntu:/# bash test.sh
File exists!
root@ubuntu:/# sudo gedit sum.sh

** (gedit:33291): WARNING **: 07:11:28.627: Set document metadata failed: 不
设置属性 metadata::gedit-spell-language

** (gedit:33291): WARNING **: 07:11:28.628: Set document metadata failed: 不
设置属性 metadata::gedit-encoding

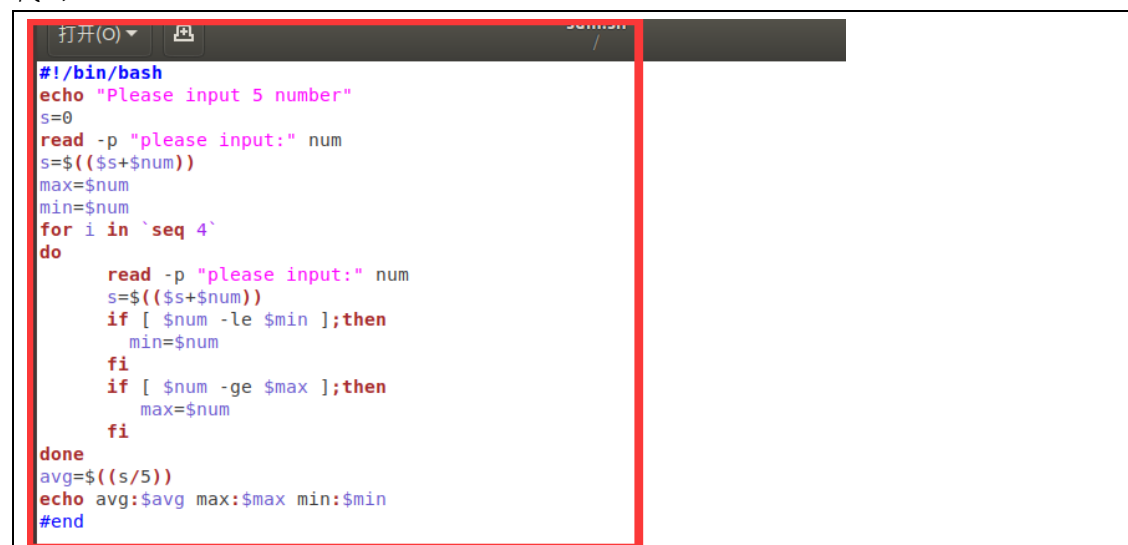
** (gedit:33291): WARNING **: 07:11:29.621: Set document metadata failed: 不
设置属性 metadata::gedit-position

root@ubuntu:/# bash sum.sh
1+2+3+...50:1275
root@ubuntu:/# sudo gedit sum.sh

** (gedit:33302): WARNING **: 07:12:06.781: Set document metadata failed: 不
```

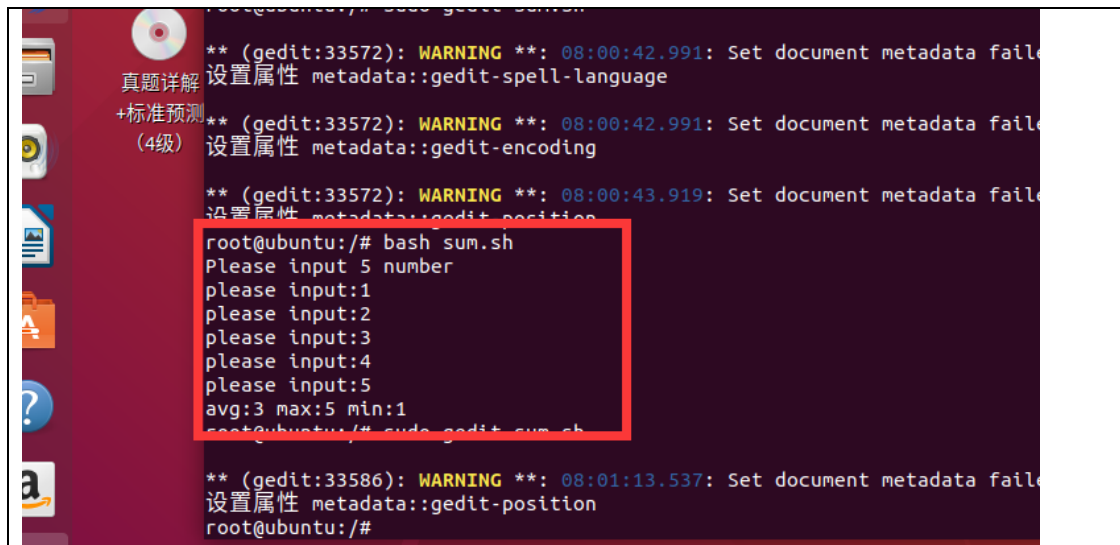
4. 编写一个 Shell 程序允许在键盘输入 5 个整数，并显示其中的最大数，最小数和平均数

代码:



```
#!/bin/bash
echo "Please input 5 number"
s=0
read -p "please input:" num
s=$((s+num))
max=num
min=num
for i in `seq 4`
do
    read -p "please input:" num
    s=$((s+num))
    if [ $num -le $min ];then
        min=num
    fi
    if [ $num -ge $max ];then
        max=num
    fi
done
avg=$((s/5))
echo avg:$avg max:$max min:$min
#end
```

截图：



```
root@ubuntu:~# bash sum.sh
Please input 5 number
please input:1
please input:2
please input:3
please input:4
please input:5
avg:3 max:5 min:1
root@ubuntu:~#
```

三、简答题

1. 在 vi 中编辑模式和指令模式有什么不同？

1、指令模式（command mode）—执行命令 在该模式中，可以输入命令来执行许多种功能。控制屏幕光标的移动，字符、字或行的删除，移动复制某区段及进入 Insert mode 下，或者到 last line mode。

2、编辑模式（Insert mode）—输入文本 vi 被运行时，通常处在命令模式下，键入以下命令 可以使 vi 退出命令模式，进入输入模式：I（i）、A（a）、O（o）。

2. 用户在编写 Shell 程序后，怎么获取程序的“执行”权限？

用户可以执行 `chmod a+x username.sh` 来设置执行的权限

四、实验总结和体会

通过这节课的学习，我对于 vi 的命令模式有了很深的了解，也学会了使用 shell 编写脚本并执行，shell 的语法相比 java 和 c 语言一类的会简洁很多，所以还是很好上手的。