

姓名：吴涵

学号：202101000720

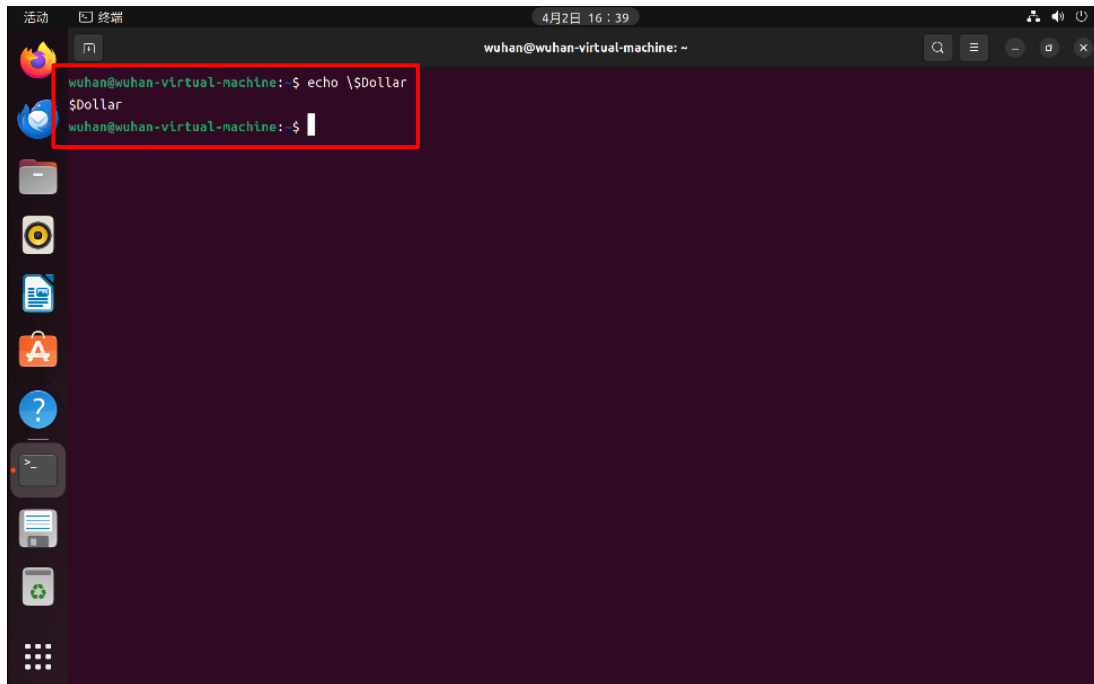
成绩：

## 实验 13 批处理操作接口 3：引用与命令替换

1、输出字符串“\$Dollar”

命令：echo \ \$Dollar

结果：

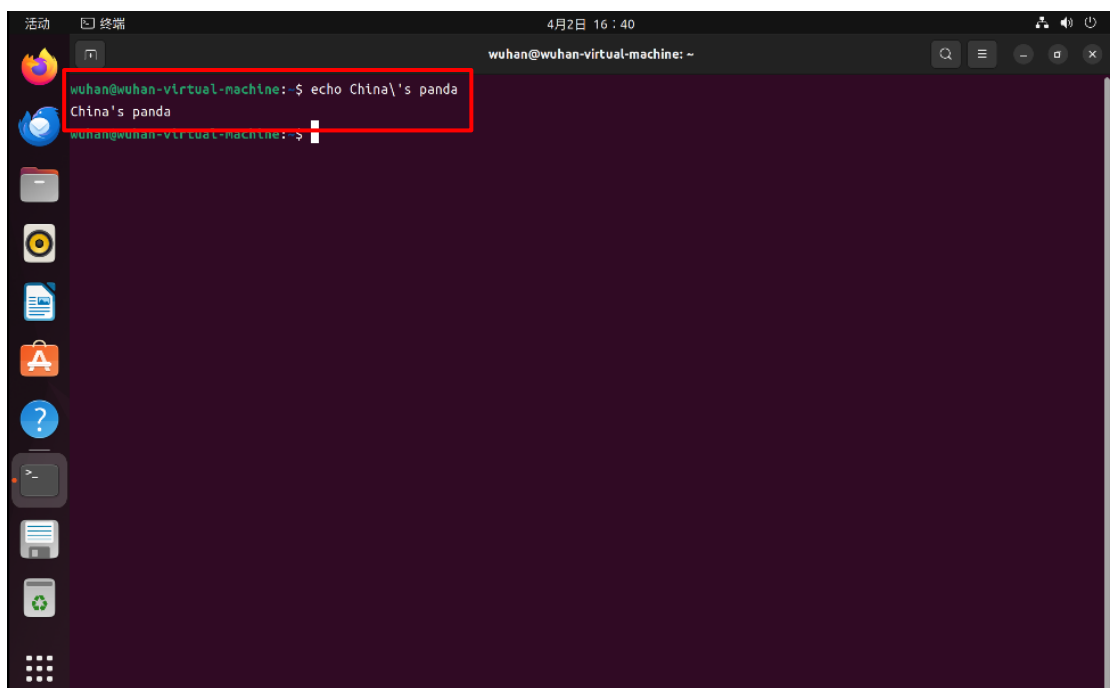


```
wuhan@wuhan-virtual-machine:~$ echo \ $Dollar
$Dollar
wuhan@wuhan-virtual-machine:~$
```

2、输出字符串“China's panda”

命令：echo China\ 's panda

结果：



```
wuhan@wuhan-virtual-machine:~$ echo China\ 's panda
China's panda
wuhan@wuhan-virtual-machine:~$
```

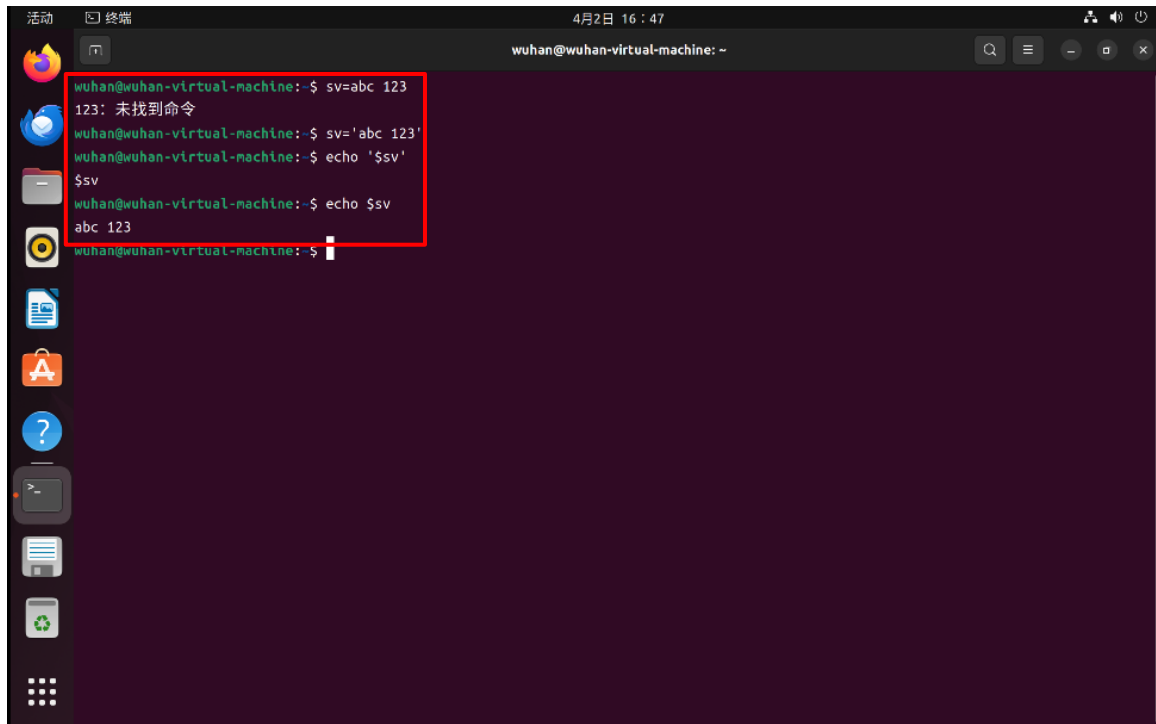
### 3、比较对错

sv=abc 123

sv='abc 123'

echo '\$sv'

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月2日 16:47'. The terminal shows the following commands and output:

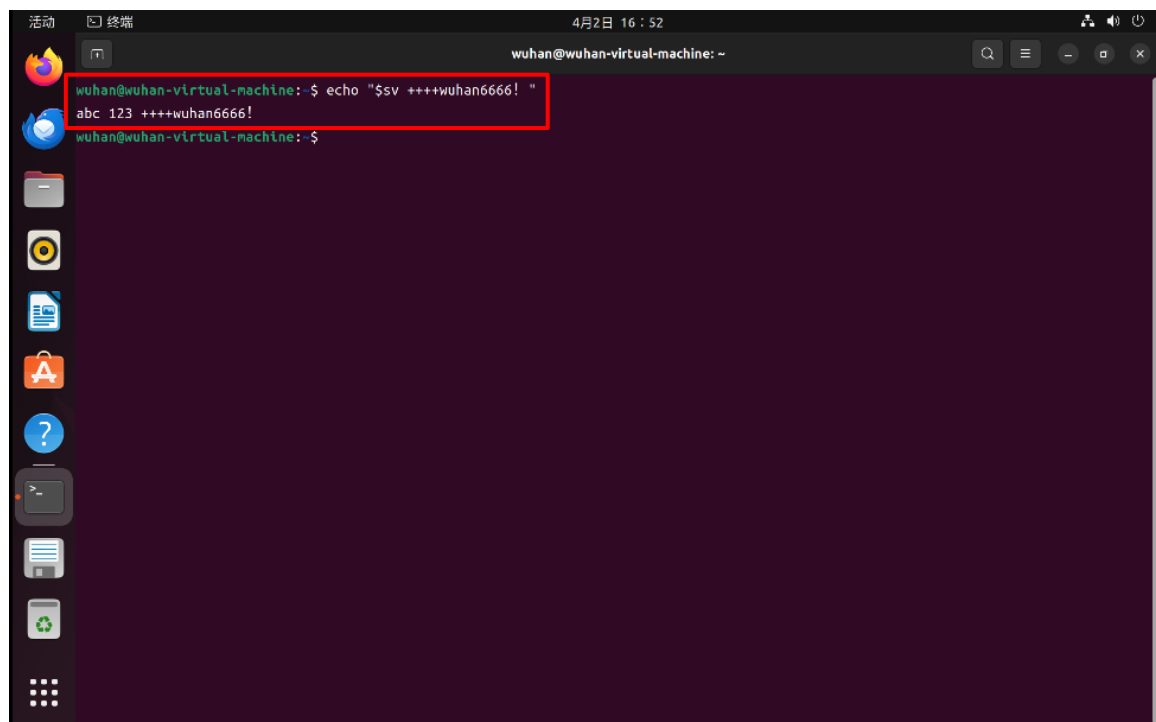
```
wuhan@wuhan-virtual-machine:~$ sv=abc 123
123: 未找到命令
wuhan@wuhan-virtual-machine:~$ sv='abc 123'
wuhan@wuhan-virtual-machine:~$ echo '$sv'
$sv
wuhan@wuhan-virtual-machine:~$ echo $sv
abc 123
wuhan@wuhan-virtual-machine:~$
```

The first two lines of the terminal output are highlighted with a red box.

### 4、引用

命令: echo "\$sv ++++加上另一些字符!"

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月2日 16:52'. The terminal shows the following command and output:

```
wuhan@wuhan-virtual-machine:~$ echo "$sv ++++wuhan6666!"
abc 123 ++++wuhan6666!
wuhan@wuhan-virtual-machine:~$
```

The command and its output are highlighted with a red box.

## 5、命令比较

命令: `"ls\"`

命令: `'ls'`

结果:

```
wuhan@wuhan-virtual-machine:~$ "ls"
c Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
wuhan@wuhan-virtual-machine:~$ 'ls'
c Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
```

命令: `ab='ls'`

`ab`

命令: `$ab`

命令: `"$ab"`

结果:

```
wuhan@wuhan-virtual-machine:~$ ab='ls'
wuhan@wuhan-virtual-machine:~$ ab
找不到命令“ab”，但可以通过以下软件包安装它：
sudo apt install apache2-utils
wuhan@wuhan-virtual-machine:~$ $ab
c Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
wuhan@wuhan-virtual-machine:~$ "$ab"
c Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
```

命令: `echo "This is ${ab}"`

命令: `echo "This is $(ab)"`

结果:

```
wuhan@wuhan-virtual-machine:~$ echo "This is ${ab}"
This is ls
wuhan@wuhan-virtual-machine:~$ echo "This is $(ab)"
找不到命令“ab”，但可以通过以下软件包安装它：
sudo apt install apache2-utils
This is
```

命令: echo "a string..." \$(\$ab)

命令: echo "a string..." \$(\$ab)"

结果:

```
wuhan@wuhan-virtual-machine:~$ echo "a string..." $($ab)
a string... c Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
wuhan@wuhan-virtual-machine:~$ echo "a string..." $($ab)"
a string... c
Desktop
Documents
Downloads
Music
Pictures
Public
shel
snap
Templates
Videos
wuhan@wuhan-virtual-machine:~$
```

命令: echo "This is \$(ls)"

命令: echo "a string..." \$(ls)

结果:

```
wuhan@wuhan-virtual-machine:~$ echo "This is $(ls)"
This is c1
Desktop
Documents
Downloads
Music
Pictures
Public
shel
snap
Templates
Videos
wuhan@wuhan-virtual-machine:~$ echo "a string..." $(ls)
a string... c1 Desktop Documents Downloads Music Pictures Public shel snap Templates Videos
wuhan@wuhan-virtual-machine:~$
```

命令: echo "a string..." \$(./c1)"

结果:

```
wuhan@wuhan-virtual-machine:~/c1$ echo "a string..." $(./c1)"
a string... hello world!
吴涵202101000720
wuhan@wuhan-virtual-machine:~/c1$
```

命令: `echo 'It\' s a dog'`

命令: `echo "It's a dog"`

结果:

```
wuhan@wuhan-virtual-machine:~/c1$ echo 'It\' s a dog'
It' s a dog
wuhan@wuhan-virtual-machine:~/c1$ echo "It's a dog"
It's a dog
wuhan@wuhan-virtual-machine:~/c1$
```

6、将命令 `date` 执行结果赋予变量 `date1`

命令: `date1='date'`

结果:

```
wuhan@wuhan-virtual-machine: ~
wuhan@wuhan-virtual-machine:~$ date
2024年 04月 02日 星期二 17:32:03 CST
wuhan@wuhan-virtual-machine:~$ date1='date'
wuhan@wuhan-virtual-machine:~$ echo $date1
2024年 04月 02日 星期二 17:32:14 CST
wuhan@wuhan-virtual-machine:~$
```

7、将命令 `date` 执行结果赋予变量 `date2`

命令:

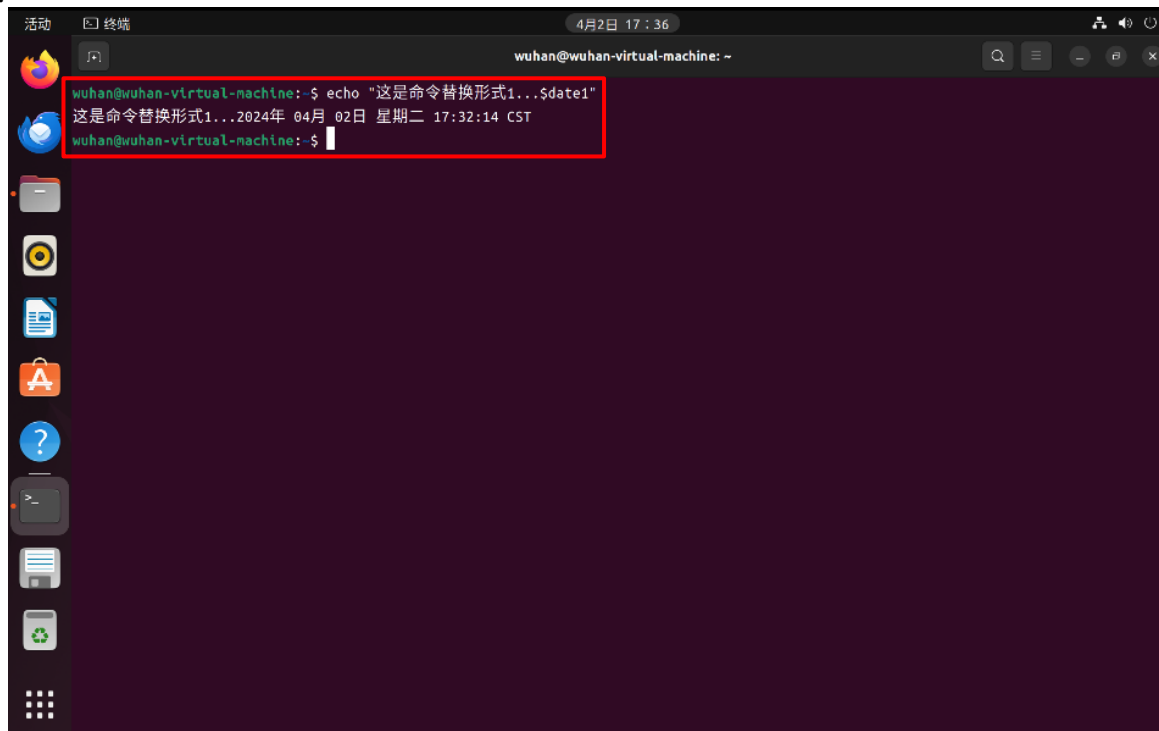
结果:

```
wuhan@wuhan-virtual-machine: ~
wuhan@wuhan-virtual-machine:~$ date
2024年 04月 02日 星期二 17:32:03 CST
wuhan@wuhan-virtual-machine:~$ date1='date'
wuhan@wuhan-virtual-machine:~$ echo $date1
2024年 04月 02日 星期二 17:32:14 CST
wuhan@wuhan-virtual-machine:~$ date2=$(date)
wuhan@wuhan-virtual-machine:~$ echo $date2
2024年 04月 02日 星期二 17:33:25 CST
wuhan@wuhan-virtual-machine:~$
```

## 8、命令替换

命令: `echo "这是命令替换形式 1...$date1"`

结果:



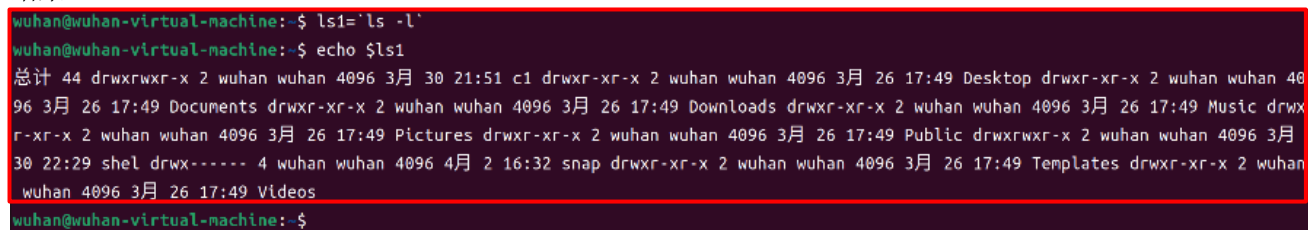
A terminal window titled 'wuhan@wuhan-virtual-machine: ~' showing the execution of the command `echo "这是命令替换形式 1...$date1"`. The output is `这是命令替换形式 1...2024年 04月 02日 星期二 17:32:14 CST`. The terminal has a dark purple background and a sidebar with application icons on the left.

## 9、命令替换

命令: `ls1='ls -l'`

`echo $ls1`

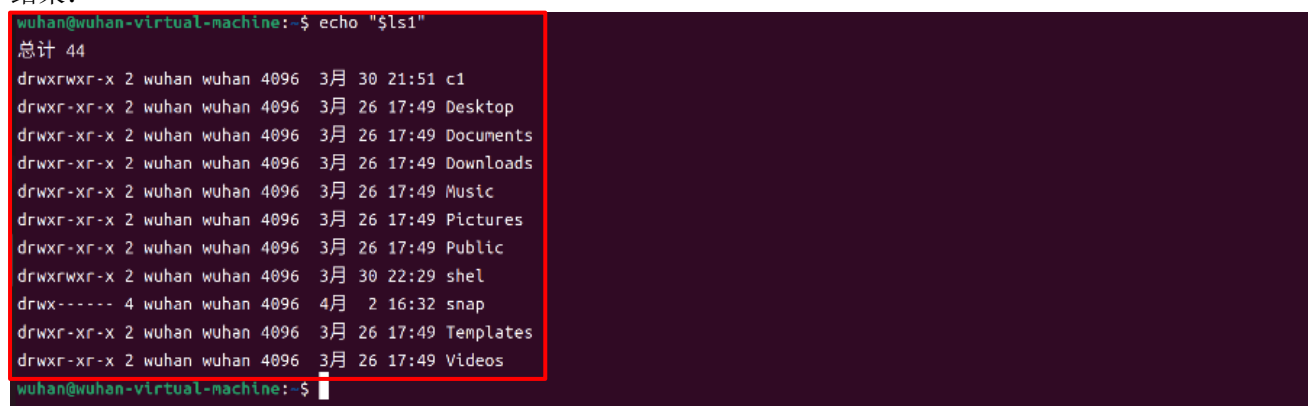
结果:



A terminal window showing the command `ls1='ls -l'` followed by `echo $ls1`. The output is a detailed `ls -l` listing of the directory contents, including files like Desktop, Documents, Downloads, Music, Pictures, Public, Templates, and Videos, along with their permissions, owners, sizes, and timestamps.

命令: `echo "$ls1"`

结果:



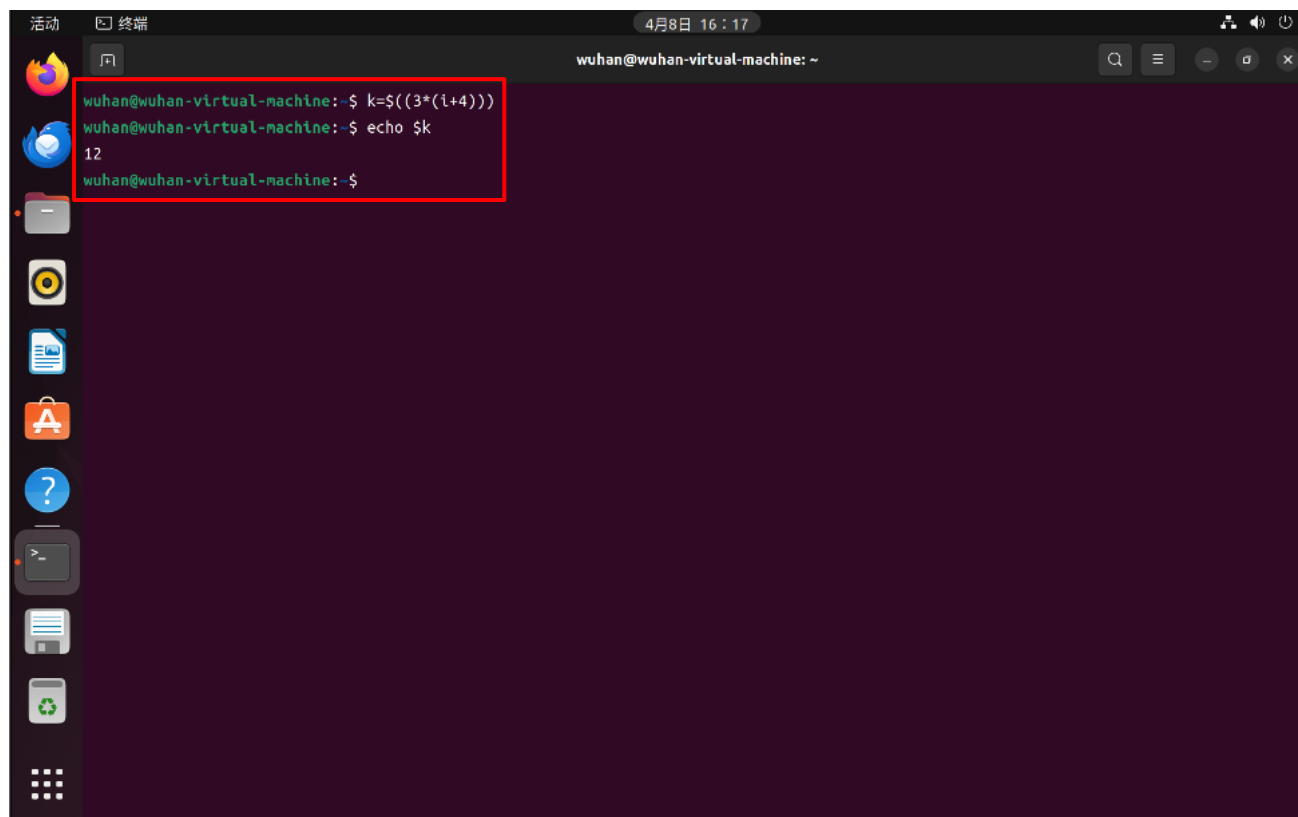
A terminal window showing the command `echo "$ls1"`. The output is the same `ls -l` listing as in the previous block, but it is enclosed in double quotes, which is the result of the command substitution.

## 10、命令替换

命令: `k=$((3*(i+4)))`

`echo $k`

结果:



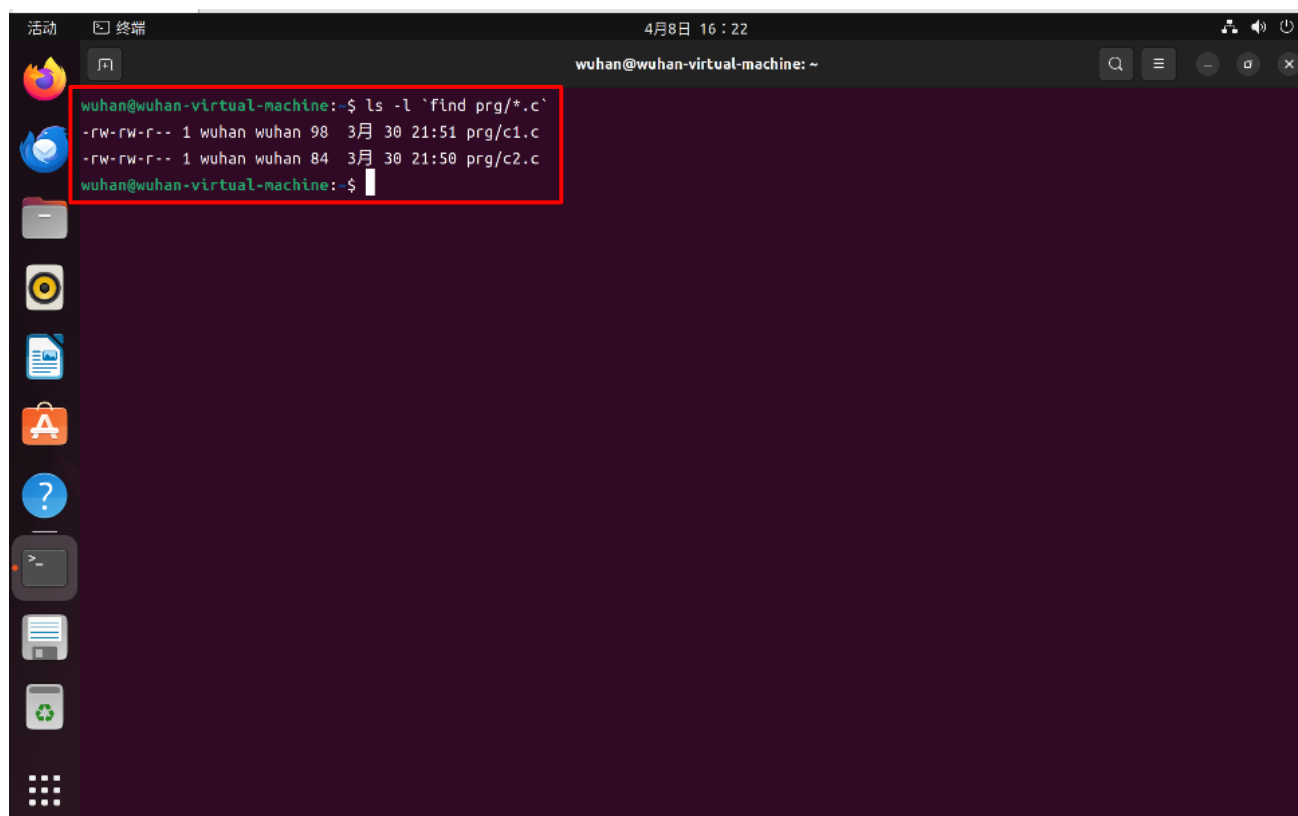
A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月8日 16:17'. The terminal shows the following commands and output:

```
wuhan@wuhan-virtual-machine:~$ k=$((3*(i+4)))
wuhan@wuhan-virtual-machine:~$ echo $k
12
wuhan@wuhan-virtual-machine:~$
```

## 11、命令替换

命令: `ls -l `find prg/*.c``

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月8日 16:22'. The terminal shows the following commands and output:

```
wuhan@wuhan-virtual-machine:~$ ls -l `find prg/*.c`
-rw-rw-r-- 1 wuhan wuhan 98  3月 30 21:51 prg/c1.c
-rw-rw-r-- 1 wuhan wuhan 84  3月 30 21:50 prg/c2.c
wuhan@wuhan-virtual-machine:~$
```

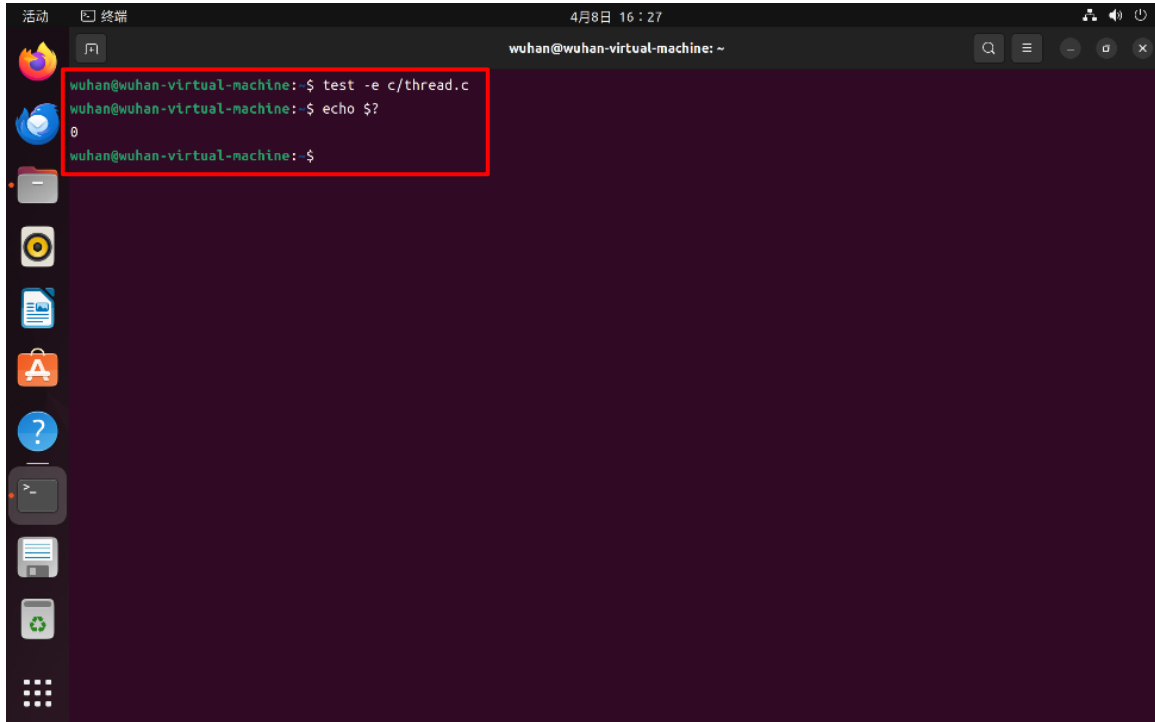
## 实验 14 批处理操作接口 4：测试、if 判断

1、测试文件 c/thread.c 是否存在

命令：test -e c/thread.c

echo \$?

结果：



```
wuhan@wuhan-virtual-machine:~$ test -e c/thread.c
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$
```

2、编写 Shell 脚本文件 testrxw.sh，对文件拥有的属性进行判断

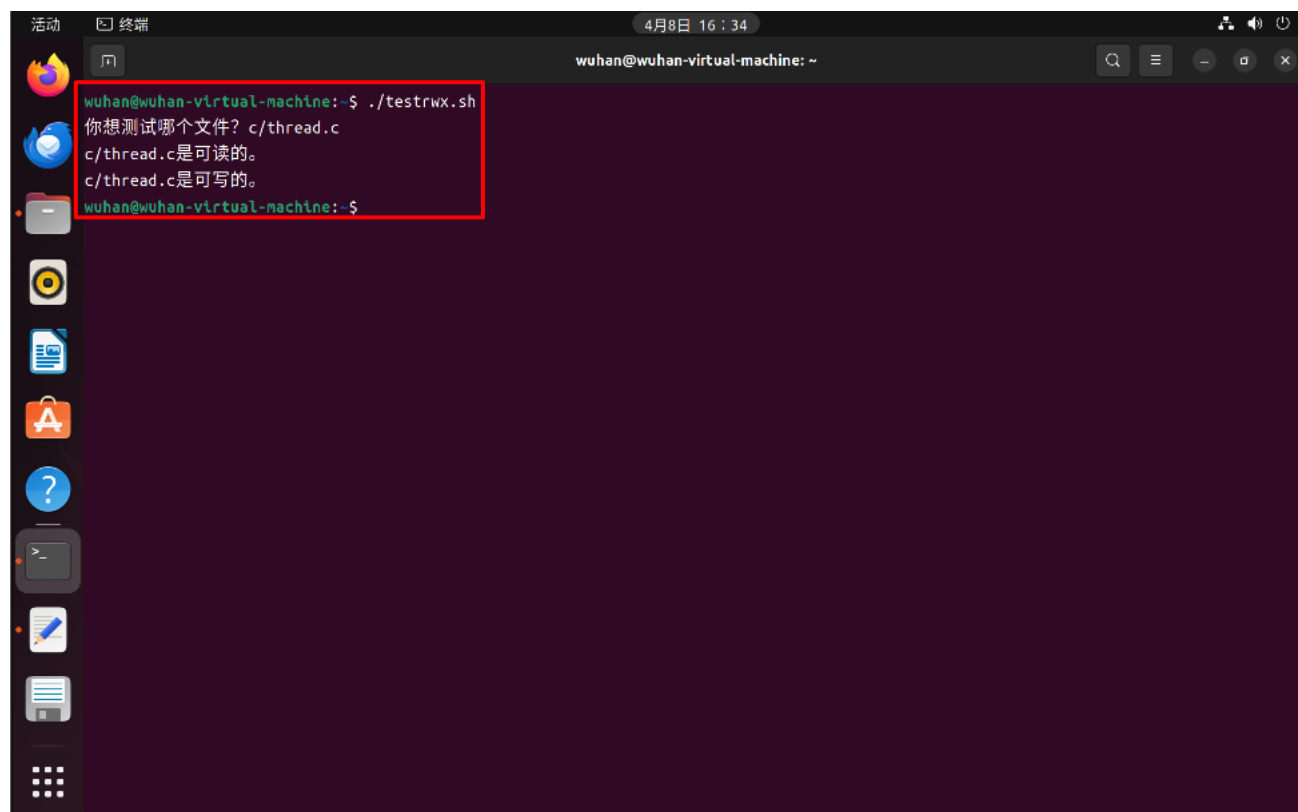
脚本：

```
#!/bin/bash
read -p "你想测试哪个文件？" filename
if [ ! -e "$filename" ]; then
    echo "这个文件不存在。"
    #exit 1
fi
if [ -r "$filename" ]; then
    echo "$filename 是可读的。"
    #exit 1
fi
if [ -w "$filename" ]; then
    echo "$filename 是可写的。"
    #exit 1
fi
if [ -x "$filename" ]; then
    echo "$filename 是可执行的。"
    #exit 1
fi
```



命令: `../testrx.sh`

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月8日 16:34'. The terminal shows the execution of the script `../testrx.sh`. The output is as follows:

```
wuhan@wuhan-virtual-machine:~$ ./testrx.sh
你想测试哪个文件? c/thread.c
c/thread.c是可读的。
c/thread.c是可写的。
wuhan@wuhan-virtual-machine:~$
```

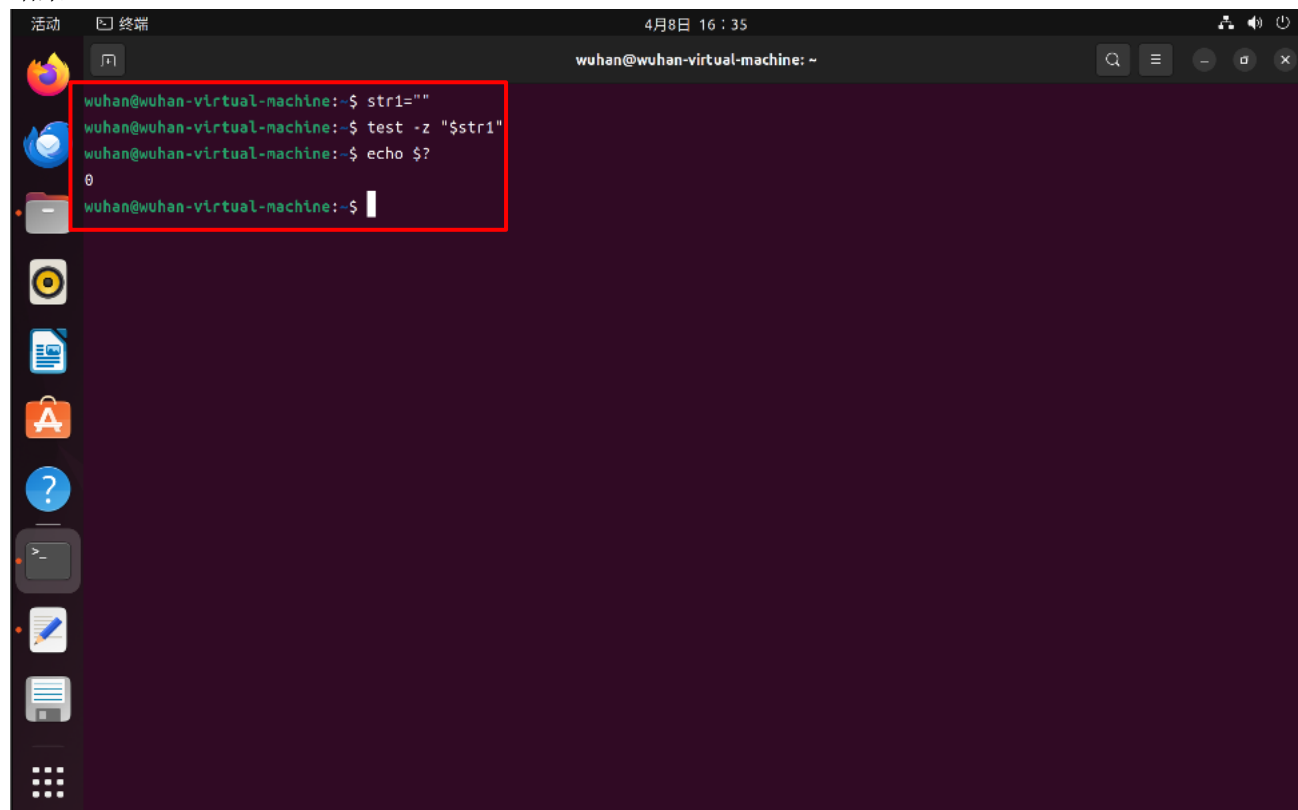
3、测试字符串 `str1` 是否为空，为空则返回 0

命令: `str1=""`

`test -z "$str1"`

`echo $?`

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月8日 16:35'. The terminal shows the execution of the following commands:

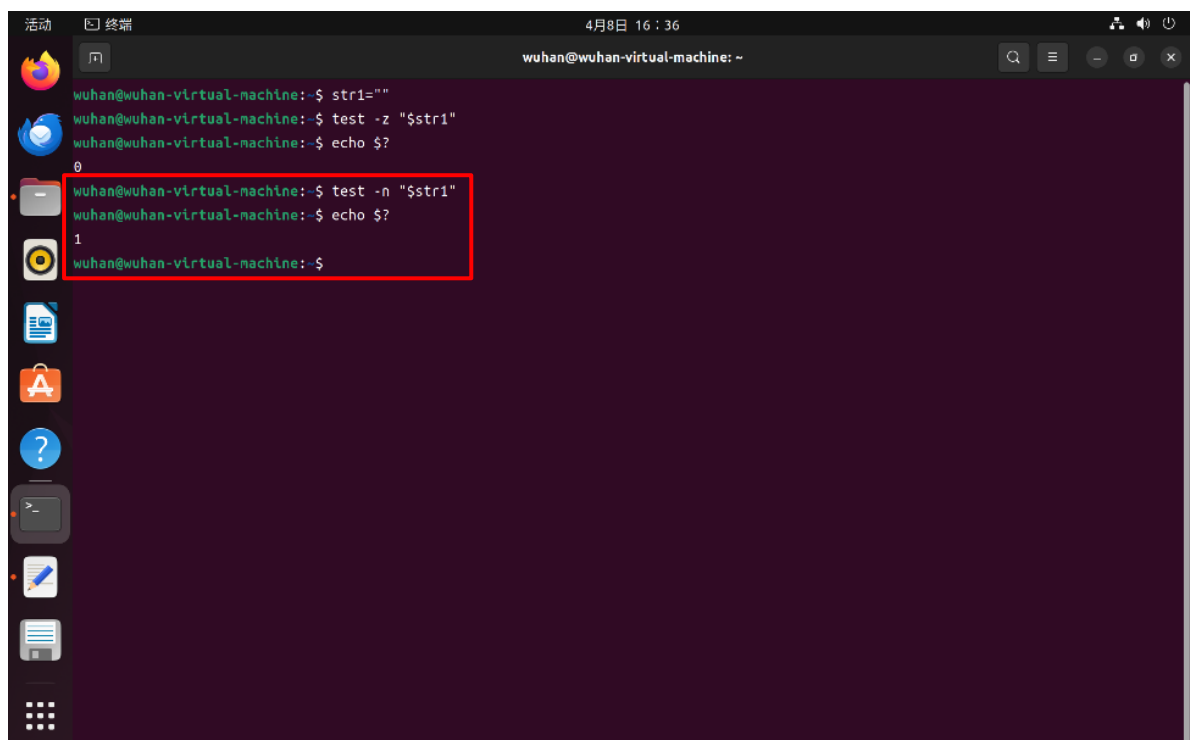
```
wuhan@wuhan-virtual-machine:~$ str1=""
wuhan@wuhan-virtual-machine:~$ test -z "$str1"
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$
```

4、测试字符串 `str1` 是否为空，非空则返回 0，为空返回非 0

命令: `test -n "$str1"`

`echo $?`

结果:



```
wuhan@wuhan-virtual-machine:~$ str1=""
wuhan@wuhan-virtual-machine:~$ test -z "$str1"
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$ test -n "$str1"
wuhan@wuhan-virtual-machine:~$ echo $?
1
wuhan@wuhan-virtual-machine:~$
```

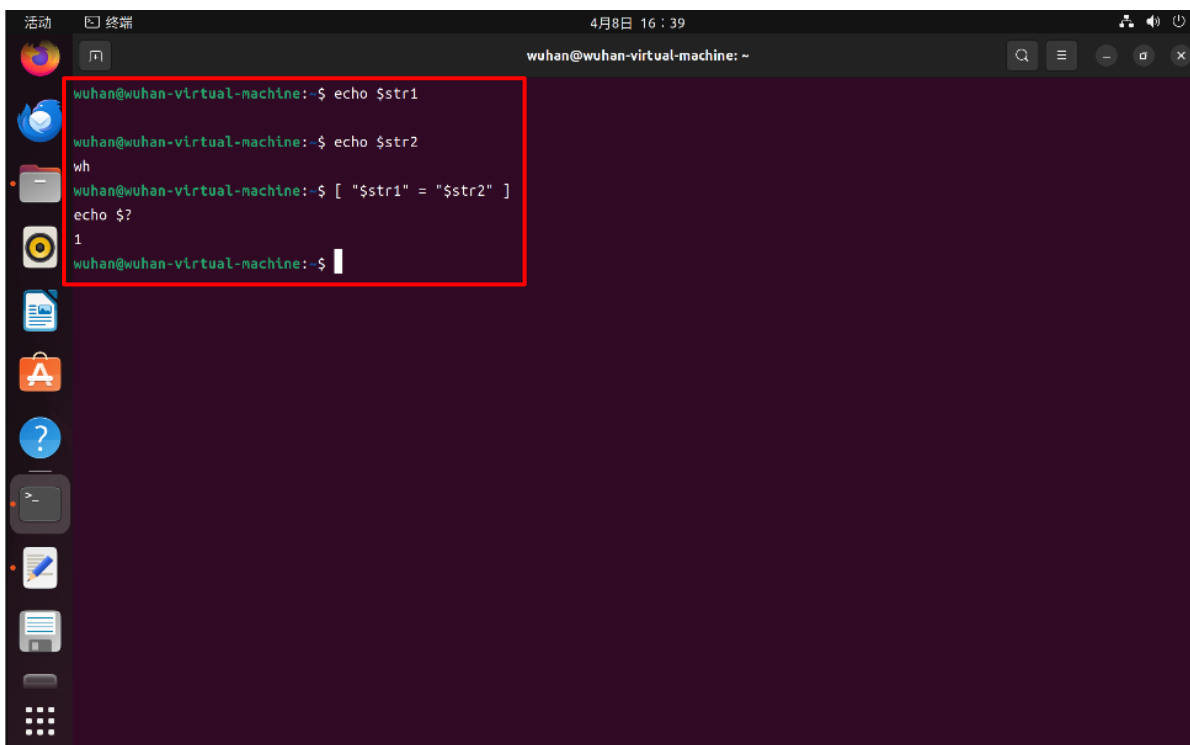
The screenshot shows a terminal window with a dark background. The user sets `str1=""`, then runs `test -z "$str1"` which returns 0. Then they run `test -n "$str1"` which returns 1. The last two lines are highlighted with a red box.

5、比较 `str1` 与 `str2` 是否相同，相同则返回 0，否则返回非 0

命令: `["$str1" = "$str2"]`

`echo $?`

结果:



```
wuhan@wuhan-virtual-machine:~$ echo $str1
wuhan@wuhan-virtual-machine:~$ echo $str2
wh
wuhan@wuhan-virtual-machine:~$ [ "$str1" = "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
1
wuhan@wuhan-virtual-machine:~$
```

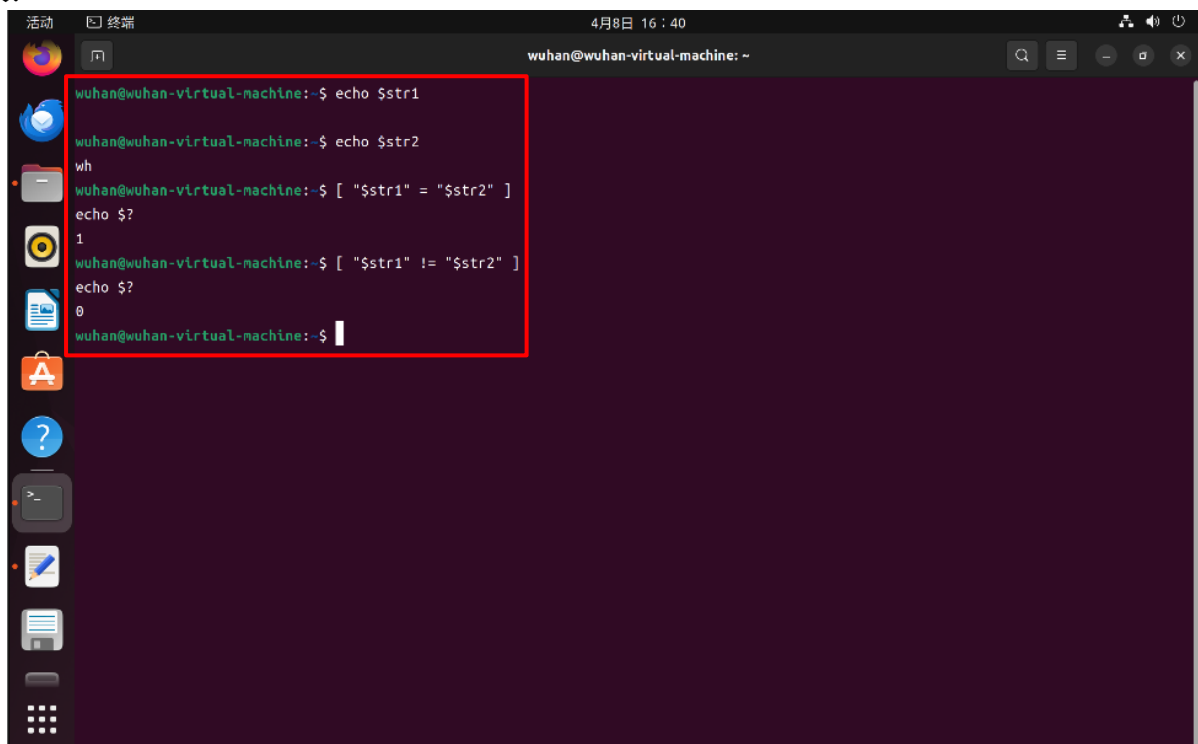
The screenshot shows a terminal window. The user echoes `$str1` (which is empty) and `$str2` (which is "wh"). Then they run the comparison command `["$str1" = "$str2"]`, which returns 1. The last two lines are highlighted with a red box.

6、比较 str1 与 str2 是否不同，不同则返回 0

命令: [ "\$str1" != "\$str2" ]

echo \$?

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' showing a series of commands and their outputs. The commands are: `echo $str1`, `echo $str2`, `[ "$str1" = "$str2" ]`, `echo $?`, `[ "$str1" != "$str2" ]`, and `echo $?`. The outputs are: `wh`, `1`, and `0` respectively. A red box highlights the commands from `[ "$str1" = "$str2" ]` to the final `echo $?`.

```
wuhan@wuhan-virtual-machine:~$ echo $str1
wuhan@wuhan-virtual-machine:~$ echo $str2
wh
wuhan@wuhan-virtual-machine:~$ [ "$str1" = "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
1
wuhan@wuhan-virtual-machine:~$ [ "$str1" != "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$
```

7、比较 str1 与 str2 的大小

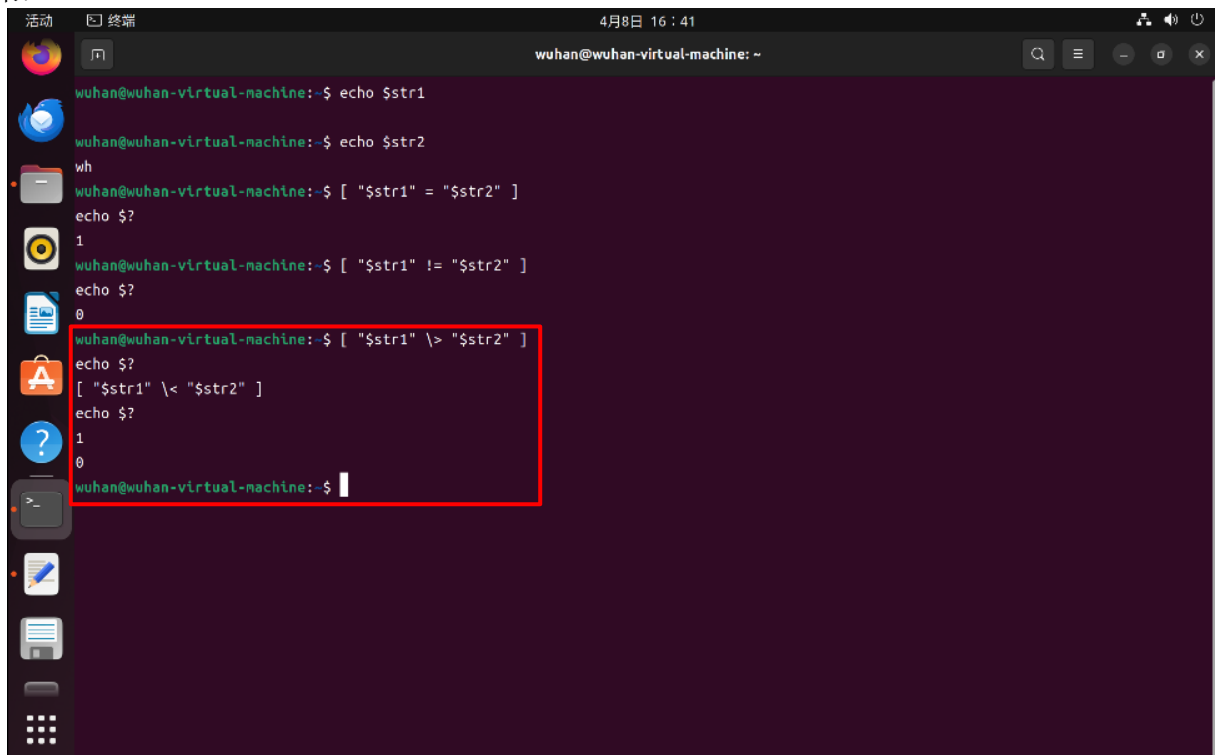
命令: [ "\$str1" \> "\$str2" ]

echo \$?

[ "\$str1" \< "\$str2" ]

echo \$?

结果:



A terminal window titled 'wuhan@wuhan-virtual-machine: ~' showing a series of commands and their outputs. The commands are: `echo $str1`, `echo $str2`, `[ "$str1" = "$str2" ]`, `echo $?`, `[ "$str1" != "$str2" ]`, `echo $?`, `[ "$str1" \> "$str2" ]`, `echo $?`, `[ "$str1" \< "$str2" ]`, `echo $?`, and `[ "$str1" \< "$str2" ]`. The outputs are: `wh`, `1`, `0`, `1`, `0`, `1`, and `0` respectively. A red box highlights the commands from `[ "$str1" \> "$str2" ]` to the final `[ "$str1" \< "$str2" ]`.

```
wuhan@wuhan-virtual-machine:~$ echo $str1
wuhan@wuhan-virtual-machine:~$ echo $str2
wh
wuhan@wuhan-virtual-machine:~$ [ "$str1" = "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
1
wuhan@wuhan-virtual-machine:~$ [ "$str1" != "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$ [ "$str1" \> "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
1
wuhan@wuhan-virtual-machine:~$ [ "$str1" \< "$str2" ]
wuhan@wuhan-virtual-machine:~$ echo $?
0
wuhan@wuhan-virtual-machine:~$ [ "$str1" \< "$str2" ]
wuhan@wuhan-virtual-machine:~$
```

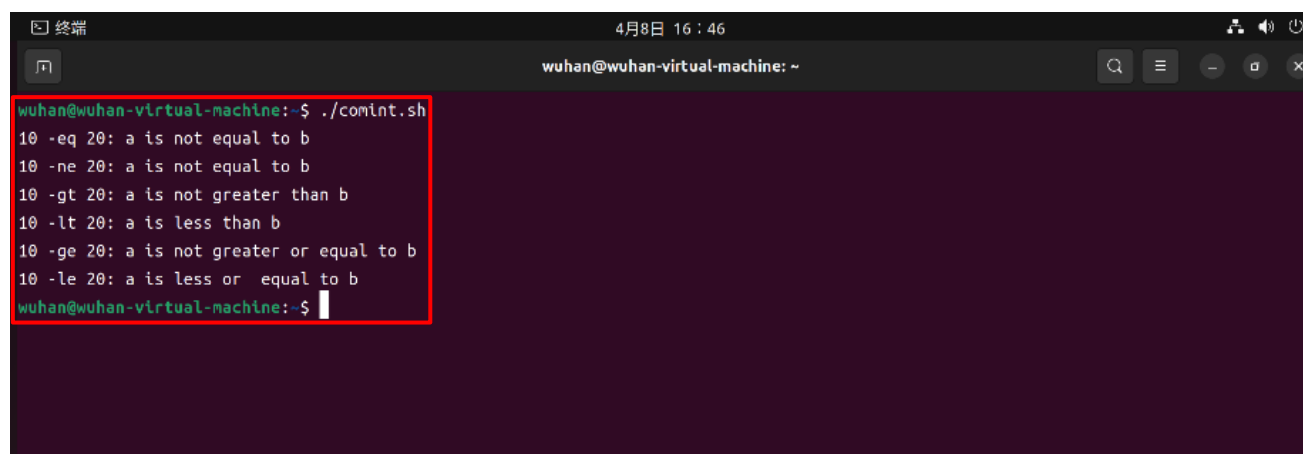
## 8、建立文件 comint.sh，进行整数比较

脚本：

```
#!/bin/sh
a=10
b=20
if [ $a -eq $b ]
then
    echo "$a -eq $b : a is equal to b"
else
    echo "$a -eq $b: a is not equal to b"
fi
if [ $a -ne $b ]
then
    echo "$a -ne $b: a is not equal to b"
else
    echo "$a -ne $b : a is equal to b"
fi
if [ $a -gt $b ]
then
    echo "$a -gt $b: a is greater than b"
else
    echo "$a -gt $b: a is not greater than b"
fi
if [ $a -lt $b ]
then
    echo "$a -lt $b: a is less than b"
else
    echo "$a -lt $b: a is not less than b"
fi
if [ $a -ge $b ]
then
    echo "$a -ge $b: a is greater or equal to b"
else
    echo "$a -ge $b: a is not greater or equal to b"
fi
if [ $a -le $b ]
then
    echo "$a -le $b: a is less or equal to b"
else
    echo "$a -le $b: a is not less or equal to b"
fi
```

命令： `./comint.sh`

结果：

A terminal window titled '终端' (Terminal) with a dark background. The prompt is 'wuhan@wuhan-virtual-machine: ~'. The command './comint.sh' has been executed, and the output is displayed. The output lines are: '10 -eq 20: a is not equal to b', '10 -ne 20: a is not equal to b', '10 -gt 20: a is not greater than b', '10 -lt 20: a is less than b', '10 -ge 20: a is not greater or equal to b', and '10 -le 20: a is less or equal to b'. The prompt 'wuhan@wuhan-virtual-machine:~\$' is visible at the bottom. A red rectangular box highlights the output lines.

```
终端 4月8日 16:46
wuhan@wuhan-virtual-machine: ~
wuhan@wuhan-virtual-machine:~$ ./comint.sh
10 -eq 20: a is not equal to b
10 -ne 20: a is not equal to b
10 -gt 20: a is not greater than b
10 -lt 20: a is less than b
10 -ge 20: a is not greater or equal to b
10 -le 20: a is less or equal to b
wuhan@wuhan-virtual-machine:~$
```

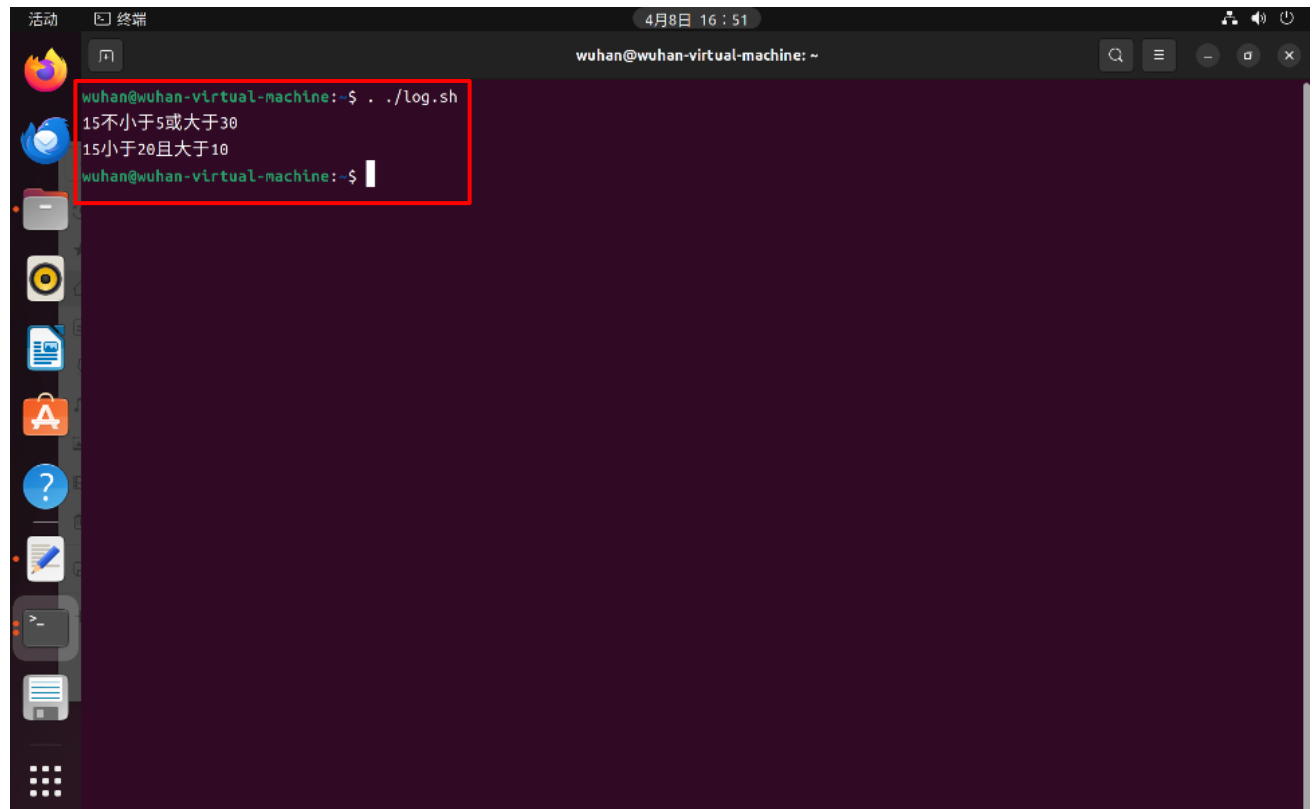
## 9、建立文件 log.sh，进行逻辑运算

脚本：

```
#!/bin/sh
iv=15
if [ ! "$iv" -lt 5 -o "$iv" -gt 30 ];then
    echo "$iv 不小于 5 或大于 30"
fi
if [ "$iv" -lt 20 -a "$iv" -gt 10 ];then
    echo "$iv 小于 20 且大于 10"
fi
```

命令：. ./log.sh

结果：

A screenshot of a Linux terminal window titled 'wuhan@wuhan-virtual-machine: ~'. The terminal shows the execution of the script './log.sh'. The output consists of two lines: '15不小于5或大于30' and '15小于20且大于10'. The first line is highlighted with a red box. The terminal also shows the prompt 'wuhan@wuhan-virtual-machine: \$' at the end of the first line and 'wuhan@wuhan-virtual-machine: \$' at the end of the second line. The terminal window has a dark background and a light-colored text. The top of the window shows the date and time '4月8日 16:51' and the username 'wuhan@wuhan-virtual-machine: ~'. The left side of the window shows a sidebar with various application icons.

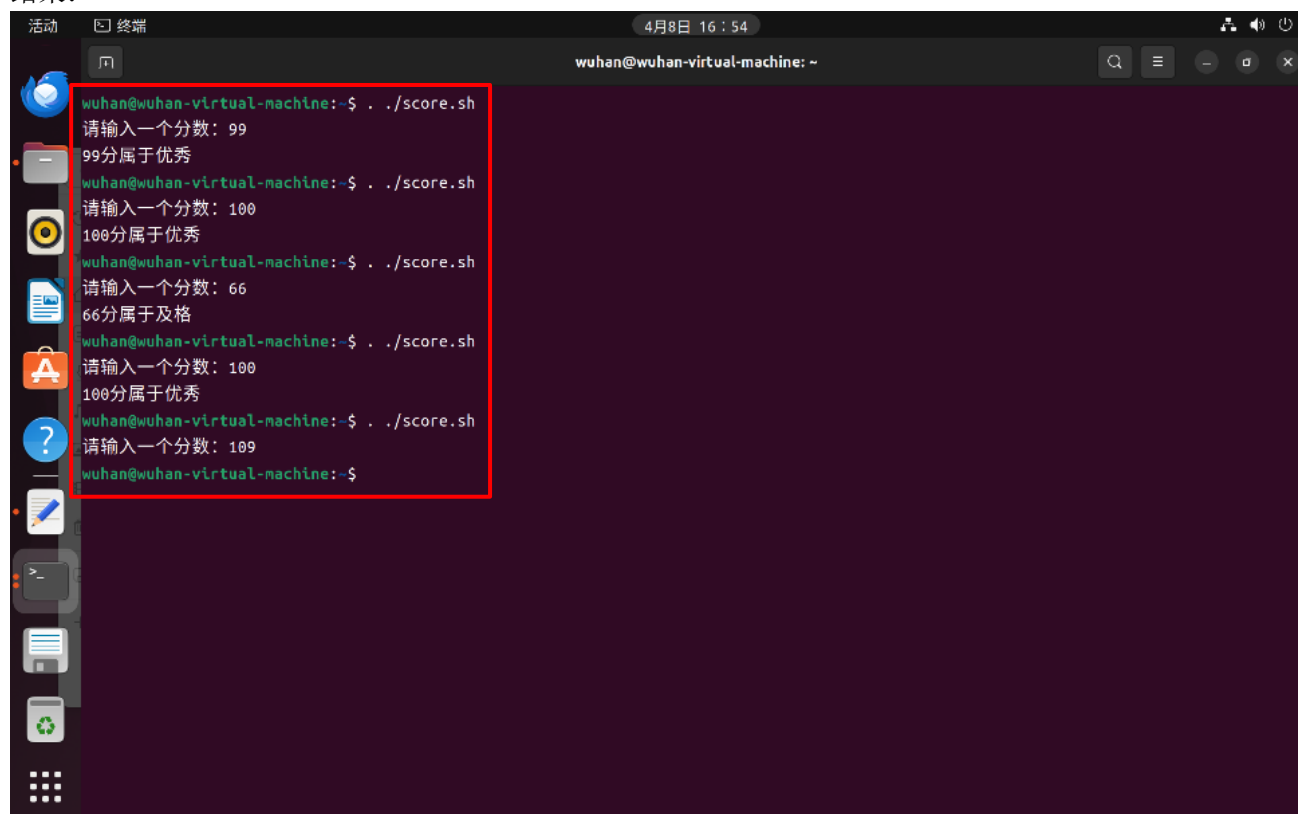
10、建立文件 score.sh，使用 if 进行成绩等级的判断

脚本：

```
#!/bin/bash
echo -n "请输入一个分数: "
read score
if [ "$score" -lt 60 ];then
    echo "$score 分属于不及格"
fi
if [ "$score" -lt 70 -a "$score" -ge 60 ];then
    echo "$score 分属于及格"
fi
if [ "$score" -lt 80 -a "$score" -ge 70 ];then
    echo "$score 分属于中等"
fi
if [ "$score" -lt 90 -a "$score" -ge 80 ];then
    echo "$score 分属于良好"
fi
if [ "$score" -le 100 -a "$score" -ge 90 ];then
    echo "$score 分属于优秀"
fi
```

命令：../score.sh

结果：



The screenshot shows a terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a dark purple background. A red box highlights the first five runs of the script. The terminal shows the prompt 'wuhan@wuhan-virtual-machine:~\$' followed by './score.sh'. The script prompts '请输入一个分数: ' (Please enter a score: ). The inputs and outputs are as follows:

Input Score	Output Grade
99	99分属于优秀
100	100分属于优秀
66	66分属于及格
100	100分属于优秀
109	109分属于优秀

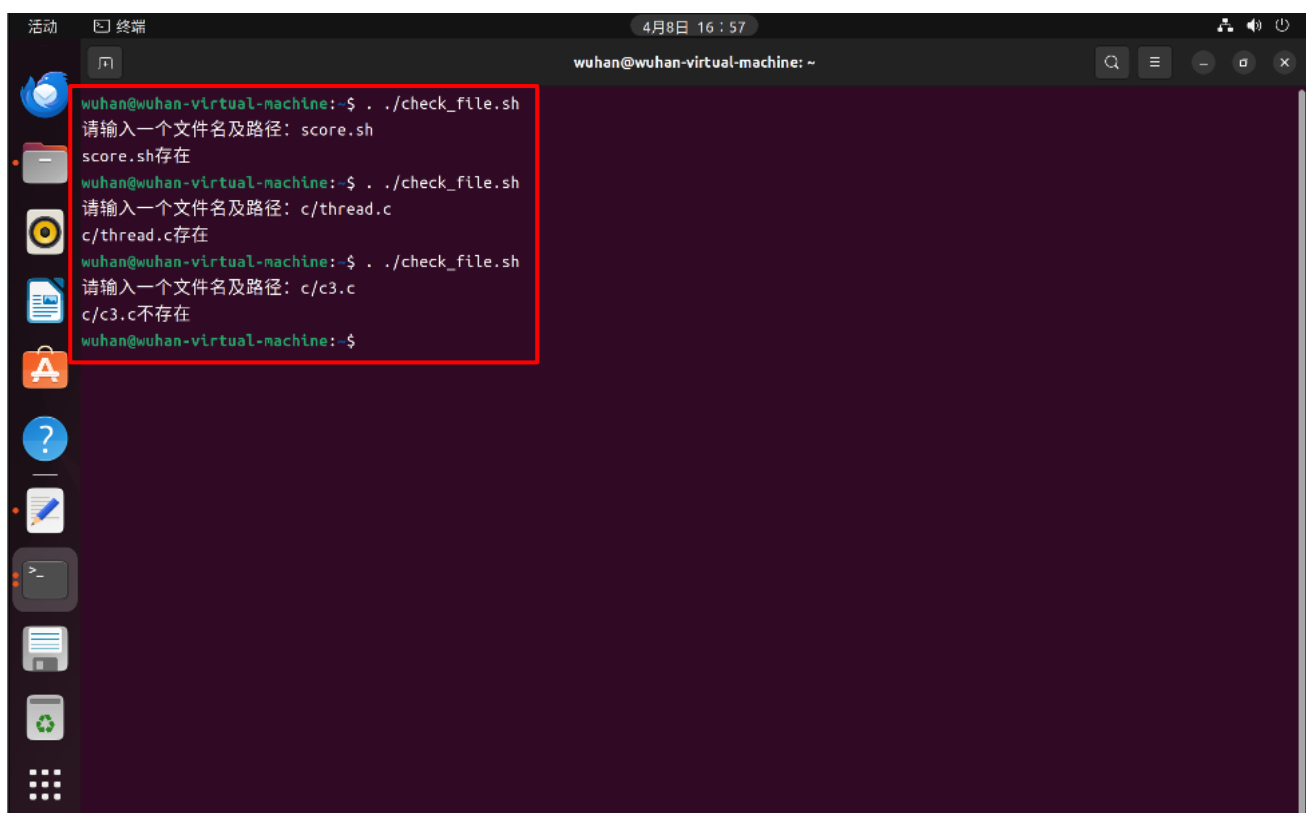
After the fifth run, the prompt returns to 'wuhan@wuhan-virtual-machine:~\$'.

11、建立文件 check\_file.sh，判断某个文件是否存在  
脚本：

```
#!/bin/bash
echo -n "请输入一个文件名及路径： "
read FILE
if [ -e $FILE ];then
    echo "$FILE 存在"
else
    echo "$FILE 不存在"
fi
```

命令： ../check\_file.sh

结果：



The screenshot shows a terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a dark purple background. The terminal displays the execution of the script ../check\_file.sh three times. The first run checks for 'score.sh' and reports it exists. The second run checks for 'c/thread.c' and reports it exists. The third run checks for 'c/c3.c' and reports it does not exist. The input prompts are in Chinese. A red box highlights the first three lines of the terminal output.

```
wuhan@wuhan-virtual-machine:~$ ./check_file.sh
请输入一个文件名及路径： score.sh
score.sh存在
wuhan@wuhan-virtual-machine:~$ ./check_file.sh
请输入一个文件名及路径： c/thread.c
c/thread.c存在
wuhan@wuhan-virtual-machine:~$ ./check_file.sh
请输入一个文件名及路径： c/c3.c
c/c3.c不存在
wuhan@wuhan-virtual-machine:~$
```

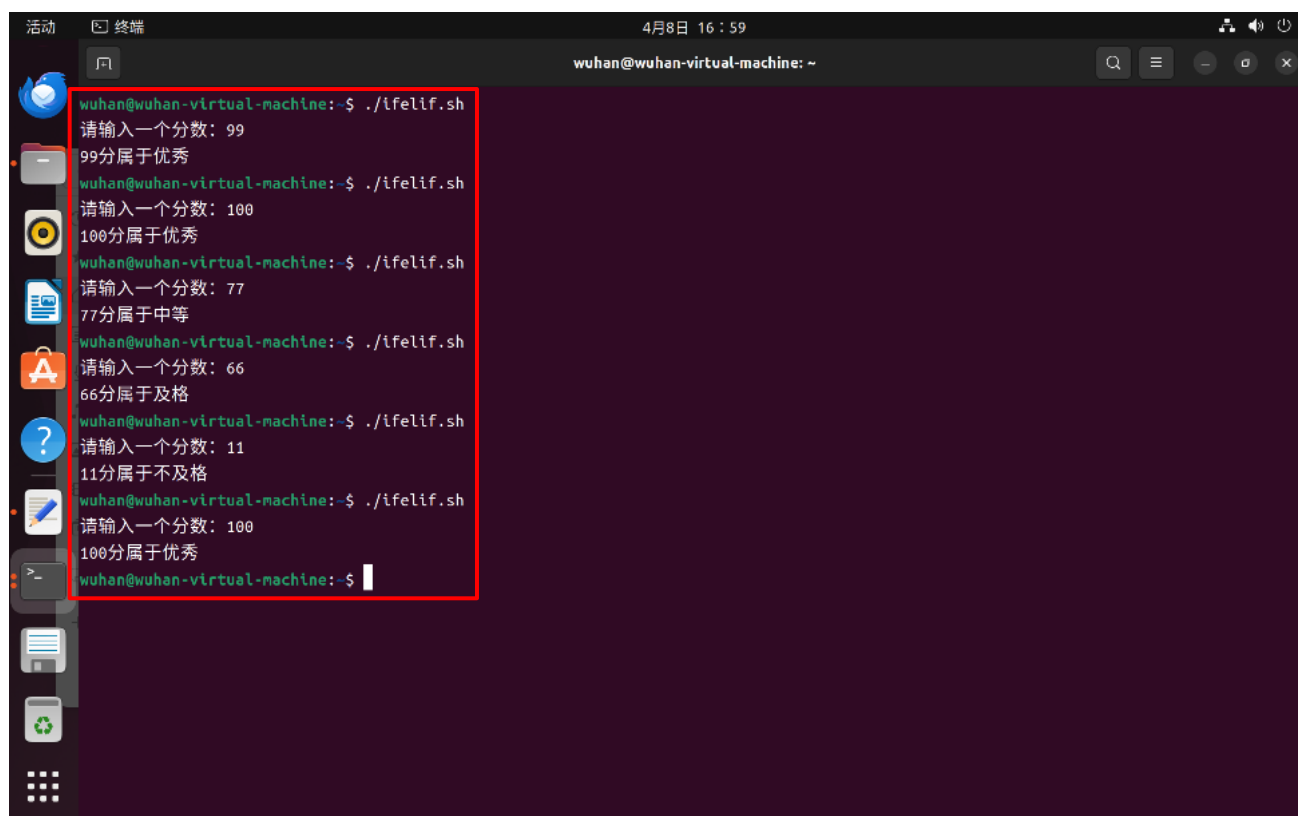
## 12、建立文件 ifelif.sh，对成绩等级进行判断

脚本：

```
#!/bin/bash
echo -n "请输入一个分数: "
read score
if [ "$score" -lt 60 ];then
    echo "$score 分属于不及格"
elif [ "$score" -lt 70 -a "$score" -ge 60 ];then
    echo "$score 分属于及格"
elif [ "$score" -lt 80 -a "$score" -ge 70 ];then
    echo "$score 分属于中等"
elif [ "$score" -lt 90 -a "$score" -ge 80 ];then
    echo "$score 分属于良好"
else
    echo "$score 分属于优秀"
fi
```

命令：./ifelif.sh

结果：



The screenshot shows a terminal window titled 'wuhan@wuhan-virtual-machine: ~' with a timestamp of '4月8日 16:59'. The terminal displays the execution of the script './ifelif.sh' multiple times with different input scores. The results are as follows:

Input Score	Output Grade
99	99分属于优秀
100	100分属于优秀
77	77分属于中等
66	66分属于及格
11	11分属于不及格
100	100分属于优秀