南昌大学实验报告

姓名：杨显申

学号：6130116092

邮箱地址：krito5898@gmail.com

专业班级：综合实验班161班

实验日期：2018年4月9日

课程名称：Linux程序设计实验

实验项目名称

Bash Programming

实验目的

> To write shell scripts to solve problems To implement some standard Linux utilities such as ls,cp,etc using systemcalls.

实验步骤

1.编辑文件内容

将脚本写入文件

Program1

> Write a Shell script that accepts a filename, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

```
$ vi program1
```

```
`#! /bin/sh
cat $1 | head -n $3 | tail -n +$2
```

Program2

> Write a Shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

```
$ vi program2
```

```
#! /bin/sh
until [ $# -eq 0 ]
do
    sed -i '/abc/d' $1
    shift
done
```

Program3

> Write a Shell script that displays list of all the files in the current directory to which the user has read, Write and execute permissions.

```
$ vi program3
```

```
#! /bin/sh
ls -l | sed -n "/^.rwx/p"
```

Program4

> Write a Shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.

```
$ vi program4
```

```
#! /bin/sh
until [ $# -eq 0 ]
do
    if [ -d $1 ]
    then
        echo "$1 is a directory"
    elif [ -f $1 ]
    then
        echo "$1 is a file"
        wc -l $1
    else
        echo "$1 is neither"
    fi
    shift
done
```

Program5

> Write a Shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

```
$ vi program5
```

```
#! /bin/bash
Dict=$1
DictCount=$(cat $Dict | wc -w)
declare -a WordCount
i=0
until [ $i -eq $DictCount ]
do
    WordCount[$i]=0
    i=$((i+1))
done
i=2
until [ $# -eq 1 ]
do
    j=0
    while read line
    do
        for word in $line
        do
            temp=`grep -o $word $2 | wc -w`
            echo  "$2 have $temp $word"
            WordCount[$j]=$((${WordCount[$j]}+$temp))
            j=$((j+1))
        done
    done < $Dict
    echo "-------------------------"
    shift
done
i=0
until [ $i -eq $DictCount ]
do
    read line
    for word in $line
    do
        echo "$word ${WordCount[$i]}"
        i=$((i+1))
    done
done < $Dict
echo "-------------------------"
```

Program6

> Write a Shell script to list all of the directory files in a directory

---

```
$ vi program6
```

```
#! /bin/sh
ls -l | grep "^d"
```

Program7

> Write a Shell script to find factorial of a given integer.

```
$ vi program7
```

```
#! /bin/sh
Product=1
i=$1
until [ $i -eq 1 ]
do
    Product=$(($Product*$i))
    i=$((i-1))
done
echo $Product
```

2.增加文件执行权限

为所有program文件增加执行权限

```
`$ chmod +x program*`
```

3.执行脚本文件并提供参数

Program1

```
$ ./program1 poem 3 5
```

Program2

```
$ ./program2 testfile1 testfile2
```

Program3

```
$ ./program3
```

Program4

```
$ ./program4
```

Program5

```
$ ./program5 dictionary program1 program2 program3 program4 program5
```

Program6

```
$ ./program6
```

Program7

```
$ ./program7 10
```

实验数据或结果

Program1

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program1 poem 3 5
will return a result,
which is not perturbed.
int global_int;int is_not_reentrant(int x) { int x = x; return global_int + x; },
krito@iZwz9j61g48vn45w3fb8p9Z:~$
```

Program2

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ cat testfile1
skfjdlwjif owije kls
wjefio abc jwiofkl
dsjfklw ojklvw
krito@iZwz9j61g48vn45w3fb8p9Z:~$ cat testfile2
jweiof jwefis wjekl
wjiov abc jiwof
jwfio b eiowjkl
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program2 testfile1 testfile2
krito@iZwz9j61g48vn45w3fb8p9Z:~$ cat testfile1
skfjdlwjif owije kls
dsjfklw ojklvw
krito@iZwz9j61g48vn45w3fb8p9Z:~$ cat testfile2
jweiof jwefis wjekl
jwfio b eiowjkl
```

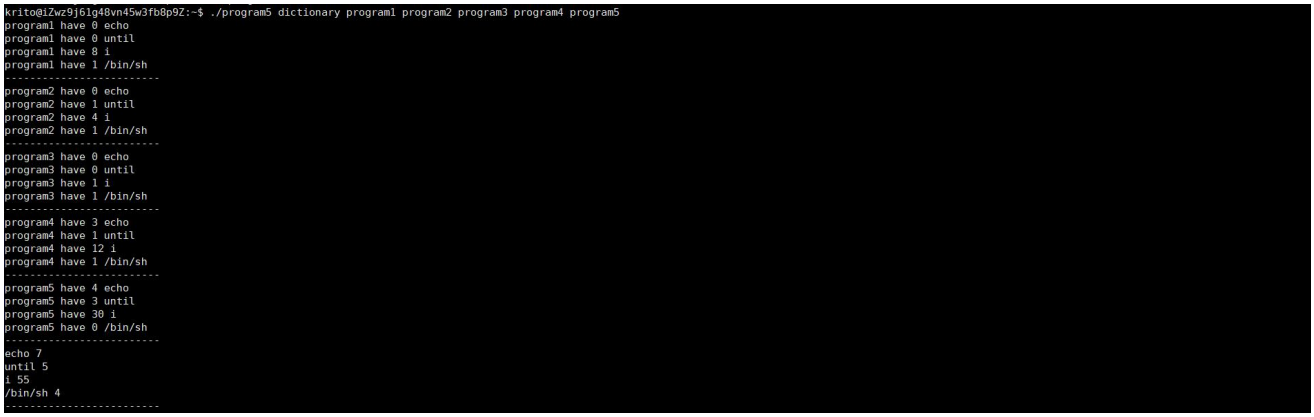
Program3

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ls -l
total 44
drwxrwxr-x 2 krito krito 4096 Apr  9 10:22 graph
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab2
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab3
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab4
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab5
-rw-rw-r-- 1 krito krito  389 Apr  9 10:27 poem
-rwxrwxr-x 1 krito krito  100 Apr  9 11:13 program1
-rwxrwxr-x 1 krito krito   63 Apr 15 17:38 program2
-rwxrwxr-x 1 krito krito   37 Apr 15 19:55 program3
-rw-rw-r-- 1 krito krito   37 Apr 15 17:58 testfile1
-rw-rw-r-- 1 krito krito   36 Apr 15 17:58 testfile2
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program3
drwxrwxr-x 2 krito krito 4096 Apr  9 10:22 graph
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab2
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab3
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab4
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab5
-rwxrwxr-x 1 krito krito  100 Apr  9 11:13 program1
-rwxrwxr-x 1 krito krito   63 Apr 15 17:38 program2
-rwxrwxr-x 1 krito krito   37 Apr 15 19:55 program3
```

Program4

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ls -l
total 48
drwxrwxr-x 2 krito krito 4096 Apr  9 10:22 graph
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab2
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab3
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab4
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab5
-rw-rw-r-- 1 krito krito  389 Apr  9 10:27 poem
-rwxrwxr-x 1 krito krito  100 Apr  9 11:13 program1
-rwxrwxr-x 1 krito krito   63 Apr 15 17:38 program2
-rwxrwxr-x 1 krito krito   37 Apr 15 19:55 program3
-rwxrwxr-x 1 krito krito  166 Apr 15 20:28 program4
-rw-rw-r-- 1 krito krito   37 Apr 15 17:58 testfile1
-rw-rw-r-- 1 krito krito   36 Apr 15 17:58 testfile2
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program4 graph lab2 lab4 program1 program3
graph is a directory
lab2 is a directory
lab4 is a directory
program1 is a file
5 program1
program3 is a file
2 program3
```

Program5



Program6

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ls -l
total 60
-rw-rw-r-- 1 krito krito   21 Apr 15 22:05 dictionary
drwxrwxr-x 2 krito krito 4096 Apr  9 10:22 graph
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab2
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab3
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab4
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab5
-rw-rw-r-- 1 krito krito  389 Apr  9 10:27 poem
-rwxrwxr-x 1 krito krito  100 Apr  9 11:13 program1
-rwxrwxr-x 1 krito krito   63 Apr 15 17:38 program2
-rwxrwxr-x 1 krito krito   37 Apr 15 21:01 program3
-rwxrwxr-x 1 krito krito  166 Apr 15 20:28 program4
-rwxrwxr-x 1 krito krito  577 Apr 15 22:08 program5
-rwxrwxr-x 1 krito krito   29 Apr 15 22:14 program6
-rw-rw-r-- 1 krito krito   37 Apr 15 17:58 testfile1
-rw-rw-r-- 1 krito krito   36 Apr 15 17:58 testfile2
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program6
drwxrwxr-x 2 krito krito 4096 Apr  9 10:22 graph
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab2
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab3
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab4
drwxrwxr-x 2 krito krito 4096 Apr 15 19:14 lab5
```

Program7

```
krito@iZwz9j61g48vn45w3fb8p9Z:~$ ./program7 10
3628800
```

实验思考

在实验五中，使用declare声明数组与使用let进行数值运算时均出现错误，将let语句改为$((expression))扩展语句后可解决，但对于声明数组却无从下手。翻看declare语法介绍的教程时发现他所使用的内核是bash而不是sh，更改内核后可正常运行。

参考资料

Linux Sed命令详解

linux如何显示一个文件的某几行（中间几行）

linux中sed的用法

linux中SHELL脚本中的数组用法

grep命令

Linux Shell 之 对文件中的行、单词、字符进行迭代

linux统计文件夹中文件数目

《Linux程序设计（第2版）/大学计算机应用技术系列教材》