

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Голчин Задех Негин¹

22 июня, 2023, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

Изучить основы программирования в оболочке ОС UNIX.
Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Задачи лабораторной работы

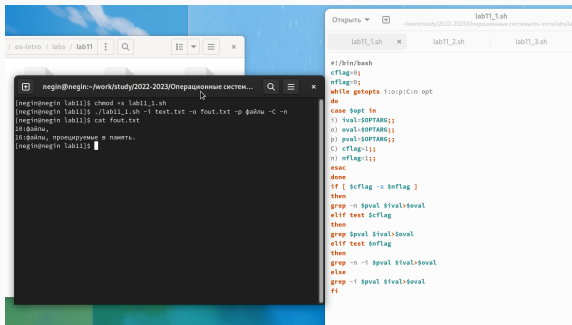
1 Выполнить 4 задания

Процесс выполнения лабораторной работы

1. Используя команды `getopts` `grep` напишем командный файл, который анализирует командную строку с ключами и выполним его: `-i inputfile` — прочитать данные из указанного файла; `-o outputfile` — вывести данные в указанный файл; `-r шаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк;

а затем ищет в указанном файле нужные строки

Выполнение работы



The image shows a terminal window and a code editor. The terminal window, titled 'negin@negin:~/work/study/2022-2023/Операционные систем...', displays the following commands and output:

```
[negin@negin lab1]$ chmod +x lab11_1.sh
[negin@negin lab1]$ ./lab11_1.sh -i text.txt -o fout.txt -p файл -C -n
[negin@negin lab1]$ cat fout.txt
10:файл,
10:файл, просируение в память.
[negin@negin lab1]$
```

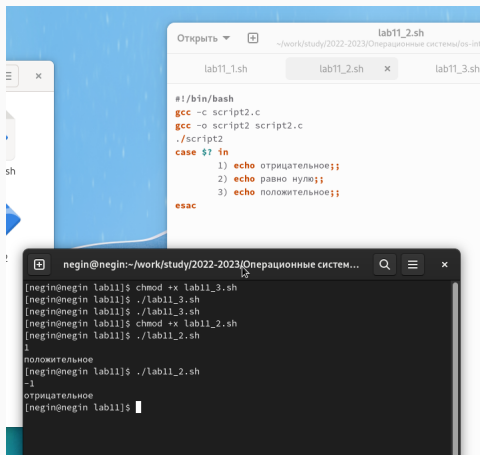
The code editor, titled 'lab11_1.sh', shows the following script:

```
#!/bin/bash
cflag=0;
nflag=0;
while getopts fio:p:Cm opt
do
case $opt in
i) ival=$OPTARG;;
o) oval=$OPTARG;;
p) pval=$OPTARG;;
C) cflag=1;;
n) nflag=1;;
esac
done
if [ $cflag -o $nflag ]
then
grep -n $pval $ival>$oval
elif test $cflag
then
grep $pval $ival>$oval
elif test $nflag
then
grep -n -i $pval $ival>$oval
else
grep -i $pval $ival>$oval
fi
```

Рис. 1: Задание 1

2. Напишем сначала на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем завершим программу при помощи функции `exit(n)`, передавая информацию о коде завершения в оболочку. Командный файл вызовет эту программу и, проанализировав с помощью команды `$?`, выдаст сообщение о том, какое число было введено

Выполнение работы



The image shows a Linux desktop environment with two terminal windows. The top window, titled 'lab11_2.sh', is an editor showing a shell script. The script uses 'gcc' to compile 'script2.c' and then runs './script2'. It features a 'case' statement with three branches: 'отрицательное;;', 'равно нулю;;', and 'положительное;;', each followed by an 'echo' command. The bottom window, titled 'negin@negin:~/work/study/2022-2023/Операционные систем...', shows the execution of these scripts. It includes 'chmod +x' commands for 'lab11_3.sh' and 'lab11_2.sh', followed by './lab11_3.sh' and './lab11_2.sh'. The output of the second script shows 'положительное' and '-1' on separate lines.

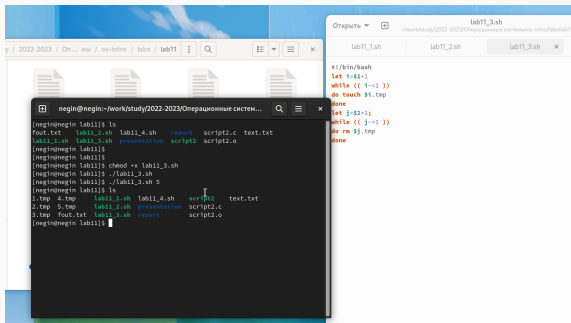
```
#!/bin/bash
gcc -c script2.c
gcc -o script2 script2.c
./script2
case $? in
    1) echo отрицательное;;
    2) echo равно нулю;;
    3) echo положительное;;
esac
```

```
[negin@negin lab11]$ chmod +x lab11_3.sh
[negin@negin lab11]$ ./lab11_3.sh
[negin@negin lab11]$ ./lab11_3.sh
[negin@negin lab11]$ chmod +x lab11_2.sh
[negin@negin lab11]$ ./lab11_2.sh
1
положительное
[negin@negin lab11]$ ./lab11_2.sh
-1
отрицательное
[negin@negin lab11]$
```

Рис. 2: Задание 2

3. Напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N

Выполнение работы



The screenshot displays a terminal window in the foreground and a script editor window in the background. The terminal window shows the execution of a shell script named `lab11_3.sh`. The script performs several actions: it lists files, runs `chmod` on `lab11_3.sh`, executes `./lab11_3.sh`, and then runs a loop that creates temporary files, runs `script2.c`, and removes them. The background window shows the content of `lab11_3.sh`, which includes a `while` loop that runs `touch $i.tmp` and a `while` loop that runs `rm $j.tmp`.

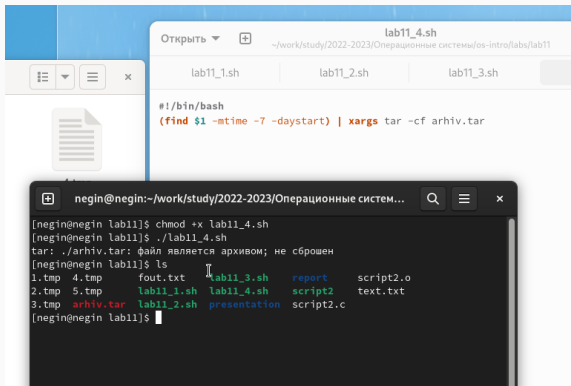
```
negin@negin:~/work/study/2022-2023/Операционные систем...  
[negin@negin lab11]$ ls  
fout.txt  lab11_3.sh  lab11_4.sh  report  script2.c  text.txt  
lab11_1.sh  lab11_2.sh  presentation  script2  script2.o  
[negin@negin lab11]$  
[negin@negin lab11]$  
[negin@negin lab11]$ chmod +x lab11_3.sh  
[negin@negin lab11]$ ./lab11_3.sh  
[negin@negin lab11]$ ./lab11_3.sh 5  
[negin@negin lab11]$ ls  
1.tmp  4.tmp  lab11_1.sh  lab11_4.sh  script2  text.txt  
2.tmp  5.tmp  lab11_2.sh  presentation  script2.c  
3.tmp  fout.txt  lab11_3.sh  report  script2.o  
[negin@negin lab11]$
```

```
#!/bin/bash  
let i=$1+1  
while (( i-<=1 ))  
do touch $i.tmp  
done  
let j=$2+1  
while (( j->=1 ))  
do rm $j.tmp  
done
```

Рис. 3: Задание 3

4. Напишем командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицируем его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад.

Выполнение работы



The screenshot shows a terminal window with the following content:

```
negin@negin:~/work/study/2022-2023/Операционные систем...  
[negin@negin lab11]$ chmod +x lab11_4.sh  
[negin@negin lab11]$ ./lab11_4.sh  
tar: ./arhiv.tar: файл является архивом; не сброшен  
[negin@negin lab11]$ ls  
1.tmp 4.tmp fout.txt lab11_3.sh report script2.o  
2.tmp 5.tmp lab11_1.sh lab11_4.sh script2 text.txt  
3.tmp arhiv.tar lab11_2.sh presentation script2.c  
[negin@negin lab11]$
```

The terminal window has a title bar with a search icon, a menu icon, and a close icon. The file explorer window in the background shows a list of files: lab11_1.sh, lab11_2.sh, and lab11_3.sh.

Рис. 4: Задание 4

Выводы по проделанной работе

В данной работе мы изучили основы программирования в оболочке ОС UNIX и писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.