

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Алексей Понамарев¹

18 апреля, 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

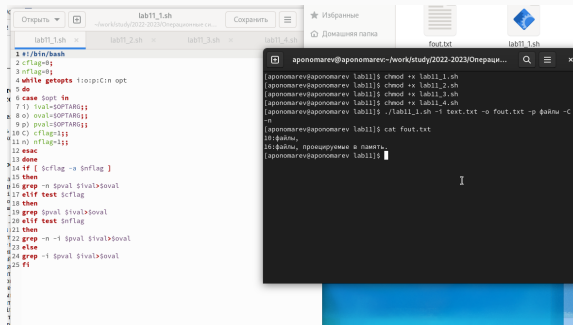
1 Выполнить 4 задания

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 screenshot shows a terminal window with a shell script and its execution. The script is a shell script that takes a file path as an argument and processes it. The script is as follows:

```
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:oi:pC:n opt
5 do
6 case $opt in
7 i) sval=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $sval>$oval
17 elif test $cflag
18 then
19 grep $pval $sval>$oval
20 elif test $nflag
21 then
22 grep -n -i $pval $sval>$oval
23 else
24 grep -i $pval $sval>$oval
25 fi
26
```

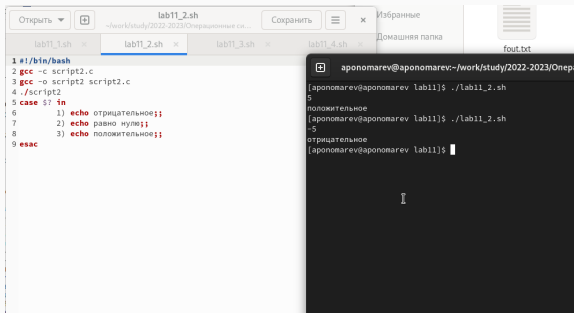
The script is executed in a terminal window with the following commands and output:

```
аромомареv@аромомареv lab11_1.sh
аромомареv@аромомареv lab11_1.sh$ chmod +x lab11_1.sh
аромомареv@аромомареv lab11_1.sh$ chmod +x lab11_2.sh
аромомареv@аромомареv lab11_1.sh$ chmod +x lab11_3.sh
аромомареv@аромомареv lab11_1.sh$ chmod +x lab11_4.sh
аромомареv@аромомареv lab11_1.sh$ ./lab11_1.sh -i text.txt -o fout.txt -p файлн -C
-n
аромомареv@аромомареv lab11_1.sh$ cat fout.txt
16:файлн, прецедующие в память.
аромомареv@аромомареv lab11_1.sh$
```

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 code editor window with several tabs. The active tab is 'lab11_2.sh', which contains the following code:

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

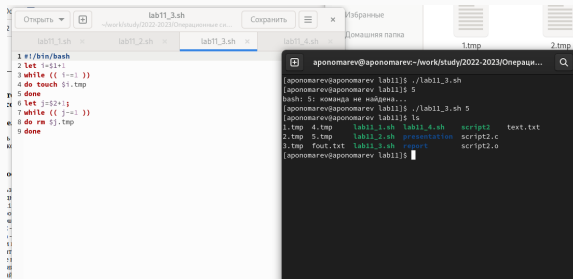
Below the code editor, a terminal window is open, showing the execution of the script. The prompt is 'aponomarev@aponomarev:~/work/study/2022-2023/Onepa'. The command './lab11_2.sh' has been executed, and the output is:

```
5
положительное
[aponomarev@aponomarev lab11]$ ./lab11_2.sh
-5
отрицательное
[aponomarev@aponomarev lab11]$
```

Рис. 2: Задание 2

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

Выполнение работы



The image shows a terminal window with a light gray title bar. The title bar contains the text "lab11_3.sh" and a search icon. Below the title bar, there are four tabs labeled "lab11_1.sh", "lab11_2.sh", "lab11_3.sh", and "lab11_4.sh". The "lab11_3.sh" tab is active. The terminal content shows a script being executed, which creates a file named "5". The script is as follows:

```
1 #!/bin/bash
2 let i=1
3 while (( i-- ))
4 do touch $i.tmp
5 done
6 let j=2
7 while (( j-- ))
8 do rm $j.tmp
9 done
```

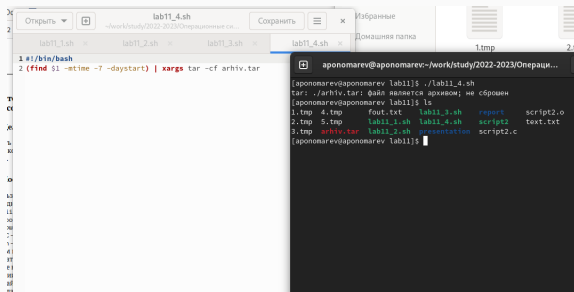
Below the script, there is a file listing command and its output:

```
[aponomarev@aponomarev lab11]$ ls
1.tmp 4.tmp lab11_1.sh lab11_4.sh script2 test.txt
2.tmp 5.tmp lab11_2.sh presentation script2.c
3.tmp fout.txt lab11_3.sh report script2.o
```

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 a file manager interface at the top. The terminal has two tabs: 'lab11_1.sh' and 'lab11_4.sh'. The 'lab11_4.sh' tab is active, showing the following commands and output:

```
1 #!/bin/bash
2 (find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar
```

The output of the script is displayed below the commands:

```
[aponomarev@aponomarev lab11]$ ./lab11_4.sh
tar: ./arhiv.tar: файл является архивом; не сброшен
[aponomarev@aponomarev 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  arhive.tar  lab11_2.sh  presentation  script2.c
```

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.