

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Григорьев Давид Андроникович НПМБД-01-21¹

27 мая, 2022, 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

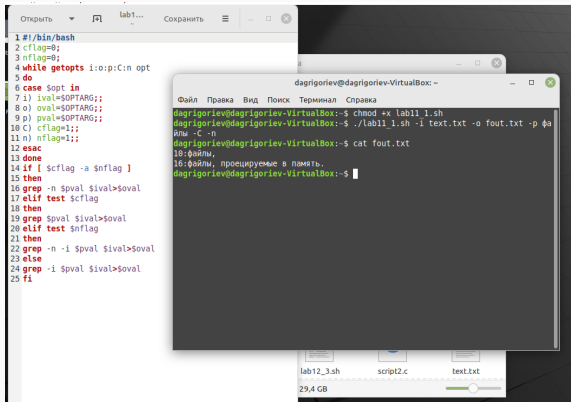
1 Выполнить 4 задания

Процесс выполнения лабораторной работы

1. Используя команды `getopts` `grep` напомним 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 Linux terminal window with a menu bar (Файл, Правка, Вид, Поиск, Терминал, Справка) and a title bar (dagrigoriev@dagrigoriev-VirtualBox: ~). The terminal displays a shell script and its execution output.

```
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:o:p:C:n opt
5 do
6 case $opt in
7 i) ival=$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 $ival>$oval
17 elif test $cflag
18 then
19 grep $pval $ival>$oval
20 elif test $nflag
21 then
22 grep -n -i $pval $ival>$oval
23 else
24 grep -i $pval $ival>$oval
25 fi
```

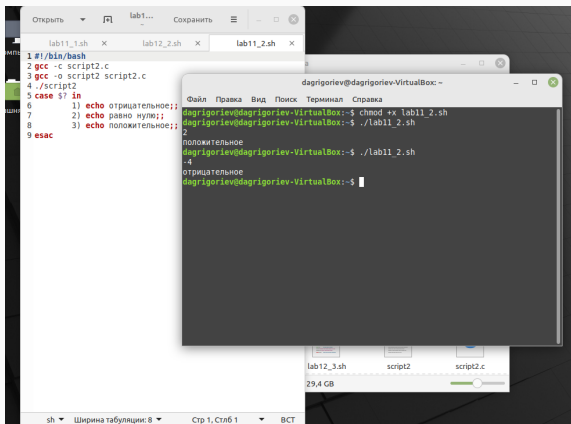
The terminal output shows the execution of the script with the following commands and results:

```
dagrigoriev@dagrigoriev-VirtualBox:~$ chmod +x lab11_1.sh
dagrigoriev@dagrigoriev-VirtualBox:~$ ./lab11_1.sh -i text.txt -o fout.txt -p ф
фл -C -n
dagrigoriev@dagrigoriev-VirtualBox:~$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
dagrigoriev@dagrigoriev-VirtualBox:~$
```

Figure 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



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

```
dagrigoriev@dagrigoriev-VirtualBox: ~$ chmod +x lab11_2.sh
dagrigoriev@dagrigoriev-VirtualBox: ~$ ./lab11_2.sh
положительное
dagrigoriev@dagrigoriev-VirtualBox: ~$ ./lab11_2.sh
-4
отрицательное
dagrigoriev@dagrigoriev-VirtualBox: ~$
```

Figure 2: Задание 2

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

Выполнение работы

The screenshot shows a terminal window with a script being executed. The script is as follows:

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

The terminal output shows the execution of the script and a subsequent file listing:

```
dagrigoriev@dagrigoriev-VirtualBox:~$ chmod +x lab11_3.sh
dagrigoriev@dagrigoriev-VirtualBox:~$ ./lab11_3.sh 10
dagrigoriev@dagrigoriev-VirtualBox:~$ ls
10.tmp  7.tmp      hello.sh  lab11_3.sh  script2.o  Общедоступные
1.tmp   8.tmp      lab10_1.sh lab11_4.sh  text.txt   'Рабочий стол'
2.tmp   9.tmp      lab10_2.sh lab12_1.sh  Видео     Шаблоны
3.tmp   backup    lab10_3.sh lab12_2.sh  Документы
4.tmp   file.txt  lab10_4.sh lab12_3.sh  Загрузки
5.tmp   fout.txt  lab11_1.sh script2     Изображения
6.tmp   '#hello.sh#' lab11_2.sh script2.c  Музыка

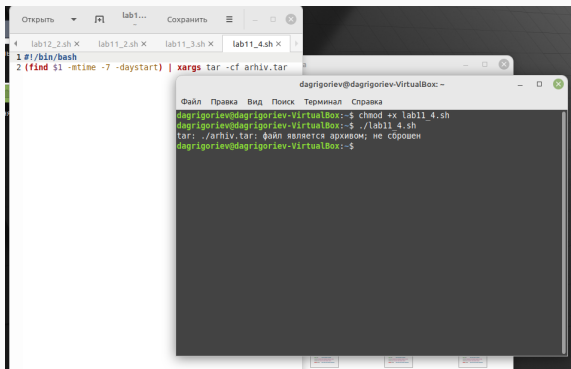
dagrigoriev@dagrigoriev-VirtualBox:~$
```

The terminal window also shows a file manager interface at the bottom with tabs for lab10_1.sh, lab10_2.sh, and lab10_3.sh, and a progress bar indicating 29.4 GB.

Figure 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 menu bar (Открыть, Сохранить) and several tabs labeled 'lab1...', 'lab12_2.sh X', 'lab11_2.sh X', 'lab11_3.sh X', and 'lab11_4.sh X'. The terminal content is as follows:

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

A second terminal window, titled 'dagrigoriev@dagrigoriev-VirtualBox: -', is overlaid on top. It shows the following commands and output:

```
dagrigoriev@dagrigoriev-VirtualBox:~$ chmod +x lab11_4.sh
dagrigoriev@dagrigoriev-VirtualBox:~$ ./lab11_4.sh
tar: ./arhiv.tar: файл является архивом; не сброшен
dagrigoriev@dagrigoriev-VirtualBox:~$
```

Figure 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.