

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Ванчинга Дэвид¹

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

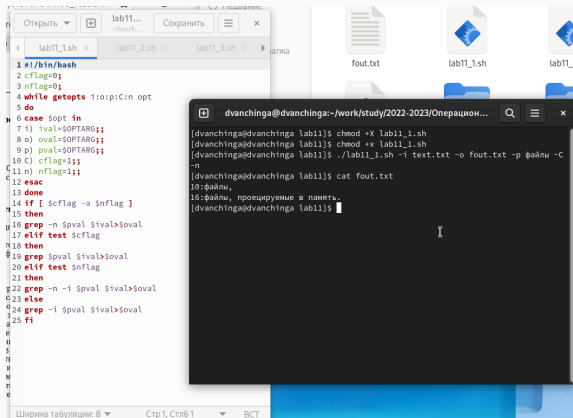
1 Выполнить 4 задания

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 image shows a Linux desktop environment. In the background, a text editor window displays a shell script named `lab11_1.sh`. The script uses `getopts` to parse command-line options `-i`, `-o`, and `-p`, and then uses `grep` to search for patterns in a file `$o`. In the foreground, a terminal window shows the execution of the script. The user sets permissions on `lab11_1.sh` and runs it with `./lab11_1.sh -i text.txt -o fout.txt -p файлы -C -n`. The output of the script is displayed in the terminal, showing the contents of `файлы` and `файлы, проецируемые в память`.

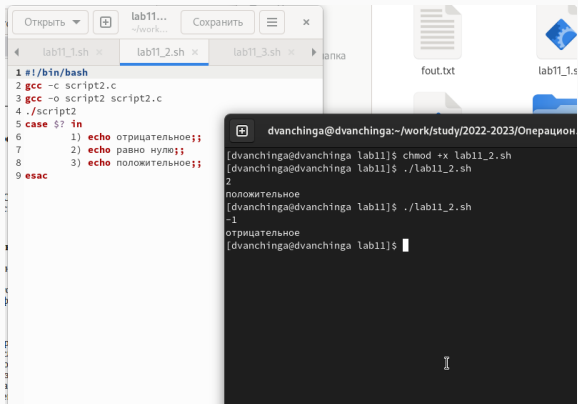
```
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:op:C:n opt
5 do
6 case $opt in
7 i) ival=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $ival>$oval
17 elif test $cflag
18 then
19 grep $pval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

```
[dvanchinga@dvanchinga lab11]$ chmod +X lab11_1.sh
[dvanchinga@dvanchinga lab11]$ chmod +x lab11_1.sh
[dvanchinga@dvanchinga lab11]$ ./lab11_1.sh -i text.txt -o fout.txt -p файлы -C
-n
[dvanchinga@dvanchinga lab11]$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
[dvanchinga@dvanchinga lab11]$
```

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 on the left and a terminal window on the right. The code editor displays a shell script named `lab11_2.sh` with the following content:

```
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
```

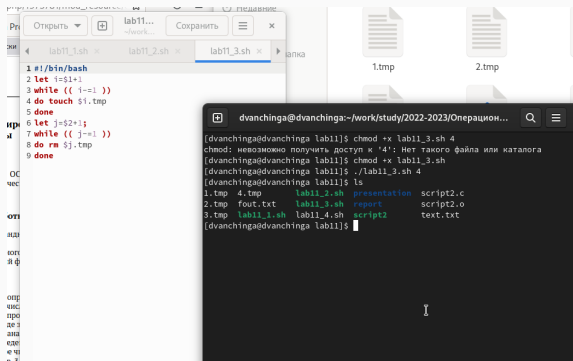
The terminal window shows the execution of the script. The user runs `chmod +x lab11_2.sh` and then `./lab11_2.sh`. The output of the script is:

```
2
положительное
[dvanchinga@dvanchinga lab11]$ ./lab11_2.sh
-1
отрицательное
[dvanchinga@dvanchinga 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 and a file manager window. The terminal window displays the execution of a shell script named `lab11_3.sh`. The script contains a `while` loop that increments a variable `i` and touches a file `1.tmp`. The file manager window shows the contents of the `lab11_3.sh` file, which includes the following code:

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

The terminal window shows the output of the script execution, including the `chmod` command and the `ls` command, which lists the files in the current directory:

```
[dvanchinga@dvanchinga lab11]$ chmod +x lab11_3.sh 4
[dvanchinga@dvanchinga lab11]$ chmod +x lab11_3.sh
[dvanchinga@dvanchinga lab11]$ ./lab11_3.sh 4
[dvanchinga@dvanchinga lab11]$ ls
1.tmp 4.tmp lab11_2.sh presentation script2.c
2.tmp fout.txt lab11_3.sh report script2.o
3.tmp lab11_1.sh lab11_4.sh script2 text.txt
[dvanchinga@dvanchinga lab11]$
```

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

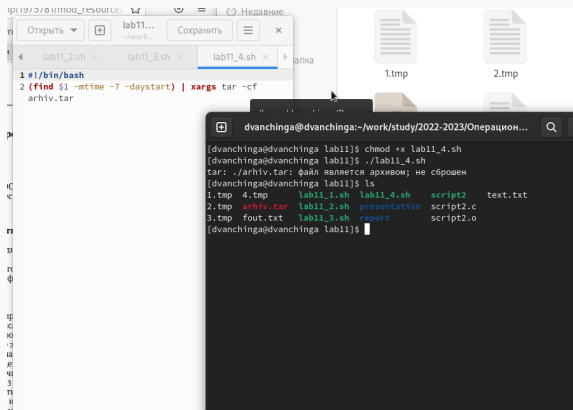


Рис. 4: Задание 4

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.