

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Старшинов Игорь НБИ-01-21¹

17 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

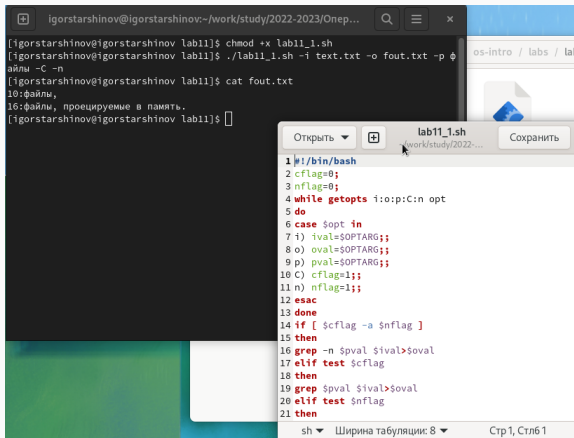
1 Выполнить 4 задания

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 terminal window and a file editor. The terminal window, titled 'igorstarshinov@igorstarshinov:~/work/study/2022-2023/Onep...', displays the following commands and output:

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

The file editor, titled 'lab11_1.sh', shows the content of the script 'lab11_1.sh' with line numbers 1 through 21. The script is a shell script that processes command-line options and prints the results of a grep command.

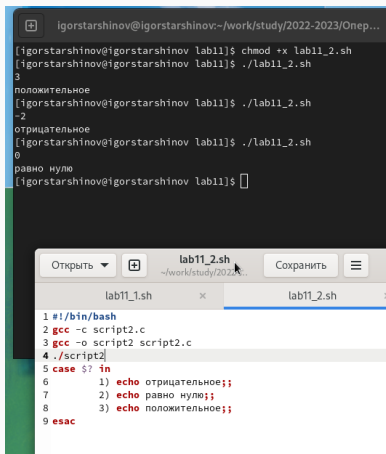
```
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:o:p:C:n opt
5 do
6 case $opt in
7 i) ival=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $ival>$oval
17 elif test $cflag
18 then
19 grep $pval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

The file editor also shows a menu bar with 'Открыть', '+', and 'Сохранить' buttons, and a status bar at the bottom indicating 'sh' and 'Ширина табуляции: 8'.

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 terminal window and a code editor. The terminal window displays the execution of a script named `lab11_2.sh`. The user runs `chmod +x lab11_2.sh` and then `./lab11_2.sh`. The script outputs the following text:

```
положительное
3
[igorstarshinov@igorstarshinov lab11]$ ./lab11_2.sh
-2
отрицательное
[igorstarshinov@igorstarshinov lab11]$ ./lab11_2.sh
0
равно нулю
[igorstarshinov@igorstarshinov lab11]$
```

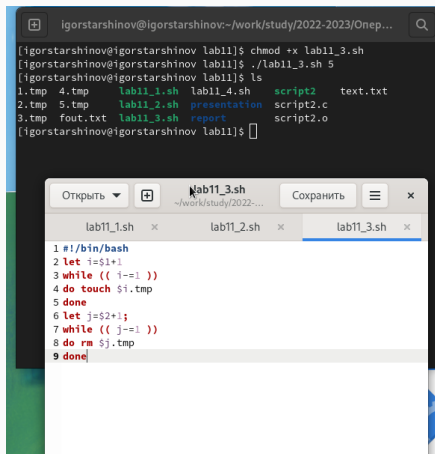
The code editor shows the content of `lab11_2.sh`:

```
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
```

Рис. 2: Задание 2

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

Выполнение работы



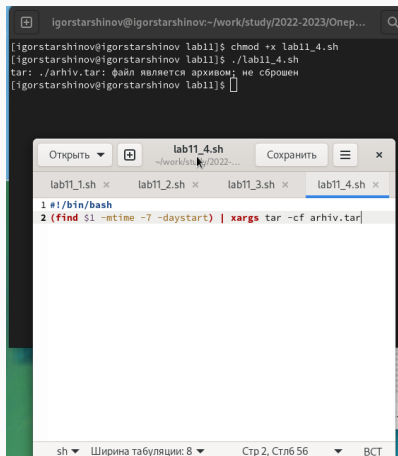
The image shows a terminal window and a code editor. The terminal window displays the execution of a shell script named `lab11_3.sh`. The user runs `chmod +x lab11_3.sh` and then `./lab11_3.sh 5`. The `ls` command shows the files in the directory: `1.tmp`, `4.tmp`, `lab11_1.sh`, `lab11_4.sh`, `script2`, and `text.txt`. The code editor shows the contents of `lab11_3.sh`, which is a shell script that uses a `while` loop to create and delete files.

```
igorstarshinov@igorstarshinov:~/work/study/2022-2023/Onep...  
[igorstarshinov@igorstarshinov lab11]$ chmod +x lab11_3.sh  
[igorstarshinov@igorstarshinov lab11]$ ./lab11_3.sh 5  
[igorstarshinov@igorstarshinov 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  
[igorstarshinov@igorstarshinov lab11]$  
  
lab11_3.sh  
~/work/study/2022-...  
lab11_1.sh x lab11_2.sh x lab11_3.sh x  
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
```

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 image shows a terminal window and a code editor. The terminal window at the top displays the following commands and output:

```
igorstarshinov@igorstarshinov:~/work/study/2022-2023/Onep...  
[igorstarshinov@igorstarshinov lab11]$ chmod +x lab11_4.sh  
[igorstarshinov@igorstarshinov lab11]$ ./lab11_4.sh  
tar: ./arhiv.tar: файл является архивом; не сброшен  
[igorstarshinov@igorstarshinov lab11]$
```

Below the terminal is a code editor window titled "lab11_4.sh". The editor shows the following code:

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

The code editor window has tabs for "lab11_1.sh", "lab11_2.sh", "lab11_3.sh", and "lab11_4.sh". The status bar at the bottom indicates "sh", "Ширина табуляции: 8", "Стр 2, Стлб 56", and "BCT".

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.