

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Саргсян Сурен Бабкенович¹

29 августа, 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

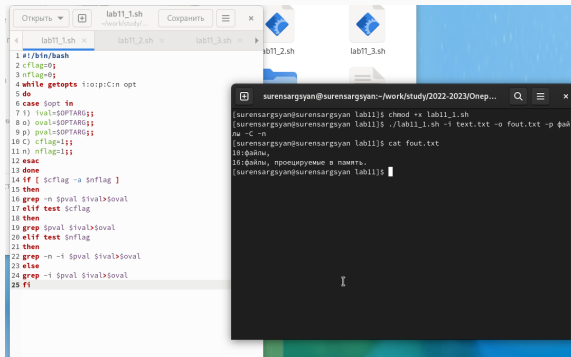
1 Выполнить 4 задания

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 code editor window with a file named `lab11_1.sh` and a terminal window showing the execution of the script.

Code Editor (lab11_1.sh):

```
1 #!/bin/bash
2 cflag=0
3 nflag=0
4 while getopts iio:p:C:m 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
```

Terminal Window:

```
surensargsyan@surensargsyan:~/work/study/2022-2023/Onep...
[surensargsyan@surensargsyan lab11]$ chmod +x lab11_1.sh
[surensargsyan@surensargsyan lab11]$ ./lab11_1.sh -i text.txt -o fout.txt -p фай
nw -C -n
[surensargsyan@surensargsyan lab11]$ cat fout.txt
16:файну, проецируемые в память.
[surensargsyan@surensargsyan 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

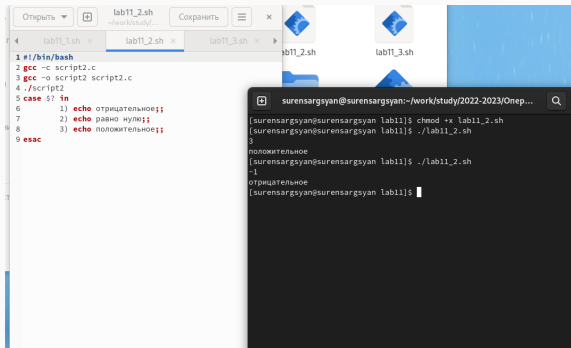
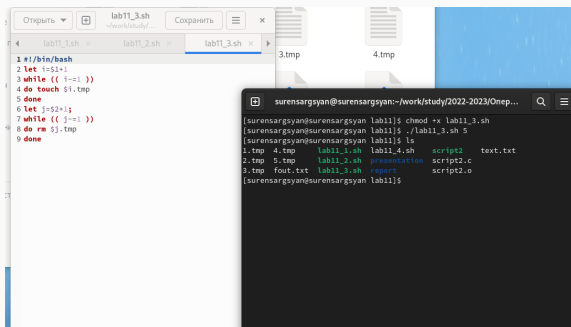


Рис. 2: Задание 2

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 explorer. The terminal window displays the execution of a script named `lab11_3.sh`. The script contains a `while` loop that increments a counter `i` and touches a file `$i.tmp` until `i` reaches 5. The file explorer shows the files `3.tmp` and `4.tmp` created in the directory `~/work/study/2022-2023/Onep...`.

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

```
[surensargssyan@surensargssyan lab11]$ chmod +x lab11_3.sh
[surensargssyan@surensargssyan lab11]$ ./lab11_3.sh 5
[surensargssyan@surensargssyan lab11]$ ls
1.tmp 4.tmp lab11_3.sh lab11_4.sh script2 text.txt
2.tmp 5.tmp lab11_3.sh presentation script2.c
3.tmp fout.txt lab11_3.sh report script2.o
[surensargssyan@surensargssyan 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

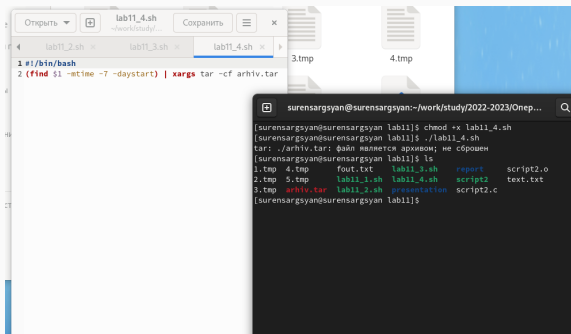


Рис. 4: Задание 4

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.