

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Лия Горелашвили НКАбд-02-23¹

17 апреля, 2024, 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

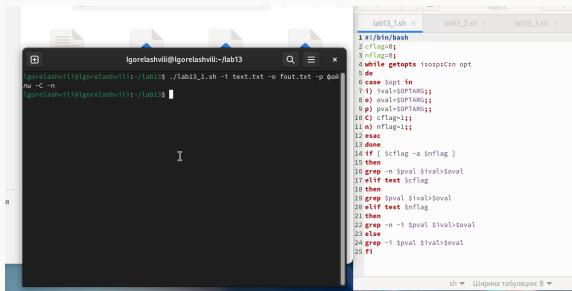
1 Выполнить 4 задания

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 on the left and a code editor on the right. The terminal window is titled 'lgorelashvili@lgorelashvili:~/lab13' and shows the execution of a script 'lab13_1.sh' with arguments '-i text.txt -o fout.txt -p 0aa'. The prompt is 'lgorelashvili@lgorelashvili:~/lab13\$'. The code editor on the right shows the content of 'lab13_1.sh', which is a shell script starting with '#!/bin/bash' and containing various conditional statements and loops.

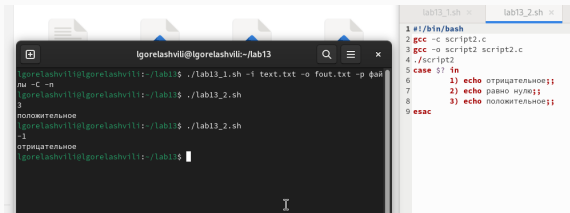
```
lgorelashvili@lgorelashvili:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p 0aa
lgorelashvili@lgorelashvili:~/lab13$
```

```
1 #!/bin/bash
2 cflag=0
3 nflag=0
4 while getopts isopscim opt
5 do
6 case $opt in
7 i) sval=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 c) cflag=1;;
11 m) 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
```

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, titled 'lgorelashvili@lgorelashvili:~/lab13', displays the execution of two shell scripts. The first script, 'lab13_1.sh', is run with arguments '-i text.txt -o fout.txt -p файлы -C -п'. The second script, 'lab13_2.sh', is run without arguments. The output of the second script shows 'положительное' (positive) and 'отрицательное' (negative) results. The code editor, titled 'lab13_1.sh' and 'lab13_2.sh', shows the source code of the scripts. The code in 'lab13_2.sh' is a shell script that uses 'case' to check if a number is positive, zero, or negative, and prints the corresponding result.

```
lgorelashvili@lgorelashvili:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -п
lgorelashvili@lgorelashvili:~/lab13$ ./lab13_2.sh
3
положительное
lgorelashvili@lgorelashvili:~/lab13$ ./lab13_2.sh
-1
отрицательное
lgorelashvili@lgorelashvili:~/lab13$
```

```
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

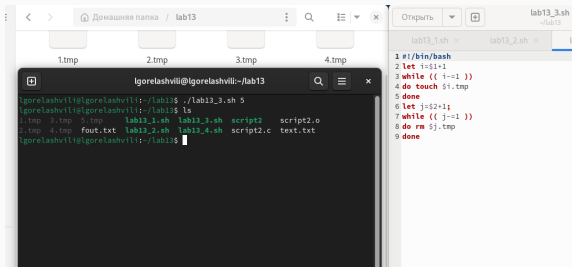
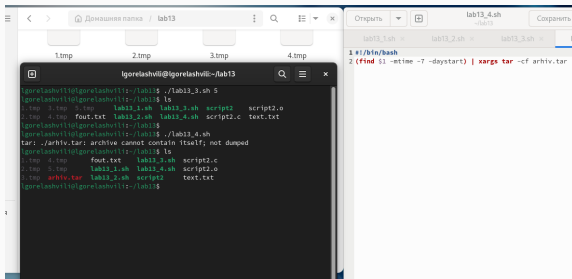


Рис. 3: Задание 3

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



```
lgorelashvill@lgorelashvill:~/lab13$ ./lab13_3.sh 5
lgorelashvill@lgorelashvill:~/lab13$ ls
1.tmp  2.tmp  3.tmp  lab13_1.sh  lab13_3.sh  script2  script2.o
2.tmp  4.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
lgorelashvill@lgorelashvill:~/lab13$
lgorelashvill@lgorelashvill:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
lgorelashvill@lgorelashvill:~/lab13$ ls
1.tmp  4.tmp  fout.txt  lab13_1.sh  lab13_3.sh  script2.c
2.tmp  3.tmp  lab13_2.sh  lab13_4.sh  script2.o
3.tmp  arhiv.tar  lab13_2.sh  script2  text.txt
lgorelashvill@lgorelashvill:~/lab13$
```

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.