

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Одеджими Олуваколаде¹

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

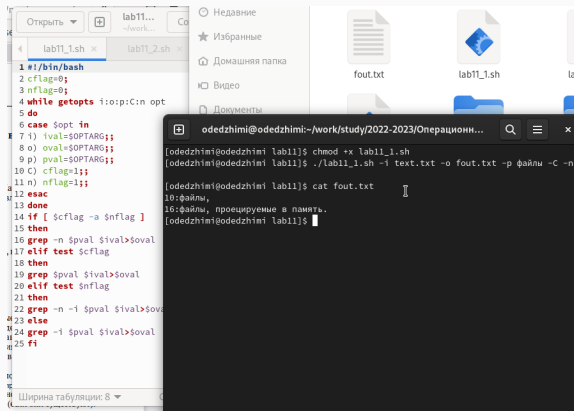
1 Выполнить 4 задания

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. On the left, a text editor window titled 'lab11_1.sh' displays a shell script. The script starts with a shebang, sets flags, and enters a loop that processes command-line options. It uses 'case' for option parsing and 'grep' for file processing. On the right, a file manager window shows the contents of the current directory, including 'fout.txt' and 'lab11_1.sh'. In the foreground, a terminal window shows the execution of the script: 'chmod +x lab11_1.sh' followed by './lab11_1.sh -i text.txt -o fout.txt -p файлы -C -n'. The terminal output shows the script's progress, including file processing and memory projection.

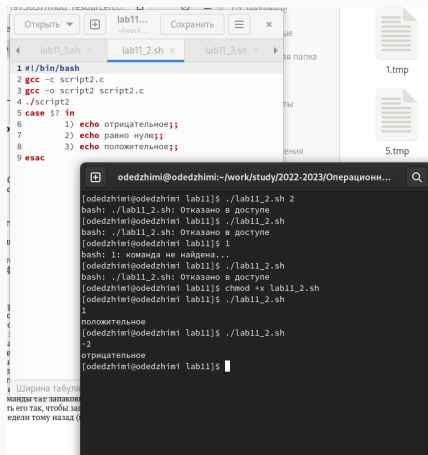
```
#!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:o:p:C:n opt
5 do
6 case $opt in
7 i) ival=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $ival>$oval
17 elif test $cflag
18 then
19 grep $pval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

```
odedzhimi@odedzhimi:~/work/study/2022-2023/Операционн...
[odedzhimi@odedzhimi lab11]$ chmod +x lab11_1.sh
[odedzhimi@odedzhimi lab11]$ ./lab11_1.sh -i text.txt -o fout.txt -p файлы -C -n
[odedzhimi@odedzhimi lab11]$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
[odedzhimi@odedzhimi 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



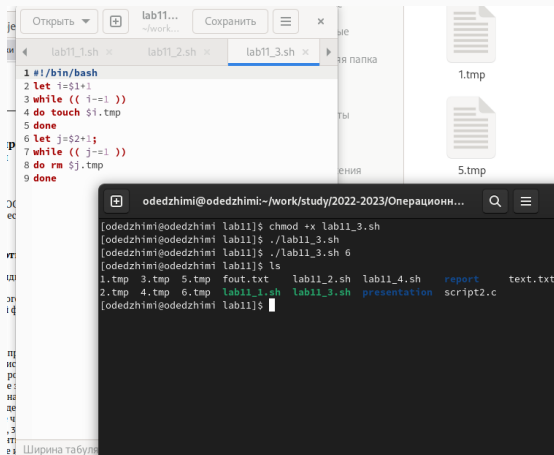
```
1 #!/bin/bash
2 gcc -c script2.c
3 gcc -o script2 script2.c
4 ./script2
5 case $? in
6   1) echo отрицательное;;
7   2) echo равно нулю;;
8   3) echo положительное;;
9 esac
```

```
[odedzhimi@odedzhimi lab11]$ ./lab11_2.sh 2
bash: ./lab11_2.sh: Отказано в доступе
[odedzhimi@odedzhimi lab11]$ ./lab11_2.sh
bash: ./lab11_2.sh: Отказано в доступе
[odedzhimi@odedzhimi lab11]$ 1
bash: 1: команда не найдена...
[odedzhimi@odedzhimi lab11]$ ./lab11_2.sh
bash: ./lab11_2.sh: Отказано в доступе
[odedzhimi@odedzhimi lab11]$ chmod +x lab11_2.sh
[odedzhimi@odedzhimi lab11]$ ./lab11_2.sh
1
положительное
[odedzhimi@odedzhimi lab11]$ ./lab11_2.sh
-2
отрицательное
[odedzhimi@odedzhimi lab11]$
```

Рис. 2: Задание 2

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

Выполнение работы



The screenshot displays a Linux desktop environment. In the background, a text editor window titled 'lab11...' shows a shell script named 'lab11_3.sh'. The script contains the following code:

```
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
```

In the foreground, a terminal window titled 'odedzhimi@odedzhimi:~/work/study/2022-2023/Операционн...' shows the execution of the script. The user has made the script executable and run it with arguments 3 and 6. The output of the 'ls' command is as follows:

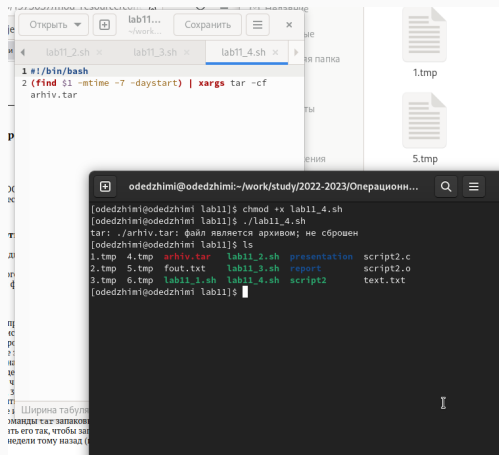
```
[odedzhimi@odedzhimi lab11]$ chmod +x lab11_3.sh
[odedzhimi@odedzhimi lab11]$ ./lab11_3.sh
[odedzhimi@odedzhimi lab11]$ ./lab11_3.sh 6
[odedzhimi@odedzhimi lab11]$ ls
1.tmp  3.tmp  5.tmp  fout.txt  lab11_2.sh  lab11_4.sh  report  text.txt
2.tmp  4.tmp  6.tmp  lab11_1.sh  lab11_3.sh  presentation  script2.c
```

The terminal window also shows the prompt '[odedzhimi@odedzhimi lab11]\$' at the end.

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 with a dark background. The prompt is `odedzhimi@odedzhimi:~/work/study/2022-2023/Операционн...`. The user has executed `chmod +x lab11_4.sh` and then `./lab11_4.sh`. The script output shows a tar command being run, followed by a file listing. The file listing shows a table of files and their sizes.

```
odedzhimi@odedzhimi:~/work/study/2022-2023/Операционн...  
[odedzhimi@odedzhimi lab11]$ chmod +x lab11_4.sh  
[odedzhimi@odedzhimi lab11]$ ./lab11_4.sh  
tar: ./arhiv.tar: файл является архивом; не сброшен  
[odedzhimi@odedzhimi lab11]$ ls  
1.tmp 4.tmp arhiv.tar lab11_2.sh presentation script2.c  
2.tmp 5.tmp fout.txt lab11_3.sh report script2.o  
3.tmp 6.tmp lab11_1.sh lab11_4.sh script2 text.txt  
[odedzhimi@odedzhimi lab11]$
```

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.