

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Диденко Дмитрий Владимирович¹

2 июня, 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

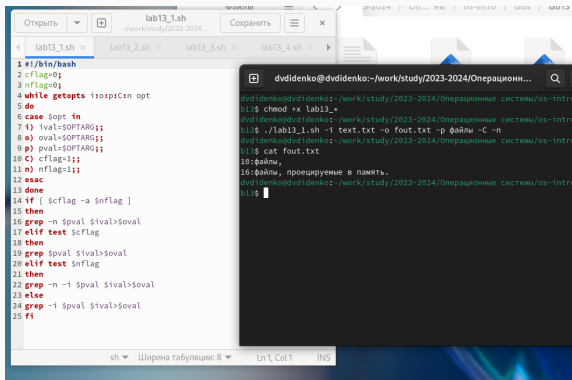
1 Выполнить 4 задания

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 screenshot shows a terminal window with a dark background. The window title is "lab13_1.sh". The script content is as follows:

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

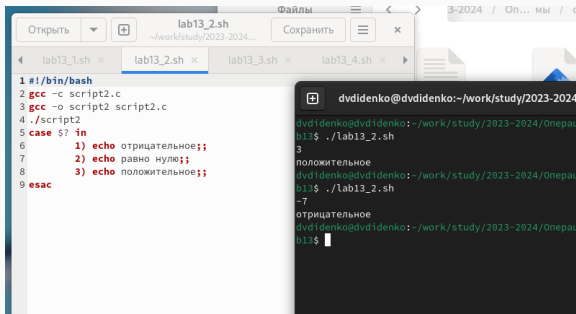
The terminal shows the execution of the script with the following commands and output:

```
dvdidenko@dvdidenko:~/work/study/2023-2024/Операционн...
b13$ chmod +x lab13_1.sh
b13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
b13$ cat fout.txt
16:файлы, проецируемые в память.
b13$
```

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 with a file named `lab13_2.sh` and a terminal window showing the execution of the script.

Code Editor 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
```

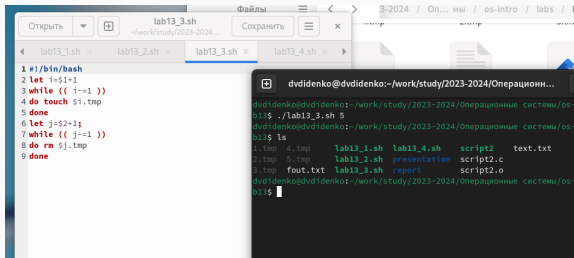
Terminal Content:

```
dvdidenko@dvdidenko:~/work/study/2023-2024/Онепау
b13$ ./lab13_2.sh
3
положительное
dvdidenko@dvdidenko:~/work/study/2023-2024/Онепау
b13$ ./lab13_2.sh
-7
отрицательное
dvdidenko@dvdidenko:~/work/study/2023-2024/Онепау
b13$
```

Рис. 2: Задание 2

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

Выполнение работы



The image shows a file manager window with several tabs labeled 'lab13_1.sh', 'lab13_2.sh', 'lab13_3.sh', and 'lab13_4.sh'. The 'lab13_3.sh' tab is active, displaying the following script:

```
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
```

Overlaid on the file manager is a terminal window. The terminal shows the user 'dvdidenko' at a prompt, navigating to the directory '/work/study/2023-2024/Операционные системы/os-b13' and running the script './lab13_3.sh 5'. The output of the script is displayed as a table:

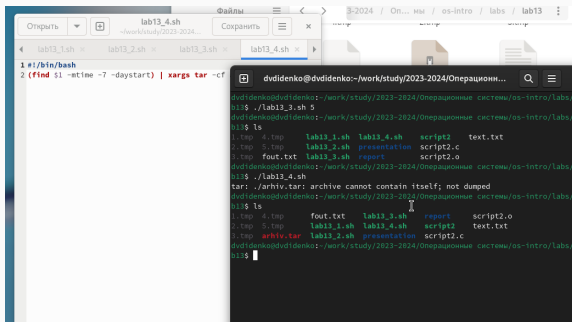
	lab13_1.sh	lab13_4.sh	script2	text.txt
1.tmp	4.tmp			
2.tmp	5.tmp	presentation	script2.c	
3.tmp	fout.txt	lab13_3.sh	report	script2.o

The terminal prompt returns to 'b13\$' after the script execution.

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 screenshot shows a terminal window with a file manager interface at the top. The terminal displays the following commands and output:

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

The output shows the execution of the script and the creation of a tar archive:

```
dvddenko@dvddenko:~/work/study/2023-2024/Операционные системы/os-intro/labs/
b13$ ./lab13_3.sh 5
dvddenko@dvddenko:~/work/study/2023-2024/Операционные системы/os-intro/labs/
b13$ ls
1.tmp 4.tmp lab13_1.sh lab13_4.sh script2 text.txt
2.tmp 5.tmp lab13_2.sh presentation script2.c
3.tmp fout.txt lab13_3.sh report script2.o
dvddenko@dvddenko:~/work/study/2023-2024/Операционные системы/os-intro/labs/
b13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
dvddenko@dvddenko:~/work/study/2023-2024/Операционные системы/os-intro/labs/
b13$ ls
1.tmp 4.tmp fout.txt lab13_1.sh report script2.o
2.tmp 5.tmp lab13_2.sh lab13_4.sh script2 text.txt
3.tmp arhiv.tar lab13_3.sh presentation script2.c
dvddenko@dvddenko:~/work/study/2023-2024/Операционные системы/os-intro/labs/
b13$
```

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.