

Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Быкасов Владислав Дмитриевич

27 июня 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

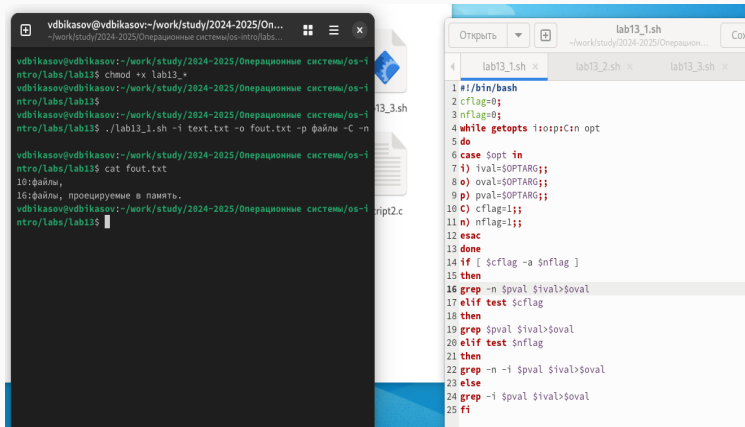
1 Выполнить 4 задания

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; `-p 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

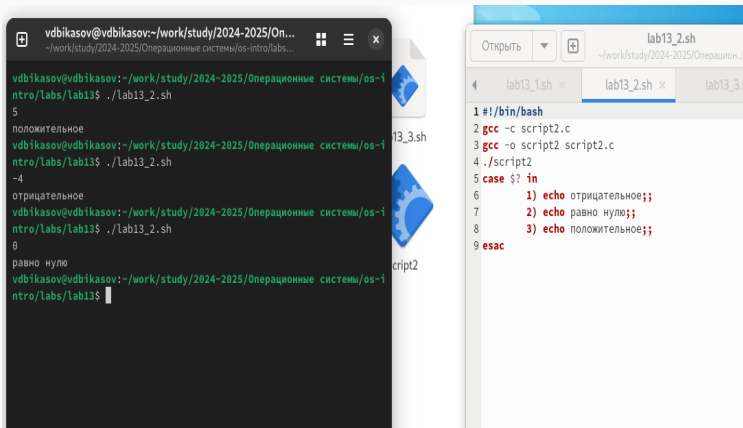


```
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs...  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ chmod +x lab13_1.sh  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n  
10:файлы,  
16:файлы, проецируемые в память.  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

```
#!/bin/bash  
2 cflag=0;  
3 nflag=0;  
4 while getopts i:op: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
```

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



The image shows two overlapping windows. The background window is a terminal with a dark theme, showing the execution of a script. The foreground window is a script editor with a light theme, showing the source code of the script.

Terminal Window:

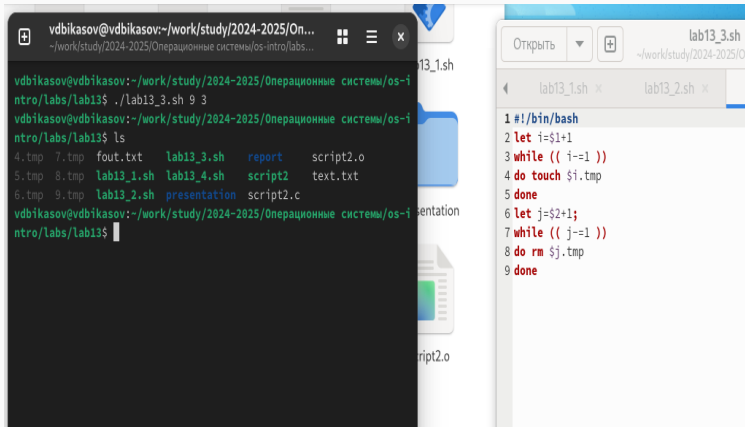
```
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs...  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_2.sh  
5  
положительное  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_2.sh  
-4  
отрицательное  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_2.sh  
0  
равно нулю  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

Script Editor Window (lab13_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



The image shows two overlapping windows from a Linux desktop environment. The background window is a terminal with the following content:

```
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs...  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_3.sh 9 3  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ls  
4.tmp 7.tmp fout.txt lab13_3.sh report script2.o  
5.tmp 8.tmp lab13_1.sh lab13_4.sh script2 text.txt  
6.tmp 9.tmp lab13_2.sh presentation script2.c  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

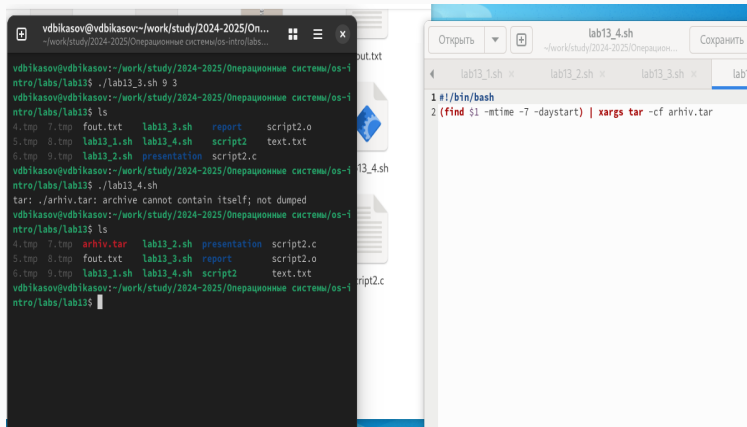
The foreground window is a text editor titled 'lab13_3.sh' showing the following script code:

```
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
```

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 two side-by-side windows. The left window is a terminal with the following content:

```
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs...  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_3.sh 9 3  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ls  
4.tmp 7.tmp fout.txt lab13_3.sh report script2.o  
5.tmp 8.tmp lab13_1.sh lab13_4.sh script2 text.txt  
6.tmp 9.tmp lab13_2.sh presentation script2.c  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_4.sh  
tar: ./arhiv.tar: archive cannot contain itself; not dumped  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ls  
4.tmp 7.tmp arhiv.tar lab13_2.sh presentation script2.c  
5.tmp 8.tmp fout.txt lab13_3.sh report script2.o  
6.tmp 9.tmp lab13_1.sh lab13_4.sh script2 text.txt  
vdbikasov@vdbikasov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

The right window is a file manager showing the contents of the 'lab13_4.sh' file:

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

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.