

Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Хаджилари Гешлаг Никта

23 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.

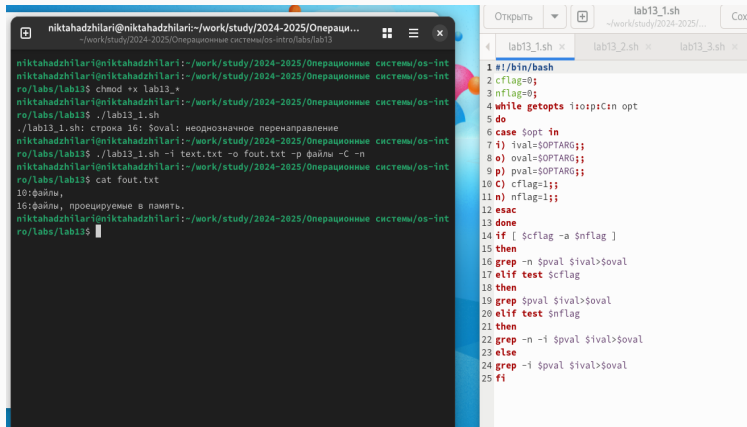
1 Выполнить 4 задания

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



The image shows a terminal window on the left and a code editor on the right. The terminal window displays the execution of a script named `lab13_1.sh`. The user runs `chmod +x lab13_*` and then `./lab13_1.sh`. The script outputs the following messages:

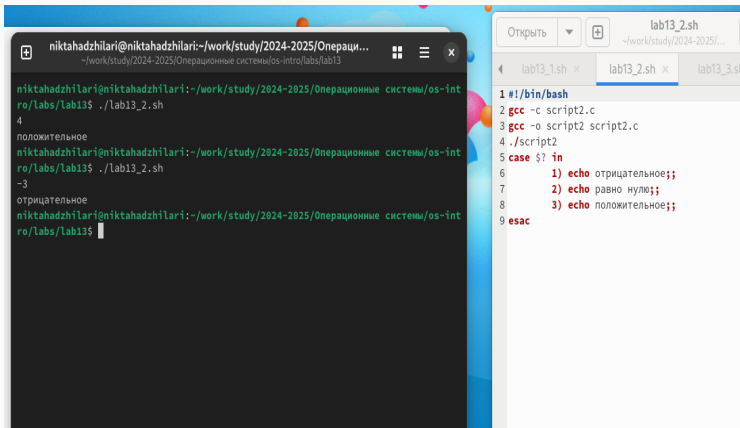
```
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
```

The code editor on the right shows the content of `lab13_1.sh`:

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

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



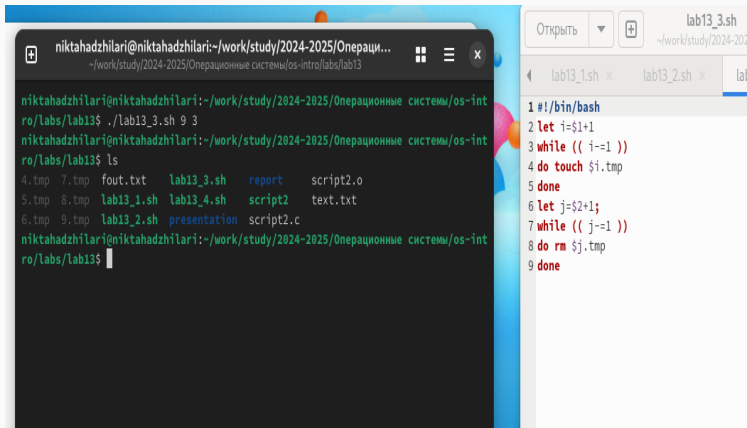
The image shows a terminal window on the left and a code editor on the right. The terminal window displays the execution of a script named `lab13_2.sh`. The user `niktahadzhilari` runs the script, which outputs the number `4` and the word `положительное` (positive). The user then runs the script again, which outputs `-3` and the word `отрицательное` (negative). The code editor on the right shows the source code of `lab13_2.sh`. The script starts with a shebang `#!/bin/bash`, compiles a C program `script2.c` using `gcc`, and then runs it. It uses a `case` statement to check the output of the program and print the corresponding result.

```
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_2.sh
4
положительное
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_2.sh
-3
отрицательное
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

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

Рис. 2: Задание 2

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



The image shows a terminal window and a file editor side-by-side. The terminal window, titled 'niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13', displays the execution of a script and the resulting file listing. The file editor, titled 'lab13_3.sh', shows the content of the script being executed.

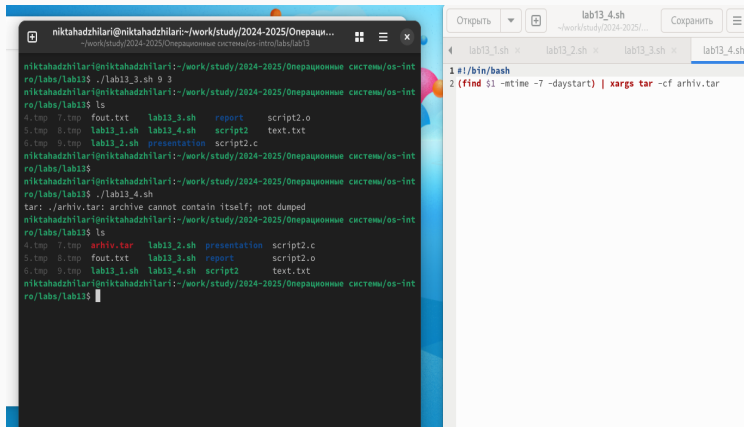
```
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_3.sh 9 3
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ls
4.tmp  7.tmp  fout.txt  lab13_3.sh  report  script2.o
5.tmp  8.tmp  lab13_1.sh  lab13_4.sh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

```
#!/bin/bash
1 let i=$1+1
2 while (( i--=1 ))
3 do touch $i.tmp
4 done
5 let j=$2+1;
6 while (( j--=1 ))
7 do rm $j.tmp
8 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 a terminal window on the left and a file manager on the right. The terminal window displays the execution of a script named `lab13_3.sh` and the subsequent creation of a tar archive named `arhiv.tar`. The file manager shows the contents of the `lab13_4.sh` script, which includes a `find` command to locate files older than 7 days and a `xargs tar` command to archive them.

```
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_3.sh 9 3
niktahadzhilari@niktahadzhilari:~/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
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
niktahadzhilari@niktahadzhilari:~/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
niktahadzhilari@niktahadzhilari:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

The file manager shows the `lab13_4.sh` script with the following content:

```
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.