

# Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Виктория Копылова

18 апреля 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

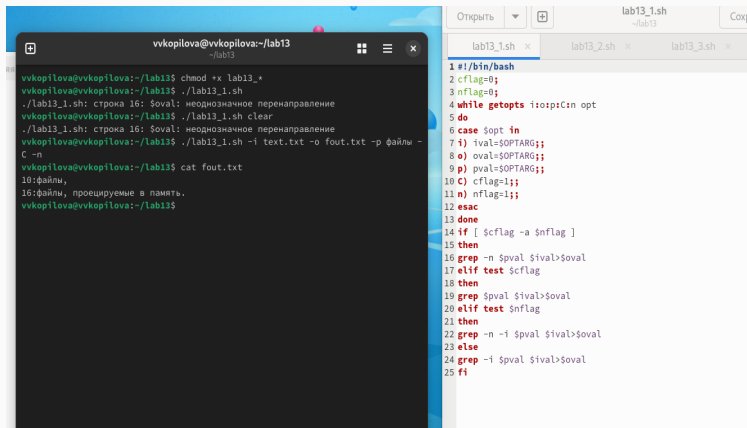
## Процесс выполнения лабораторной работы

---

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, titled 'vvpkopilova@vvpkopilova:~/lab13', shows the execution of a script 'lab13\_1.sh'. The script sets permissions, runs itself, and then uses 'clear' and 'cat' to display error messages. The code editor, titled 'lab13\_1.sh', shows the script's source code, which includes a 'while' loop and a 'case' statement for argument parsing.

```
vvpkopilova@vvpkopilova:~/lab13$ chmod +x lab13_*
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh clear
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
vvpkopilova@vvpkopilova:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
vvpkopilova@vvpkopilova:~/lab13$
```

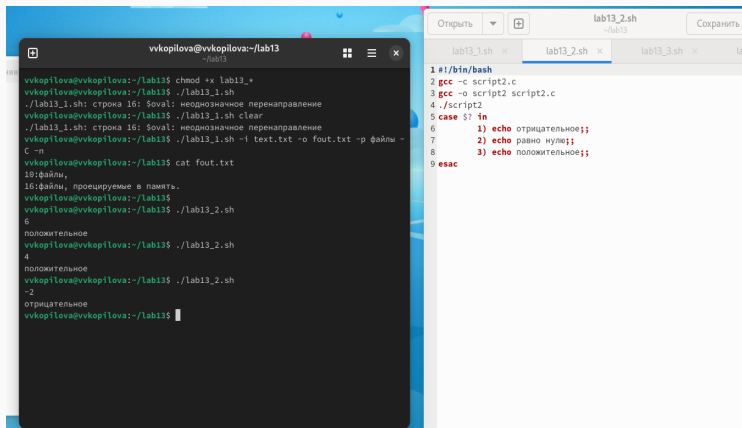
```
1#!/bin/bash
2cflag=0;
3nflag=0;
4while getopts i:op:C:n opt
5do
6case $opt in
7i) ival=$OPTARG;;
8o) oval=$OPTARG;;
9p) pval=$OPTARG;;
10C) cflag=1;;
11n) nflag=1;;
12esac
13done
14if [ $cflag -a $nflag ]
15then
16grep -n $pval $ival>$oval
17elif test $cflag
18then
19grep $pval $ival>$oval
20elif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

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 on the left and a code editor on the right. The terminal window, titled 'vvpkopilova@vvpkopilova:~/lab13', shows the execution of a shell script 'lab13\_1.sh' and the contents of a file 'fout.txt'. The code editor, titled 'lab13\_2.sh', shows the source code of the script, which includes a case statement for handling command-line arguments.

```
vvpkopilova@vvpkopilova:~/lab13$ chmod +x lab13_*
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh clear
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -
C -n
vvpkopilova@vvpkopilova:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
6
положительное
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
4
положительное
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
-2
отрицательное
vvpkopilova@vvpkopilova:~/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

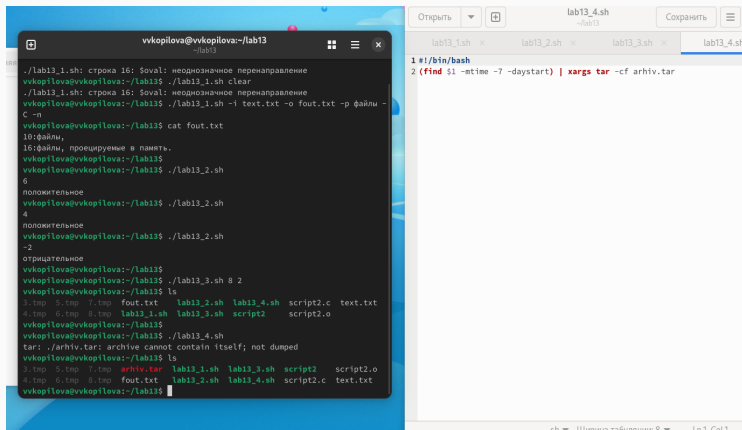
The image shows a terminal window on the left and a file editor on the right. The terminal window, titled 'vvpkopilova@vvpkopilova:~/lab13', shows the execution of several shell scripts. The file editor, titled 'lab13\_3.sh', shows the source code of the script being executed.

```
vvpkopilova@vvpkopilova:~/lab13$ chmod +x lab13_*
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh clear
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -
C -n
vvpkopilova@vvpkopilova:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
vvpkopilova@vvpkopilova:~/lab13$
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
6
положительное
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
4
положительное
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_2.sh
-2
отрицательное
vvpkopilova@vvpkopilova:~/lab13$
vvpkopilova@vvpkopilova:~/lab13$ ./lab13_3.sh 8 2
vvpkopilova@vvpkopilova:~/lab13$ ls
3.tmp 5.tmp 7.tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
4.tmp 6.tmp 8.tmp lab13_1.sh lab13_3.sh script2 script2.o
vvpkopilova@vvpkopilova:~/lab13$
```

```
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.



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 `lab13_4.sh` with various commands and their outputs. The file manager shows the contents of the `lab13_4.sh` file, which includes a `find` command and a `xargs tar` command.

```
vvkopi1ova@vvkopi1ova:~/lab13
~/lab13
./lab13_1.sh: строка 16: $oal: неоднозначное перенаправление
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_1.sh clear
./lab13_1.sh: строка 16: $oal: неоднозначное перенаправление
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -
C -n
vvkopi1ova@vvkopi1ova:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
vvkopi1ova@vvkopi1ova:~/lab13$
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_2.sh
6
положительное
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_2.sh
4
положительное
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_2.sh
-2
отрицательное
vvkopi1ova@vvkopi1ova:~/lab13$
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_3.sh 8 2
vvkopi1ova@vvkopi1ova:~/lab13$ ls
3.tmp 5.tmp 7.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
4.tmp 6.tmp 8.tmp  lab13_1.sh  lab13_3.sh  script2    script2.o
vvkopi1ova@vvkopi1ova:~/lab13$
vvkopi1ova@vvkopi1ova:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
vvkopi1ova@vvkopi1ova:~/lab13$ ls
3.tmp 5.tmp 7.tmp  arhiv.tar  lab13_1.sh  lab13_3.sh  script2    script2.o
4.tmp 6.tmp 8.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
vvkopi1ova@vvkopi1ova:~/lab13$
```

The file manager shows 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.