

# Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Алиев Руслан Нияз оглы

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; `-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 two side-by-side windows from a Linux environment. The left window is a terminal titled 'ruslanaliev@ruslanaliev:~/lab13'. It shows the execution of a script 'lab13\_1.sh' which sets permissions, runs itself, and processes a file 'text.txt' into 'fout.txt'. It also shows the user running 'at fout.txt' and 'cat fout.txt'. The right window is a file editor titled 'lab13\_1.sh' showing the script's source code. The code includes a shebang, variable initialization, a loop with a case statement for argument parsing, and several grep commands for file processing.

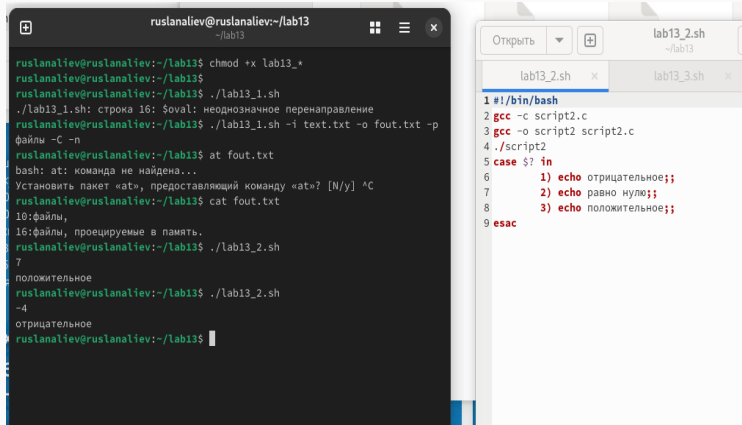
```
ruslanaliev@ruslanaliev:~/lab13$ chmod +x lab13_*
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p
файлы -C -n
ruslanaliev@ruslanaliev:~/lab13$ at fout.txt
bash: at: команда не найдена...
Установить пакет «at», предоставляющий команду «at»? [N/y] ^C
ruslanaliev@ruslanaliev:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
ruslanaliev@ruslanaliev:~/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





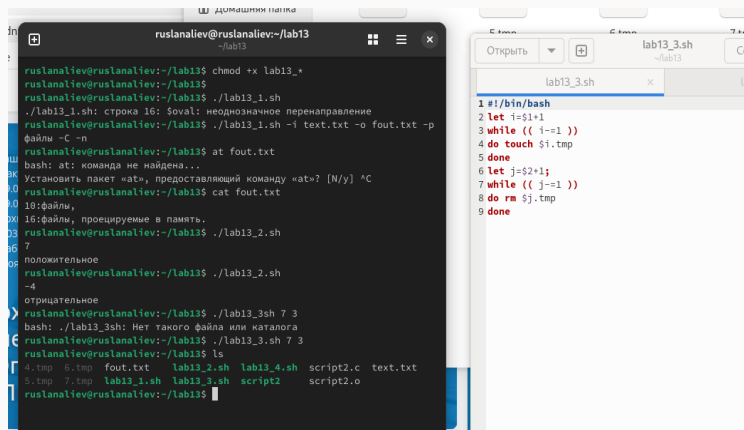
The image shows a terminal window on the left and a script file on the right. The terminal window is titled 'ruslanaliev@ruslanaliev:~/lab13' and shows the execution of a script 'lab13.sh'. The script performs several operations: it sets permissions on 'lab13\_\*', runs './lab13\_1.sh', and then './lab13\_1.sh -i text.txt -o fout.txt -p файлы -C -n'. It then attempts to run 'at fout.txt' but fails with the message 'bash: at: команда не найдена...'. The user is prompted to install the 'at' package, which they do. Finally, the user runs 'cat fout.txt', which outputs '10:файлы, 16:файлы, проецируемые в память.' and then '7 положительное'. The user then runs './lab13\_2.sh', which outputs '-4 отрицательное'. The script file on the right is titled 'lab13\_2.sh' and contains the following code:

```
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

# Выполнение работы



The image shows two windows from a Linux desktop environment. The left window is a terminal titled 'ruslanaliev@ruslanaliev:~/lab13'. It displays the following commands and their outputs:

```
ruslanaliev@ruslanaliev:~/lab13$ chmod +x lab13_*
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p
файлы -C -n
ruslanaliev@ruslanaliev:~/lab13$ at fout.txt
bash: at: команда не найдена...
Установить пакет «at», предоставляющий команду «at»? [N/y] ^C
ruslanaliev@ruslanaliev:~/lab13$ cat fout.txt
18:файлы,
16:файлы, проецируемые в память.
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_2.sh
7
положительное
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_2.sh
-4
отрицательное
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_3sh 7 3
bash: ./lab13_3sh: Нет такого файла или каталога
ruslanaliev@ruslanaliev:~/lab13$ ./lab13_3.sh 7 3
ruslanaliev@ruslanaliev:~/lab13$ ls
4.tmp 6.tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
5.tmp 7.tmp lab13_1.sh lab13_3.sh script2 script2.o
ruslanaliev@ruslanaliev:~/lab13$
```

The right window is a file manager titled 'lab13\_3.sh'. It shows the contents of the file 'lab13\_3.sh' with 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
```

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

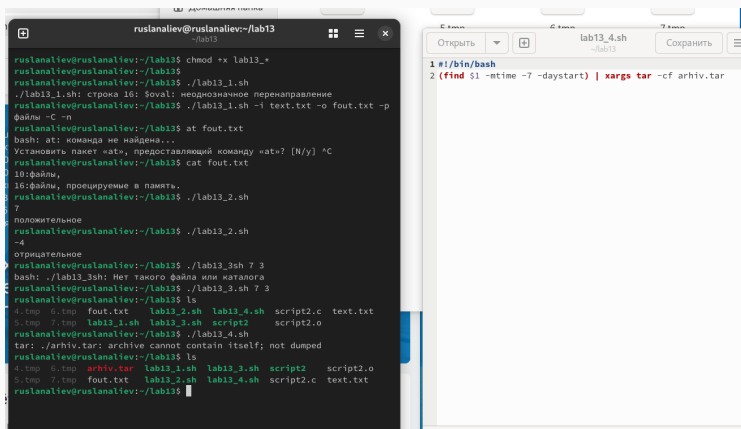


Рис. 4: Задание 4

## 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.