

# Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Байрам Эргешов<sup>1</sup>

19 апреля, 2024, Москва, Россия

<sup>1</sup>Российский 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

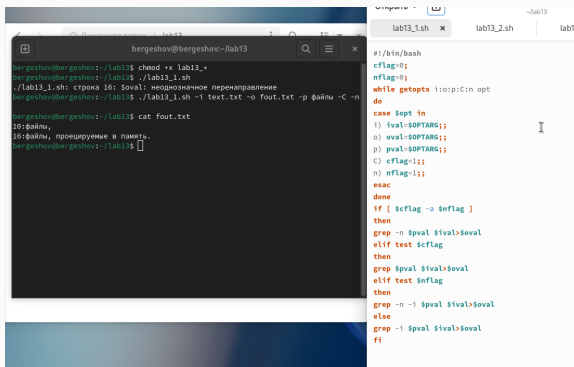
# **Процесс выполнения лабораторной работы**

---

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 image shows a terminal window and a code editor side-by-side. The terminal window, titled 'bergeshov@bergeshov:~/lab13', shows the execution of a script 'lab13\_1.sh' with arguments '16' and 'foval'. The script outputs '16:foval, procedure in memory.' and '16:foval, procedure in memory.' to a file 'fout.txt'. The code editor, titled 'lab13\_1.sh', shows the script's source code, which includes a case statement for 'foval' and a while loop for 'getopts'.

```
bergeshov@bergeshov:~/lab13$ chmod +x lab13_1.sh
bergeshov@bergeshov:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
bergeshov@bergeshov:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p foval -C -n
bergeshov@bergeshov:~/lab13$ cat fout.txt
16:foval,
16:foval, procedure in memory.
bergeshov@bergeshov:~/lab13$
```

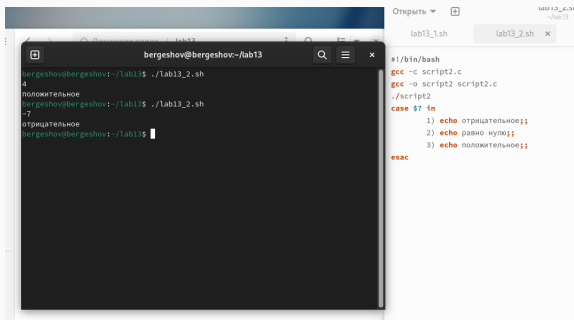
```
#!/bin/bash
cflag=0;
nflag=0;
while getopts i:osp:C:n opt
do
case $opt in
s) ival=$OPTARG;;
o) oval=$OPTARG;;
p) pval=$OPTARG;;
C) cflag=1;;
n) nflag=1;;
esac
done
if [ $cflag -a $nflag ]
then
grep -n $pval $ival>$oval
elif test $cflag
then
grep $pval $ival>$oval
elif test $nflag
then
grep -n -i $pval $ival>$oval
else
grep -i $pval $ival>$oval
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



# Выполнение работы



The image shows a terminal window and a code editor. The terminal window, titled 'bergeshov@bergeshov:~/lab13', displays the execution of a script named 'lab13\_2.sh'. The script takes an argument '4' and prints 'положительное' (positive), then takes an argument '-7' and prints 'отрицательное' (negative). The code editor, titled 'lab13\_2.sh', shows the script's content, which includes compilation with 'gcc', execution with './script2', and a case statement that prints 'отрицательное' for negative numbers, 'равно нулю' (equal to zero) for zero, and 'положительное' for positive numbers.

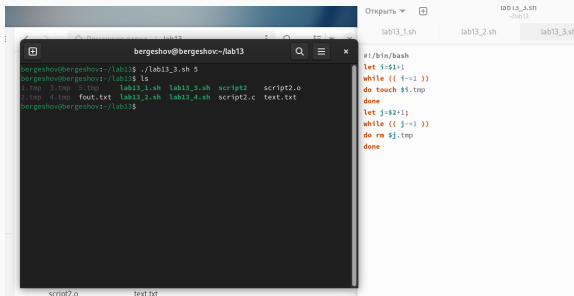
```
bergeshov@bergeshov:~/lab13$ ./lab13_2.sh
4
положительное
bergeshov@bergeshov:~/lab13$ ./lab13_2.sh
-7
отрицательное
bergeshov@bergeshov:~/lab13$
```

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

Рис. 2: Задание 2

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

# Выполнение работы



The image shows a terminal window and a file explorer. The terminal window, titled 'bergeshov@bergeshov:~/lab13', displays the following commands and output:

```
bergeshov@bergeshov:~/lab13$ ./lab13_3.sh 5
bergeshov@bergeshov:~/lab13$ ls
1.tmp  3.tmp  5.tmp  lab13_1.sh  lab13_3.sh  script2  script2.o
2.tmp  4.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
bergeshov@bergeshov:~/lab13$
```

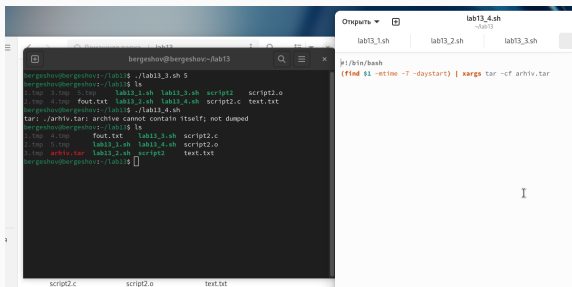
The file explorer shows a directory named 'lab13' containing several files, including 'lab13\_1.sh', 'lab13\_2.sh', 'lab13\_3.sh', 'lab13\_4.sh', 'script2', 'script2.c', 'text.txt', and '1.tmp'. The file 'lab13\_3.sh' is selected, and its contents are displayed in a separate window:

```
#!/bin/bash
let i=$1+1
while (( i-->1 ))
do touch $i.tmp
done
let j=$2+1
while (( j-->1 ))
do rm $j.tmp
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



```
bergeshov@bergeshov:~/lab13$ ./lab13_3.sh 5
bergeshov@bergeshov:~/lab13$ ls
1.tmp  3.tmp  5.tmp  lab13_1.sh  lab13_3.sh  script2  script2.o
2.tmp  4.tmp  fout.txt lab13_2.sh  lab13_4.sh  script2.c  text.txt
bergeshov@bergeshov:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
bergeshov@bergeshov:~/lab13$ ls
1.tmp  4.tmp  fout.txt  lab13_3.sh  script2.c
2.tmp  5.tmp  lab13_1.sh  lab13_4.sh  script2.o
3.tmp  arhiv.tar  lab13_2.sh  script2  text.txt
bergeshov@bergeshov:~/lab13$
```

script2.c      script2.o      text.txt

lab13\_4.sh  
~/lab13

lab13\_1.sh      lab13\_2.sh      lab13\_3.sh

~/bin/bash

(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.