

# Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Зырянов Артём Алексеевич НБИбд-01-22<sup>1</sup>

17 апреля, 2023, 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**

---

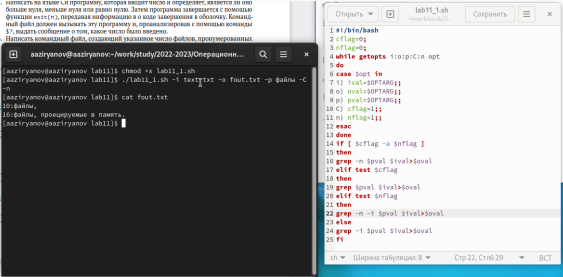
1. Используя команды `getopts` `grep` напишем командный файл, который анализирует командную строку с ключами и выполним его: `-i inputfile` — прочитать данные из указанного файла; `-o outputfile` — вывести данные в указанный файл; `-r шаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк;

а затем ищет в указанном файле нужные строки

# Выполнение работы

4. написать на языке C++ программу, которая выводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

3. Написать командный файл, создающий указанное число файлов, проанализировав



```
aziziryanov@aziziryanov:~/work/study/2022-2023/Операционн...  
[aziziryanov@aziziryanov lab11]$ chmod +x lab11_1.sh  
[aziziryanov@aziziryanov lab11]$ ./lab11_1.sh -i text.txt -o fout.txt -p файлы -C n  
[aziziryanov@aziziryanov lab11]$ cat fout.txt  
10:файлы,  
16:файлы, просеиваемые в память.  
[aziziryanov@aziziryanov lab11]$
```

```
1 #!/bin/bash  
2 cflag=0  
3 nflag=0  
4 while getopts i:oi:p:C:n opt  
5 do  
6 case $opt in  
7 i) sval=$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 $sval>$oval  
17 elif test $cflag  
18 then  
19 grep $pval $sval>$oval  
20 elif test $nflag  
21 then  
22 grep -n -i $pval $sval>$oval  
23 else  
24 grep -i $pval $sval>$oval  
25 fi  
sh Шириня табуляция: 8 Стр 22, Стр 29 BCT
```

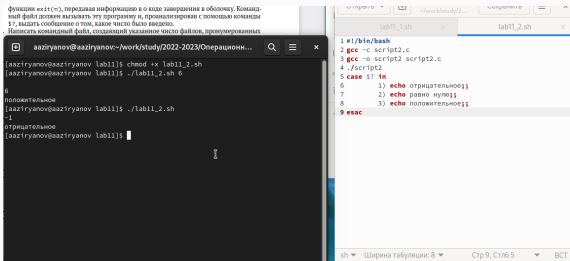
Рис. 1: Задание 1

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

функции `exit(=)`, передавая информацию в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `if`, выдать сообщение о том, какое число было введено.  
Написать командный файл, создающий указанное число файлов, пронумерованных



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 `lab11_2.sh`. The user runs `chmod +x lab11_2.sh` and then `./lab11_2.sh 6`. The script outputs "положительное" (positive) for the number 6, "-1" for the number -1, and "отрицательное" (negative) for the number -1. The code editor on the right shows the source code of `lab11_2.sh`, which is a shell script that takes a number as input and uses a `case` statement to output "отрицательное" for negative numbers, "равно нулю" (equal to zero) for zero, and "положительное" for positive numbers.

```
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

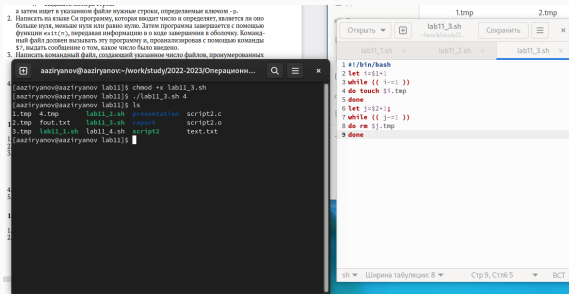
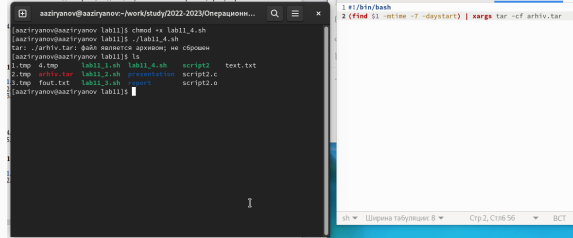


Рис. 3: Задание 3

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

- а затем идет в удаленном файле нужные строки, определенные ключом -r;
- Написать на языке Си программу, которая выводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции «exit(с)», передавая информацию в оболочку. Командный файл должен выводить эту программу и, проанализировав с помощью команды \$?, выдать сообщение о том, какое число было введено.
- Написать командный файл, создающий указанное число файлов, проанализированных



The image shows a terminal window on the left and a file manager window on the right. The terminal window displays the following commands and output:

```
aaziryanov@aaziryanov:~/work/study/2022-2023/Операционные...  
4 [aaziryanov@aaziryanov lab11]$ chmod +x lab11_4.sh  
[aaziryanov@aaziryanov lab11]$ ./lab11_4.sh  
tar: ./arhiv.tar: файл является архивом; не сбрасыв  
[aaziryanov@aaziryanov lab11]$ ls  
1.tmp 4.tmp lab11_3.sh lab11_4.sh script2 text.txt  
2.tmp arhiv.tar lab11_2.sh presentation script2.c  
3.tmp fout.txt lab11_3.sh report script2.o  
[aaziryanov@aaziryanov lab11]:
```

The file manager window shows a directory with files lab11\_2.sh, lab11\_3.sh, and lab11\_4.sh. The file lab11\_4.sh is selected, and its contents are displayed in the right pane:

```
1 #!/bin/bash  
2 ((find $1 -mtime -T -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.