

# Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Арвин Мелкумян<sup>1</sup>

1 сентября, 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

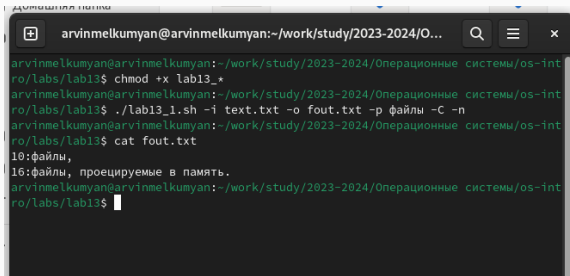
# **Процесс выполнения лабораторной работы**

---

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

A terminal window with a dark background and light green text. The window title bar shows the user 'arvinmelkumyan' and the current directory '~/work/study/2023-2024/O...'. The terminal content shows a series of commands and their outputs. The user sets permissions for 'lab13\_\*', runs a script 'lab13\_1.sh' with input 'text.txt' and output 'fout.txt', and then uses 'cat' to view the output. The output lists '10:файлы,' and '16:файлы, проецируемые в память.'.

```
arvinmelkumyan@arvinmelkumyan:~/work/study/2023-2024/Операционные системы/os-int
ro/labs/lab13$ chmod +x lab13_*
arvinmelkumyan@arvinmelkumyan:~/work/study/2023-2024/Операционные системы/os-int
ro/labs/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
arvinmelkumyan@arvinmelkumyan:~/work/study/2023-2024/Операционные системы/os-int
ro/labs/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
arvinmelkumyan@arvinmelkumyan:~/work/study/2023-2024/Операционные системы/os-int
ro/labs/lab13$
```

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

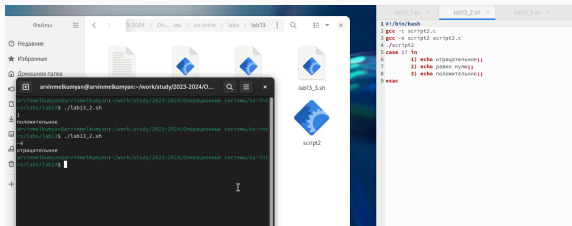


Рис. 2: Задание 2

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

# Выполнение работы

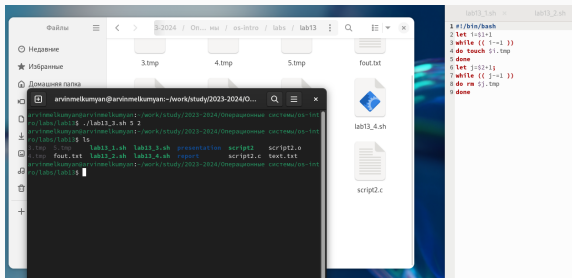


Рис. 3: Задание 3

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

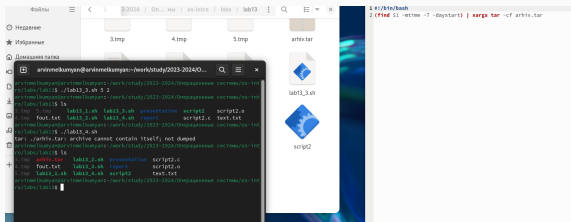


Рис. 4: Задание 4

## **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.