

# 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

## Выполнение работы

1. Используя команды `grep` напомним 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

```
[keisaev@fedora lab13]$ ./lab13_2.sh
./lab13_2.sh: строка 7: синтаксическая ошибка рядом
./lab13_2.sh: строка 7: `      2) echo равно нулю
[keisaev@fedora lab13]$ ./lab13_2.sh
равно нулю
```

{ #fig:001 width=70% height=70%}

## Выполнение работ

```
< lab13_2.sh x lab13_3.
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
```

ы

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

```
[keisaev@fedora ~]$ ./lab13_3.sh 8 3
bash: ./lab13_3.sh: Нет такого файла или каталога
[keisaev@fedora ~]$ cd ~/work/study/2024-2025/Операционные\ системы/os-intro/labs/lab13/
[keisaev@fedora lab13]$ ./lab13_3.sh 8 3
[keisaev@fedora lab13]$ ls
4.tmp 6.tmp 8.tmp fout.txt lab13_2.sh lab13_4.sh report script2.c text.txt
5.tmp 7.tmp 9.tmp lab13_1.sh lab13_3.sh presentation script2 script2.o
[keisaev@fedora lab13]$
```

{ #fig:002 width=70% height=70%}

## Выполнение работы

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

## Выполнение работ

```
< lab13_2.sh x lab13_3.sh x
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
```

ы

```
[keisaev@fedora ~]$ cd ~/work/study/2024-2025/Операционные\ системы/os-intro/labs/lab13/
[keisaev@fedora lab13]$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
bash: ./lab13_1.sh: Отказано в доступе
[keisaev@fedora lab13]$ chmod +x lab13_*
[keisaev@fedora lab13]$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
./lab13_1.sh: недопустимый параметр —C
grep: text.txt: Нет такого файла или каталога
[keisaev@fedora lab13]$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
grep: text.txt: Нет такого файла или каталога
[keisaev@fedora lab13]$ touch text.txt
[keisaev@fedora lab13]$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
[keisaev@fedora lab13]$ cat fout.txt
[keisaev@fedora lab13]$
```

#fig:003 width=70% height=70%}

```
lab13_1.sh x lab13_1
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:o:p:C:n opt
5 do
6 case $opt in
7 i) ival=$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 $ival>$oval
17 elif test $cflag
18 then
19 grep $pval $ival>$oval
20 elif test $nflag
21 then
22 grep -n -i $pval $ival>$oval
23 else
24 grep -i $pval $ival>$oval
25 fi
```

**Выполнение работ**

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**

```
[keisaev@fedora ~]$ cd ~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13/
[keisaev@fedora lab13]$ ./lab13_3.sh 8 3
[keisaev@fedora lab13]$ ls
4.tmp 6.tmp 8.tmp  fout.txt  lab13_2.sh  lab13_4.sh  report  script2.c  text.txt
5.tmp 7.tmp 9.tmp  lab13_1.sh  lab13_3.sh  presentation  script2  script2.o
[keisaev@fedora lab13]$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
[keisaev@fedora lab13]$ ls
4.tmp 6.tmp 8.tmp  arhiv.tar  lab13_1.sh  lab13_3.sh  presentation  script2  script2.o
5.tmp 7.tmp 9.tmp  fout.txt  lab13_2.sh  lab13_4.sh  report  script2.c  text.txt
[keisaev@fedora lab13]$
```

```
lab13_2.sh x lab13_3.sh x lab13_4.sh x
1 #!/bin/bash
2 (find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar
```

{ #fig:004 width=70% height=70%}

**Выводы по проделанной работе**

## Вывод

В данной работе мы изучили основы программирования в оболочке ОС UNIX и писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.