

# **Отчёт по лабораторной работе №11**

**Программирование в командном процессоре ОС UNIX. Ветвления и  
циклы**

Мокочунина Влада Сергеевна

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	15

## Список иллюстраций

3.1	Командный файл . . . . .	8
3.2	Проверка . . . . .	9
3.3	Командный файл на Си . . . . .	10
3.4	Командный файл . . . . .	11
3.5	Проверка . . . . .	12
3.6	Командный файл . . . . .	13
3.7	Проверка . . . . .	13
3.8	Командный файл . . . . .	14
3.9	Проверка . . . . .	14

## **Список таблиц**

# 1 Цель работы

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

## 2 Задание

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

### 3 Выполнение лабораторной работы

1. Используя команды `getopts` `grep`, написала командный файл, который анализирует командную строку с ключами: `-iinputfile` — прочитать данные из указанного файла; `-ooutputfile` — вывести данные в указанный файл; `-rшаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк.(рис. [3.1]).



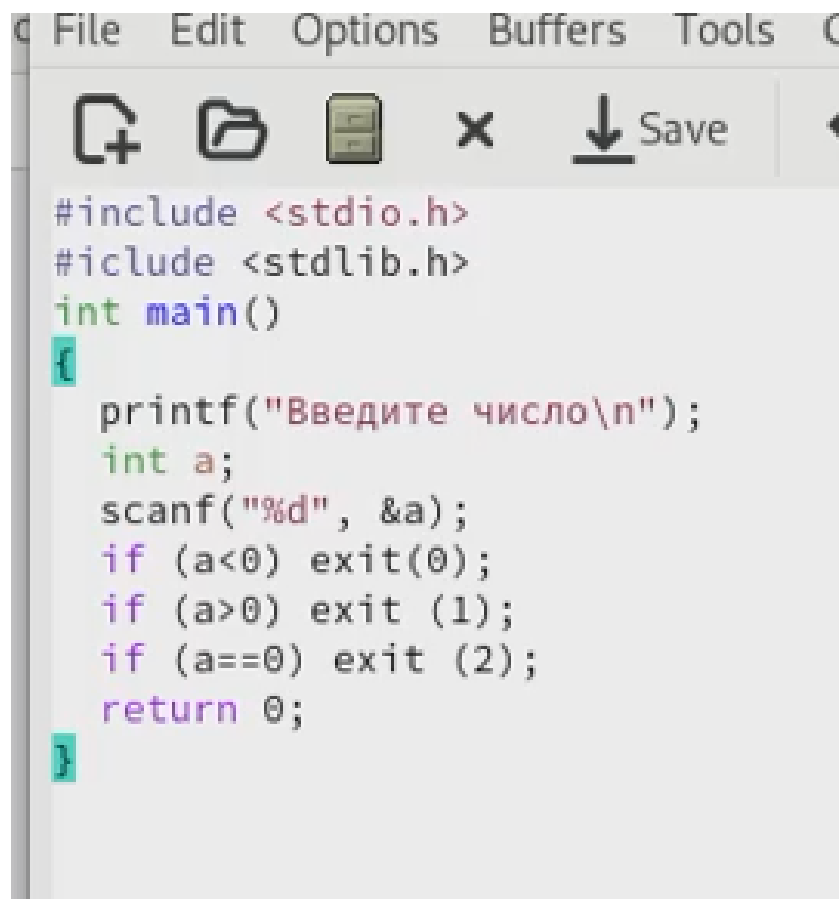


## 2. Проверила на выполнение

```
[vsmokochunina@10 ~]$ cat 1.txt
Hello
HELLO
hello
text
[vsmokochunina@10 ~]$ ./prog1.sh -i 1.txt -o 2.txt -p hello -C -n
[vsmokochunina@10 ~]$ cat 2.txt
1:Hello
2:HELLO
3:hello
[vsmokochunina@10 ~]$ ./prog1.sh -i 1.txt -o 2.txt -p hello -n
./prog1.sh: строка 33: prep: команда не найдена
[vsmokochunina@10 ~]$ emacs
[vsmokochunina@10 ~]$ ./prog1.sh -i 1.txt -o 2.txt -p hello -n
[vsmokochunina@10 ~]$ cat 2.txt
3:hello
[vsmokochunina@10 ~]$ ./prog1.sh -i 1.txt -C -n
Шаблон не найден
[vsmokochunina@10 ~]$ ./prog1.sh -o 2.txt -p hello -C -n
Файл не найден
[vsmokochunina@10 ~]$
```

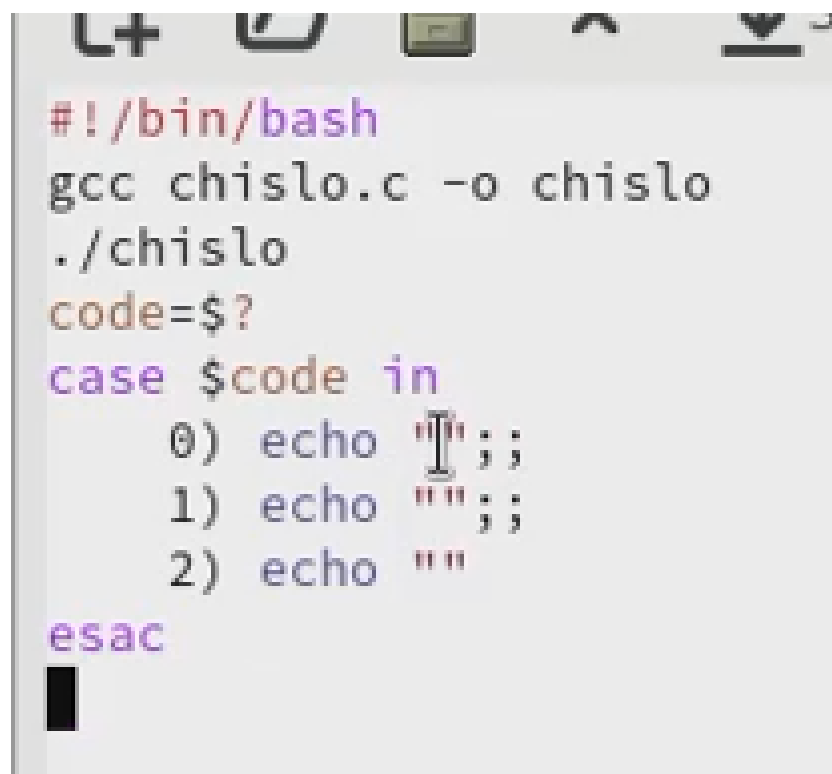
Рис. 3.2: Проверка

3. Написала на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию в о коде завершения в оболочку. Команд- ный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено



```
File Edit Options Buffers Tools C
[Icons] Save
#include <stdio.h>
#include <stdlib.h>
int main()
{
    printf("Введите число\n");
    int a;
    scanf("%d", &a);
    if (a<0) exit(0);
    if (a>0) exit (1);
    if (a==0) exit (2);
    return 0;
}
```

Рис. 3.3: Командный файл на Си



```
#!/bin/bash
gcc chislo.c -o chislo
./chislo
code=$?
case $code in
    0) echo "0";;
    1) echo "1";;
    2) echo "2";;
esac
```

Рис. 3.4: Командный файл

#### 4. Проверила на выполнение

```
[vsmokochunina@10 ~]$ ./chislo.sh
Введите число
0
Число равно 0
[vsmokochunina@10 ~]$ ./chislo.sh
Введите число
3
Число больше 0
[vsmokochunina@10 ~]$ ./chislo.sh
Введите число
-1
Число меньше 0
[vsmokochunina@10 ~]$
```

Рис. 3.5: Проверка

5. Написала командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до ☐ (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).

```
#!/bin/bash
opt=$1;
format=$2;
number=$3;
function Files()
{
    for (( i=1; i<=$number; i++ )) do
        file=$(echo $format | tr '#' "$i")
        if [ $opt == "-r" ]
        then
            rm -f $file
        elif [ $opt == "-c" ]
        then
            touch $file
        fi
    done
}
```

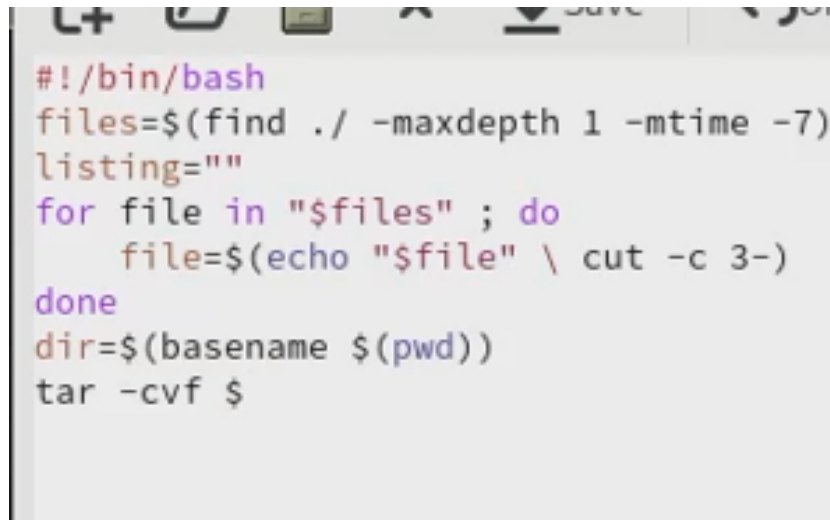
Рис. 3.6: Командный файл

## 6. Проверка на выполнение

```
[vsmokochunina@10 ~]$ ./prog3.sh -c abc#.txt 3
[vsmokochunina@10 ~]$ ls
1.txt      backup4.sh~  file1.doc   ssh
2.txt      backup.sh~   file2.doc   ssh.pub
abc1.txt   backup.sh~   file.pdf    Vmokochunina.github.io
abc2.txt   bin          file.txt    work
abc3.txt   blog         lab07.sh    Видео
abc.txt    chislo       lab07.sh~   Документы
backup     chislo.c     lab101.sh   Загрузки
backup2.sh chislo.c~    lab101.sh~  Изображения
backup2.sh~ chislo.sh    prog1.sh~   Музыка
backup3.sh chislo.sh~   prog1.sh~   Общедоступные
backup3.sh~ conf.txt     prog3.sh~   'Рабочий стол'
backup4.sh feathers     prog3.sh~   Шаблоны
[vsmokochunina@10 ~]$ #./prog3.sh -r abc#.txt 3
[vsmokochunina@10 ~]$ ./prog3.sh -r abc#.txt 3
[vsmokochunina@10 ~]$ ls
```

Рис. 3.7: Проверка

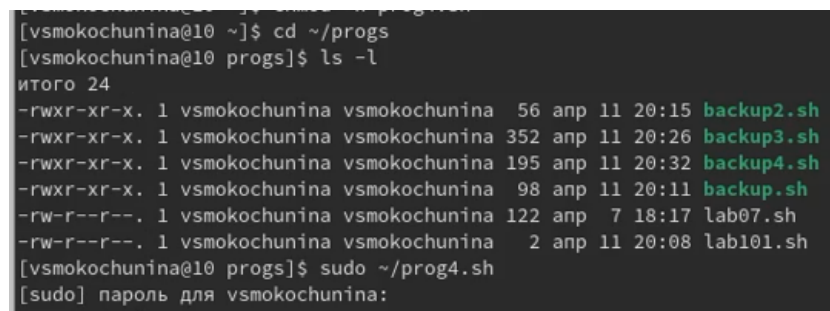
7. Написала командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировала его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find).



```
#!/bin/bash
files=$(find ./ -maxdepth 1 -mtime -7)
listing=""
for file in "$files" ; do
    file=$(echo "$file" \ cut -c 3-)
done
dir=$(basename $(pwd))
tar -cvf $
```

Рис. 3.8: Командный файл

8. Проверила на выполнение



```
[vsmokochunina@10 ~]$ cd ~/progs
[vsmokochunina@10 progs]$ ls -l
итого 24
-rwxr-xr-x. 1 vsmokochunina vsmokochunina 56 anp 11 20:15 backup2.sh
-rwxr-xr-x. 1 vsmokochunina vsmokochunina 352 anp 11 20:26 backup3.sh
-rwxr-xr-x. 1 vsmokochunina vsmokochunina 195 anp 11 20:32 backup4.sh
-rwxr-xr-x. 1 vsmokochunina vsmokochunina 98 anp 11 20:11 backup.sh
-rw-r--r--. 1 vsmokochunina vsmokochunina 122 anp 7 18:17 lab07.sh
-rw-r--r--. 1 vsmokochunina vsmokochunina 2 anp 11 20:08 lab101.sh
[vsmokochunina@10 progs]$ sudo ~/prog4.sh
[sudo] пароль для vsmokochunina:
```

Рис. 3.9: Проверка

## 4 Выводы

Изучила основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.