

# Лабораторная работа № 11.

---

Махорин Иван Сергеевич

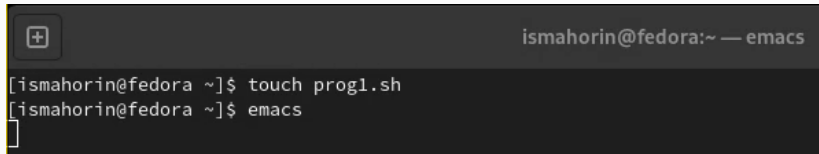
2022, 25 мая

RUDN, Москва, Россия

Программирование в командном  
процессоре ОС UNIX. Ветвления и  
циклы.

---

## Создание первого файла для скрипта

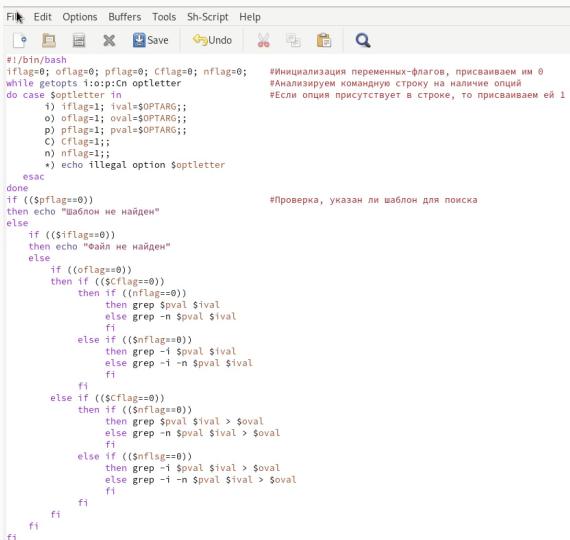
A terminal window with a dark background. The title bar at the top shows a square icon with a plus sign on the left and the text "ismahorin@fedora:~ — emacs" on the right. The terminal content shows two lines of text: "[ismahorin@fedora ~]\$ touch prog1.sh" followed by "[ismahorin@fedora ~]\$ emacs". Below the second line, there is a small white cursor icon.

```
ismahorin@fedora:~ — emacs
[ismahorin@fedora ~]$ touch prog1.sh
[ismahorin@fedora ~]$ emacs
█
```

Figure 1: Создание нового файла для скрипта

Написать скрипт, который анализирует командную строку с определёнными ключами, а затем ищет в указанном файле нужные строки, определяемые ключом -p.

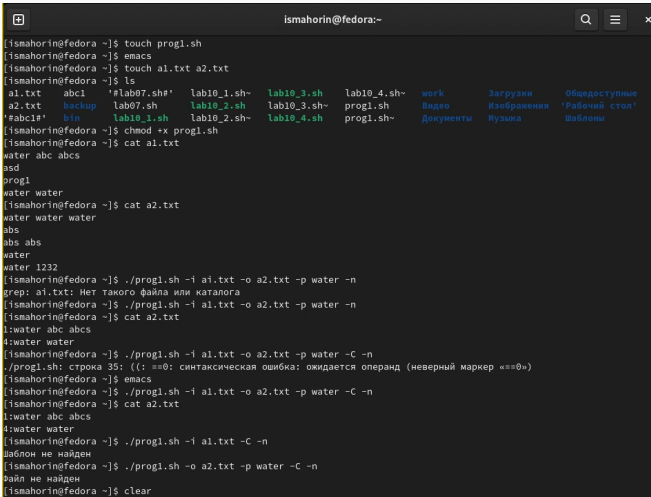
# Написание первого скрипта



```
#!/bin/bash
iflag=0; oflag=0; pflag=0; cflag=0; nflag=0;      #Инициализация переменных-флагов, присваиваем им 0
while getopts 1:o:p:cn optletter                 #Анализируем командную строку на наличие опций
do case $optletter in                             #Если опция присутствует в строке, то присваиваем ей 1
  i) iflag=1; ival=$OPTARG;;
  o) oflag=1; oval=$OPTARG;;
  p) pflag=1; pval=$OPTARG;;
  C) cflag=1;;
  n) nflag=1;;
  *) echo illegal option $optletter
esac
done
if (($pflag==0))                                  #Проверка, указан ли шаблон для поиска
then echo "Шаблон не найден"
else
  if (($iflag==0))
  then echo "Файл не найден"
  else
    if ((oflag==0))
    then if (($cflag==0))
        then if ((nflag==0))
            then grep $pval $ival
            else grep -n $pval $ival
            fi
        else if (($nflag==0))
            then grep -i $pval $ival
            else grep -i -n $pval $ival
            fi
        fi
    else if (($cflag==0))
        then if (($nflag==0))
            then grep $pval $ival > $oval
            else grep -n $pval $ival > $oval
            fi
        else if (($nflag==0))
            then grep -i $pval $ival > $oval
            else grep -i -n $pval $ival > $oval
            fi
        fi
    fi
  fi
fi
```

Figure 2: Написание первого скрипта

# Запуск и проверка выполнения первого скрипта



```
ismahorin@fedora:~  
[ismahorin@fedora ~]$ touch prog1.sh  
[ismahorin@fedora ~]$ emacs  
[ismahorin@fedora ~]$ touch a1.txt a2.txt  
[ismahorin@fedora ~]$ ls  
a1.txt  abc1  '#lab07.sh#'  lab10_1.sh~  lab10_3.sh  lab10_4.sh~  work  Загрузки  Общедоступные  
a2.txt  backup  lab07.sh      lab10_2.sh  lab10_3.sh~  prog1.sh  Видео  Изображения  'Рабочий стол'  
'#abc1#' bin    lab10_1.sh  lab10_2.sh~  lab10_4.sh  prog1.sh~  Документы  Музыка  Шаблоны  
[ismahorin@fedora ~]$ chmod +x prog1.sh  
[ismahorin@fedora ~]$ cat a1.txt  
water abc abcs  
asd  
prog1  
water water  
[ismahorin@fedora ~]$ cat a2.txt  
water water water  
abs  
abs abs  
water  
water 1232  
[ismahorin@fedora ~]$ ./prog1.sh -i a1.txt -o a2.txt -p water -n  
grep: a1.txt: Нет такого файла или каталога  
[ismahorin@fedora ~]$ ./prog1.sh -i a1.txt -o a2.txt -p water -n  
[ismahorin@fedora ~]$ cat a2.txt  
1:water abc abcs  
4:water water  
[ismahorin@fedora ~]$ ./prog1.sh -i a1.txt -o a2.txt -p water -C -n  
./prog1.sh: строка 35: ((: ==0: синтаксическая ошибка: ожидается операнд (неверный маркер «==0»)  
[ismahorin@fedora ~]$ emacs  
[ismahorin@fedora ~]$ ./prog1.sh -i a1.txt -o a2.txt -p water -C -n  
[ismahorin@fedora ~]$ cat a2.txt  
1:water abc abcs  
4:water water  
[ismahorin@fedora ~]$ ./prog1.sh -i a1.txt -C -n  
шаблон не найден  
[ismahorin@fedora ~]$ ./prog1.sh -o a2.txt -p water -C -n  
файл не найден  
[ismahorin@fedora ~]$ clear
```

Figure 3: Право на выполнение, запуск файла и проверка

## Создание файлов для второго задания

A terminal window with a dark background. The title bar at the top shows a window icon on the left and the text "ismahorin@fedora:~ — emacs" on the right. The terminal content shows three lines of commands and their prompts: the first line is "[ismahorin@fedora ~]\$ touch chslo.c", the second line is "[ismahorin@fedora ~]\$ touch chslo.sh", and the third line is "[ismahorin@fedora ~]\$ emacs".

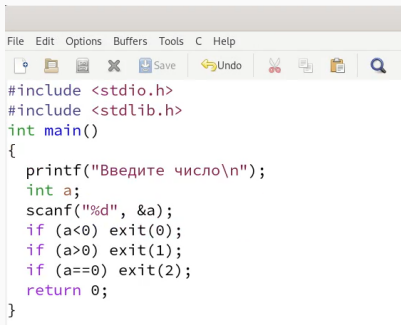
```
ismahorin@fedora:~ — emacs
[ismahorin@fedora ~]$ touch chslo.c
[ismahorin@fedora ~]$ touch chslo.sh
[ismahorin@fedora ~]$ emacs
```

Figure 4: Создание двух файлов и открытие emacs

Написать 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.



# Написание программы на языке программирования Си

A screenshot of a text editor window showing a C program. The window has a menu bar with 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'C', and 'Help'. Below the menu bar is a toolbar with icons for opening files, saving, undo, redo, and search. The code is written in a monospaced font with syntax highlighting: preprocessor directives are in blue, function names in green, variables in red, and control flow keywords in blue. The program includes `<stdio.h>` and `<stdlib.h>`, defines a `main` function, prints a prompt in Russian, reads an integer, and exits with different codes based on the input's sign.

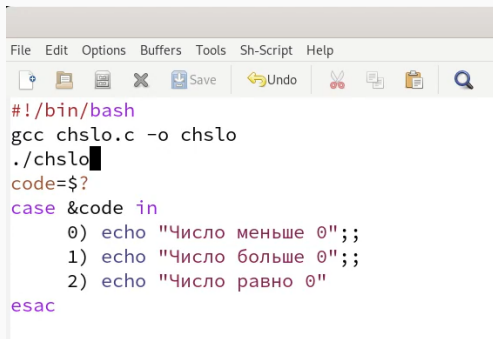
```
File Edit Options Buffers Tools C Help
[Icons: Open, Save, Undo, Redo, Find]

#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;
}
```

Figure 5: Написание программы на языке Си

## Написание скрипта для второго задания

Написать командный файл, который должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

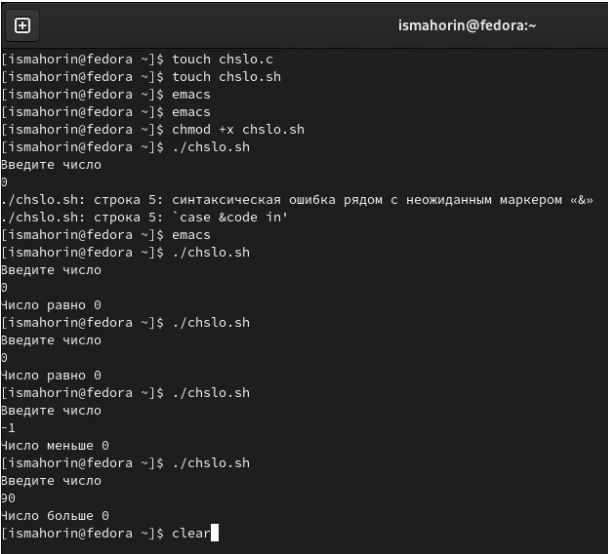


```
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Undo, Cut, Copy, Paste, Find]

#!/bin/bash
gcc chslo.c -o chslo
./chslo
code=$?
case &code in
    0) echo "Число меньше 0";;
    1) echo "Число больше 0";;
    2) echo "Число равно 0"
esac
```

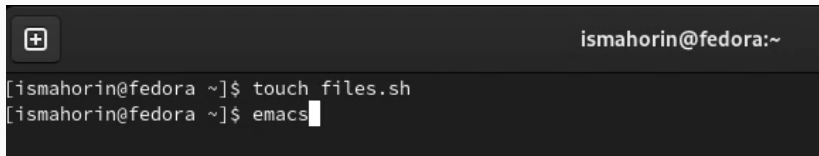
Figure 6: Написание командного файла для второго задания

## Право на выполнение и последующая проверка



```
ismahorin@fedora:~  
[ismahorin@fedora ~]$ touch chslo.c  
[ismahorin@fedora ~]$ touch chslo.sh  
[ismahorin@fedora ~]$ emacs  
[ismahorin@fedora ~]$ emacs  
[ismahorin@fedora ~]$ chmod +x chslo.sh  
[ismahorin@fedora ~]$ ./chslo.sh  
Введите число  
0  
./chslo.sh: строка 5: синтаксическая ошибка рядом с неожиданным маркером «&»  
./chslo.sh: строка 5: `case &code in'  
[ismahorin@fedora ~]$ emacs  
[ismahorin@fedora ~]$ ./chslo.sh  
Введите число  
0  
Число равно 0  
[ismahorin@fedora ~]$ ./chslo.sh  
Введите число  
0  
Число равно 0  
[ismahorin@fedora ~]$ ./chslo.sh  
Введите число  
-1  
Число меньше 0  
[ismahorin@fedora ~]$ ./chslo.sh  
Введите число  
90  
Число больше 0  
[ismahorin@fedora ~]$ clear
```

Figure 7: Право на выполнение, запуск файла



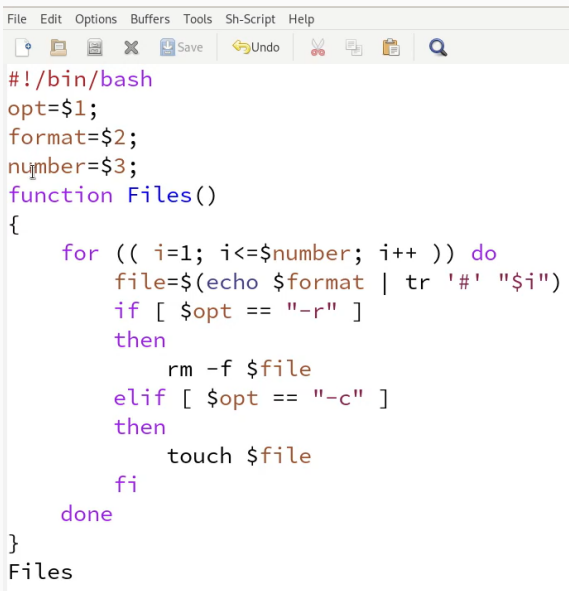
A terminal window with a dark background. The title bar shows a window icon on the left and the text "ismahorin@fedora:~" on the right. The terminal content shows two lines of text: the first line is "[ismahorin@fedora ~]\$ touch files.sh" and the second line is "[ismahorin@fedora ~]\$ emacs" followed by a white cursor block.

```
ismahorin@fedora:~  
[ismahorin@fedora ~]$ touch files.sh  
[ismahorin@fedora ~]$ emacs
```

Figure 8: Создание третьего файла

Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (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).

## Написание третьего скрипта

A screenshot of a text editor window with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Help) and a toolbar with icons for file operations. The editor contains a shell script with the following content:

```
#!/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
}
Files
```

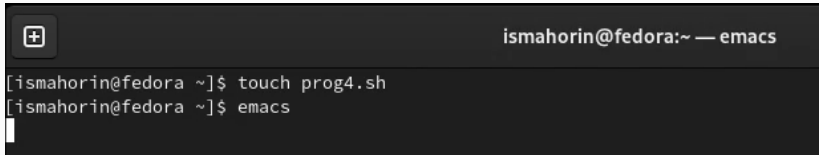
Figure 9: Написание третьего скрипта

# Право на выполнение и запуск файла

```
ismahorin@fedora:~$ touch files.sh
ismahorin@fedora:~$ emacs
ismahorin@fedora:~$ chmod +x files.sh
ismahorin@fedora:~$ ls
al.txt  backup  chslo.c~  files.sh~  lab10_1.sh~  lab10_3.sh~  prog1.sh~  Загрузки  'Рабочий стол'
a2.txt  bin     chslo.sh  '#lab07.sh#'  lab10_2.sh  lab10_4.sh  work      Изображения  Шаблоны
#abc1#  chslo  chslo.sh~  lab07.sh  lab10_2.sh~  lab10_4.sh~  Видео     Музыка
abc1    chslo.c  files.sh  lab10_1.sh  lab10_3.sh  prog1.sh  Документы  Общедоступные
ismahorin@fedora:~$ ./files.sh -c abc#.txt 3
ismahorin@fedora:~$ ls
al.txt  abc2.txt  chslo.c  files.sh~  lab10_2.sh  lab10_4.sh~  Документы  'Рабочий стол'
a2.txt  abc3.txt  chslo.c~  '#lab07.sh#'  lab10_2.sh~  prog1.sh  Загрузки  Шаблоны
#abc1#  backup  chslo.sh  lab07.sh  lab10_3.sh  prog1.sh~  Изображения
abc1    bin     chslo.sh~  lab10_1.sh  lab10_3.sh~  work      Музыка
abc1.txt  chslo  files.sh  lab10_1.sh~  lab10_4.sh  Видео     Общедоступные
ismahorin@fedora:~$ ./files.sh -r abc#.txt 3
ismahorin@fedora:~$ ls
al.txt  backup  chslo.c~  files.sh~  lab10_1.sh~  lab10_3.sh~  prog1.sh~  Загрузки  'Рабочий стол'
a2.txt  bin     chslo.sh  '#lab07.sh#'  lab10_2.sh  lab10_4.sh  work      Изображения  Шаблоны
#abc1#  chslo  chslo.sh~  lab07.sh  lab10_2.sh~  lab10_4.sh~  Видео     Музыка
abc1    chslo.c  files.sh  lab10_1.sh  lab10_3.sh  prog1.sh  Документы  Общедоступные
ismahorin@fedora:~$ clear
```

Figure 10: Право на выполнение, запуск файла

## Создание файла для четвёртого задания



```
ismahorin@fedora:~ — emacs  
[ismahorin@fedora ~]$ touch prog4.sh  
[ismahorin@fedora ~]$ emacs  
|
```

Figure 11: Создание четвёртого файла



## Написание четвёртого скрипта

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

A screenshot of a text editor window with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Help) and a toolbar with icons for file operations. The editor contains a shell script that uses the `find` command to locate files modified within the last 7 days and then uses `tar` to archive them.

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

Figure 12: Написание четвёртого скрипта

# Право на выполнение и запуск файла для нужного каталога

```
ismahorin@fedora:~/Catalog1

[ismahorin@fedora ~]$ touch prog4.sh
[ismahorin@fedora ~]$ emacs
[ismahorin@fedora ~]$ chmod +x prog4.sh
[ismahorin@fedora ~]$ mkdir Catalog1
[ismahorin@fedora ~]$ ls
a1.txt  bin      chslo.sh  lab07.sh  lab10_3.sh  prog1.sh~  Документы  'Рабочий стол'
a2.txt  Catalog1 chslo.sh~ lab10_1.sh lab10_3.sh~ prog4.sh   Загрузки   Шаблоны
'#abc1#' chslo    files.sh  lab10_1.sh~ lab10_4.sh  prog4.sh~  Изображения
abc1    chslo.c  files.sh~ lab10_2.sh  lab10_4.sh~ work       Музыка
backup  chslo.c~ '#lab07.sh#' lab10_2.sh~ prog1.sh   Видео      Общедоступные

[ismahorin@fedora ~]$ cd ~/Catalog1
[ismahorin@fedora Catalog1]$ ls
[ismahorin@fedora Catalog1]$ ls
a1.txt a2.txt abc1 chslo.c chslo.sh files.sh prog1.sh
[ismahorin@fedora Catalog1]$ ls -l
102216 a1.txt 102217 a2.txt 102218 abc1 102219 chslo.c 102220 chslo.sh 102221 files.sh 102222 prog1.sh
[ismahorin@fedora Catalog1]$ sudo ~/prog4.sh
[sudo] пароль для ismahorin:
a1.txt
a2.txt
chslo.c
chslo.sh
files.sh
prog1.sh
[ismahorin@fedora Catalog1]$ tar -tf Catalog1.tar
a1.txt
a2.txt
chslo.c
chslo.sh
files.sh
prog1.sh
[ismahorin@fedora Catalog1]$
[ismahorin@fedora Catalog1]$
```

Figure 13: Право на выполнение, запуск файла для каталога Catalog1

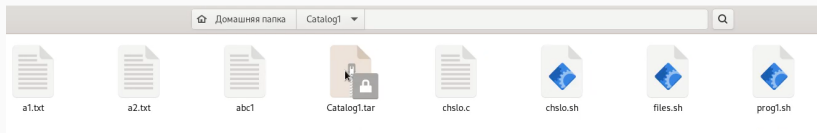


Figure 14: Проверка

## Выводы

---

В ходе выполнения лабораторной работы мы изучили основы программирования в оболочке ОС UNIX и научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Спасибо за внимание!