

Программирование в командном процессоре ОС UNIX.

Ветвления и циклы.

Матвеева Анастасия Сергеевна НПМбд-02-20¹

26 мая, 2021, Москва, Россия

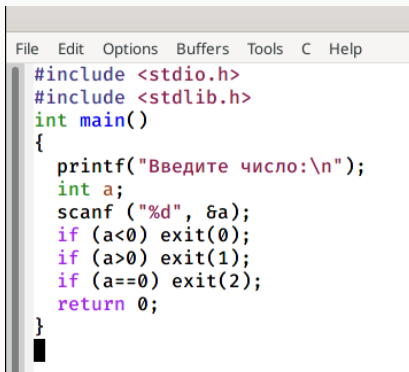
¹RUDN University, Moscow, Russian Federation

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

1. Познакомиться с логическими управляющими конструкцией и циклов.
2. В ходе работы написать 4 командных файла.

Выполнение лабораторной работы

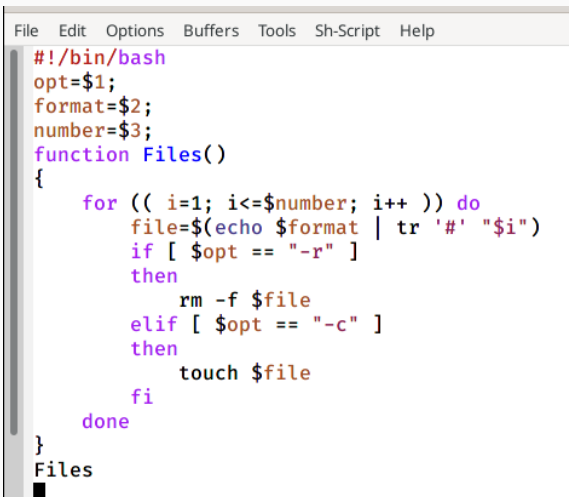
Предварительно для командных файлов создаем файл и открываем emacs. Первый скрипт анализирует командную строку с ключами, используя команду `getopts` `grep`. Второй командный файл должен вызывать предварительно написанную на языке Си программу, которая выводит число и определяет его равенство или неравенство с нулем. (рис. 1)



```
#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;
}
```

Выполнение лабораторной работы

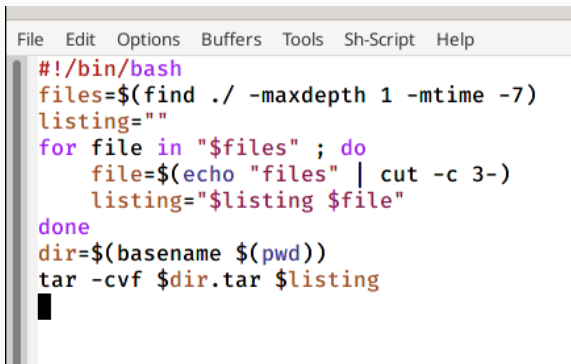
Третий командный файл создает указанное число файлов, пронумерованных от 1 до n. Также он умеет удалять все созданные им файлы. (рис. 2)

A screenshot of a terminal window with a menu bar at the top containing 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The terminal displays 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
```

Выполнение лабораторной работы

Четвертый командный файл с помощью команды tar запаковывает в архив все файлы в указанной директории. Он также модифицирован таким образом, чтобы запаковывались только те файлы, которые были изменены менее неделю тому назад. (рис. 3)

A screenshot of a terminal window with a menu bar at the top containing 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The terminal displays a shell script with syntax highlighting. The script starts with a shebang line, then uses 'find' to locate files, 'echo' and 'cut' to format the file list, and finally 'tar' to create an archive. The script ends with a cursor on a new line.

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

Figure 3: Четвертый скрипт

В ходе выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX и научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.