

# Лабораторная работа № 11

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

---

Шулуужук Айраана Вячеславовна НПИбд-02-22

4 апреля 2023

Российский университет дружбы народов, Москва, Россия

- Шулуужук Айраана Вячеславовна
- НПИбд-02-22
- 1132221890
- Российский университет дружбы народов

## Цели и задачи

---

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

## Выполнение лабораторной работы

---

## Выполнение лабораторной работы

Используя команды `getopts` `grep`, напомним командный файл, который анализирует командную строку с ключами

```
while getopts "i:o:p:c:n" opt
do
case $opt in
i)inputfile="$OPTARG";;
o)outputfile="$OPTARG";;
p)shablon="OPTARG";;
c)registr="";;
n)number="";;
esac
done

grep -n "$shablon" "$inputfile" > "$outputfile"
```

Рис. 1: скрипт 1

Скомпилируем данный файл и проверим его работу

```
./lab11_1 -i new.txt -o output.txt -p h -c -n  
[avshuluuzhuk@fedora lab11]$ vi lab11_1  
[avshuluuzhuk@fedora lab11]$ ./lab11_1 -i new.txt -o output.txt -p h -c -n  
[avshuluuzhuk@fedora lab11]$ ls  
lab11_1 new.txt output.txt  
[avshuluuzhuk@fedora lab11]$
```

Рис. 2: результат работы командного файла

## Выполнение лабораторной работы

Напишем на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю

```
#include <iostream>
using namespace std;

int main(int argument, char *arg[]){
    if (atoi(arg[1]) > 0){
        exit(1);
    }
    else if (atoi(arg[1]) == 0) {
        exit(2);
    }
    else {
        exit(3);
    }
    return 0;
}
```

Рис. 3: программа сравнения чисел на С



## Выполнение лабораторной работы

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

```
#!/bin/bash

CC=g++
EXEC=program
SRC=program.cpp

if [ "$SRC" -nt "$EXEC" ]
then
echo "Rebuilding $EXEC ....."
$CC -o $EXEC $SRC
fi

./$EXEC $1
ec=$?
if [ "$ec" == "1" ]
then
echo "argument > 0"
fi
if [ "$ec" == "2" ]
then
echo "argument = 0"
fi
if [ "$ec" == "3" ]
then
echo "argument < 0"
fi
~
```

Рис. 4: скрипт 2

Проверим работу командного файла, используя число для сравнения в качестве аргумента

```
[avshuluuzhuk@fedora lab11]$ ./lab11_2 3
Rebuilding program .....
argument > 0
[avshuluuzhuk@fedora lab11]$ ./lab11_2 0
argument = 0
[avshuluuzhuk@fedora lab11]$ ./lab11_2 -8
argument < 0
```

Рис. 5: результат запуска скрипта 2

Создаем новый файл lab11\_3 для скрипта 3. Напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до n и удаляющий все созданные им файлы

```
#!/bin/bash
while getopts c:r opt
do
case $opt in
    c)n="$OPTARG"; for i in $(seq 1 $n); do touch "$i.tmp";done;;
    r)for i in $(find -name "*.tmp"); do rm $i; done;;
    esac
done
```

Рис. 6: скрипт lab11\_3

Запустим этот файл и создадим файл tmp, а также сразу удалим этот файл

```
[avshuluuzhuk@fedora lab11]$ ./lab11_3 -c 1
[avshuluuzhuk@fedora lab11]$ ls
1.tmp  lab11_1  lab11_2  lab11_3  new.txt  output.txt  program  program.cpp
[avshuluuzhuk@fedora lab11]$ vi lab11_3
[avshuluuzhuk@fedora lab11]$ ./lab11_3 -r
[avshuluuzhuk@fedora lab11]$ ls
lab11_1  lab11_2  lab11_3  new.txt  output.txt  program  program.cpp  prog
[avshuluuzhuk@fedora lab11]$
```

Рис. 7: результат запуска скрипта 3

## Выполнение лабораторной работы

В файле lab11\_4 напишем скрипт, который с помощью команды tar запаковывает в архив все файлы в указанной директории.

```
[#!/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
```

Рис. 8: скрипт lab11\_4

Запустим файл и запакуем архив с файлами каталога lab11

```
[avshuluuzhuk@fedora lab11]$ ./lab11_4 lab11
lab11_1
output.txt
lab11_2
program.cpp~
program.cpp
program
lab11_3
lab11_4
tar: lab11.tar: файл является архивом; не сброшен
[avshuluuzhuk@fedora lab11]$ ls
lab11_1  lab11_3  lab11.tar  output.txt  program.cpp
lab11_2  lab11_4  new.txt   program     program.cpp~
[avshuluuzhuk@fedora lab11]$
```

Рис. 9: результат запуска скрипта 4

## Выводы

---

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