

# Презентация по лабораторной работе №11

Программирование в командном процессоре ОС UNIX. Ветвления и циклы.

---

Галацан Николай

Российский университет дружбы народов, Москва, Россия

- Галацан Николай
- 1032225763
- уч. группа: НПИбд-01-22
- Факультет физико-математических и естественных наук
- Российский университет дружбы народов

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

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами:
  - `-i inputfile` — прочитать данные из указанного файла;
  - `-o outputfile` — вывести данные в указанный файл;
  - `-p шаблон` — указать шаблон для поиска;
  - `-C` — различать большие и малые буквы;
  - `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-p`.

Открываю `emacs`. Создаю файл `lab11_1.sh`, набираю текст программы.

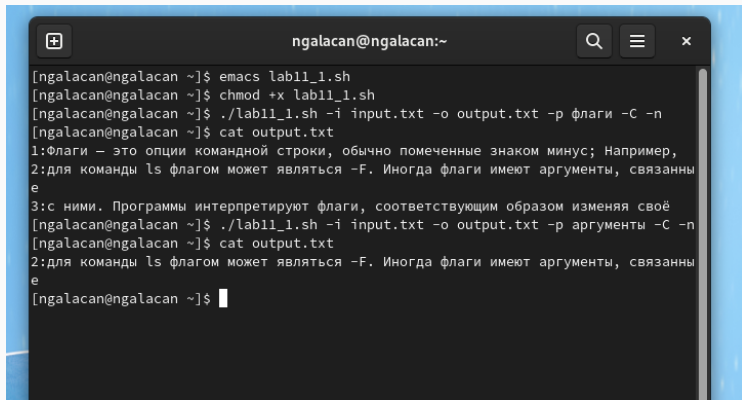
## Выполнение лабораторной работы

*Листинг программы №1:*

```
#!/bin/bash
iflag=0; oflag=0; pflag=0; Cflag=0; nflag=0;
while getopts i:o:p:Cn optletter
do case $optletter in
    i) iflag=1; ival=$OPTARG;;
    o) oflag=1; oval=$OPTARG;;
    p) pflag=1; pval=$OPTARG;;
    C) Cflag=1;;
    n) nflag=1;;
    *) echo "Недопустимая опция" $optletter
esac
done
```

## Выполнение лабораторной работы

Даю файлу право на исполнение и запускаю программу.



```
ngalacan@ngalacan:~  
[ngalacan@ngalacan ~]$ emacs lab11_1.sh  
[ngalacan@ngalacan ~]$ chmod +x lab11_1.sh  
[ngalacan@ngalacan ~]$ ./lab11_1.sh -i input.txt -o output.txt -p флаги -C -n  
[ngalacan@ngalacan ~]$ cat output.txt  
1:Флаги – это опции командной строки, обычно помеченные знаком минус; Например,  
2:для команды ls флагом может являться -F. Иногда флаги имеют аргументы, связанны  
е  
3:с ними. Программы интерпретируют флаги, соответствующим образом изменяя своё  
[ngalacan@ngalacan ~]$ ./lab11_1.sh -i input.txt -o output.txt -p аргументы -C -n  
[ngalacan@ngalacan ~]$ cat output.txt  
2:для команды ls флагом может являться -F. Иногда флаги имеют аргументы, связанны  
е  
[ngalacan@ngalacan ~]$
```

Рис. 1: Запуск программы №1

2. 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. Команд- ный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

Создаю файл для программы на С `lab11_2_prog.c`, набираю текст программы.

*Листинг программы на C:*

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Введите число: ");
    int n;
    scanf("%d", &n);
    if (n<0) exit(1);
    if (n>0) exit(2);
    if (n==0) exit(3);
    return 0;
}
```



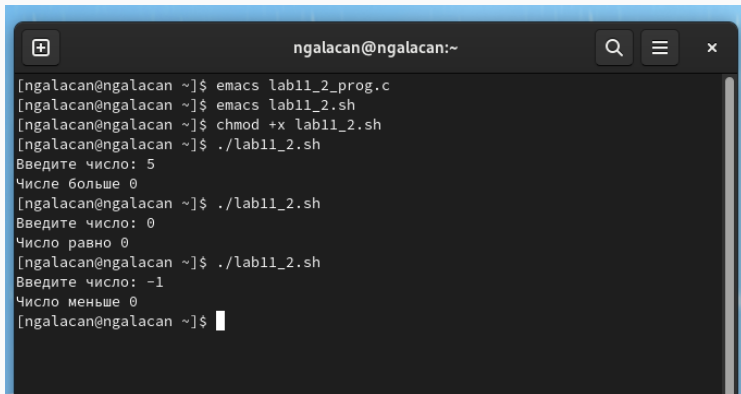
Создаю файл `lab11_2.sh`, набираю текст программы.

*Листинг программы №2:*

```
#!/bin/bash
gcc lab11_2_prog.c -o lab11_2_prog
./lab11_2_prog
exc=$?
case $exc in
    1) echo "Число меньше 0";;
    2) echo "Числе больше 0";;
    3) echo "Число равно 0";;
esac
```

## Выполнение лабораторной работы

Даю файлу право на исполнение и запускаю.

A terminal window titled 'ngalacan@ngalacan:~' with search, menu, and close buttons. It shows a series of commands and their outputs. The user runs 'emacs lab11\_2\_prog.c', 'emacs lab11\_2.sh', and 'chmod +x lab11\_2.sh'. Then they run './lab11\_2.sh' three times, each time entering a number (5, 0, -1) and receiving a message about whether the number is greater, equal, or less than 0.

```
[ngalacan@ngalacan ~]$ emacs lab11_2_prog.c
[ngalacan@ngalacan ~]$ emacs lab11_2.sh
[ngalacan@ngalacan ~]$ chmod +x lab11_2.sh
[ngalacan@ngalacan ~]$ ./lab11_2.sh
Введите число: 5
Числе больше 0
[ngalacan@ngalacan ~]$ ./lab11_2.sh
Введите число: 0
Число равно 0
[ngalacan@ngalacan ~]$ ./lab11_2.sh
Введите число: -1
Число меньше 0
[ngalacan@ngalacan ~]$
```

Рис. 2: Запуск программы №2

3. 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).

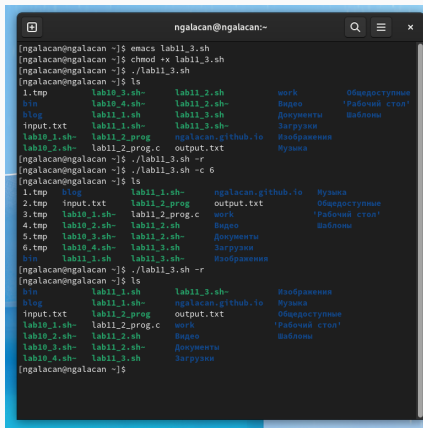
Создаю файл `lab11_3.sh`, набираю текст программы.

*Листинг программы №3:*

```
#!/bin/bash
while getopts c:r optletter
do case $optletter in
        c) cflag=1; cval=$OPTARG;;
        r) rflag=1;;
        esac
done
if ((rflag==0))
then for i in $(seq 1 $cval)
do touch "$i.tmp"
done
else for i in $(find -name "*.tmp")
do rm $i.
```

## Выполнение лабораторной работы

Даю файлу право на исполнение и запускаю. В зависимости от введенной опции (с или r) скрипт создает или удаляет файлы с расширением `.tmp`



```
ngalacan@ngalacan:~$ emacs lab11_3.sh
[ngalacan@ngalacan ~]$ chmod +x lab11_3.sh
[ngalacan@ngalacan ~]$ ./lab11_3.sh
[ngalacan@ngalacan ~]$ ls
1.tmp      lab10_3.sh-  lab11_2.sh      work      Общедоступные
bin        lab10_4.sh-  lab11_2.sh-     Видео     'Рабочий стол'
blog       lab11_1.sh   lab11_3.sh      Документы  Шаблоны
input.txt  lab11_1.sh-  lab11_3.sh-     Загрузки
lab10_1.sh- lab11_2_prog ngalacan.github.io  Изображения
lab10_2.sh- lab11_2_prog.c output.txt      Музыка
[ngalacan@ngalacan ~]$ ./lab11_3.sh -r
[ngalacan@ngalacan ~]$ ./lab11_3.sh -c 6
[ngalacan@ngalacan ~]$ ls
1.tmp  blog      lab11_1.sh-  ngalacan.github.io  Музыка
2.tmp  input.txt  lab11_2_prog  output.txt          Общедоступные
3.tmp  lab10_1.sh- lab11_2_prog.c work              'Рабочий стол'
4.tmp  lab10_2.sh- lab11_2.sh    Видео            Шаблоны
5.tmp  lab10_3.sh- lab11_2.sh-   Документы
6.tmp  lab10_4.sh- lab11_3.sh    Загрузки
bin    lab11_1.sh   lab11_3.sh-   Изображения
[ngalacan@ngalacan ~]$ ./lab11_3.sh -r
[ngalacan@ngalacan ~]$ ls
bin      lab11_1.sh   lab11_3.sh-  Изображения
blog     lab11_1.sh-  ngalacan.github.io  Музыка
input.txt lab11_2_prog  output.txt    Общедоступные
lab10_1.sh- lab11_2_prog.c work          'Рабочий стол'
lab10_2.sh- lab11_2.sh   Видео        Шаблоны
lab10_3.sh- lab11_2.sh-  Документы
lab10_4.sh- lab11_3.sh   Загрузки
[ngalacan@ngalacan ~]$
```

Рис. 3: Запуск программы №3

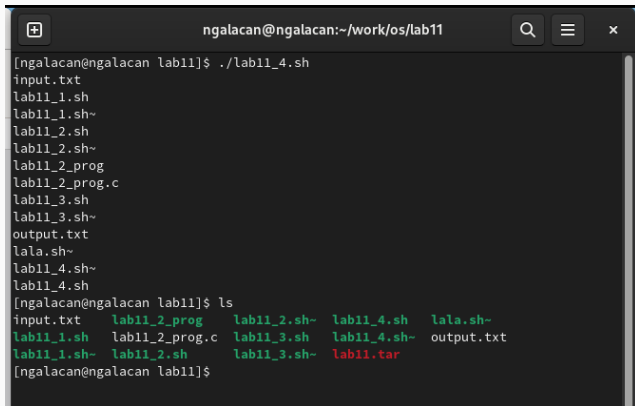
4. Написать командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`)

Создаю файл `lab11_4.sh`, набираю текст программы.

*Листинг программы №4:*

```
#!/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
```

Даю файлу право на исполнение и запускаю

A terminal window titled 'ngalacan@ngalacan:~/work/os/lab11' with search, menu, and close icons. The prompt is '[ngalacan@ngalacan lab11]\$'. The user enters './lab11\_4.sh', followed by a list of files: 'input.txt', 'lab11\_1.sh', 'lab11\_1.sh~', 'lab11\_2.sh', 'lab11\_2.sh~', 'lab11\_2\_prog', 'lab11\_2\_prog.c', 'lab11\_3.sh', 'lab11\_3.sh~', 'output.txt', 'lala.sh~', 'lab11\_4.sh~', and 'lab11\_4.sh'. Then the user enters 'ls', and the terminal displays a color-coded listing of the files.

```
[ngalacan@ngalacan lab11]$ ./lab11_4.sh
input.txt
lab11_1.sh
lab11_1.sh~
lab11_2.sh
lab11_2.sh~
lab11_2_prog
lab11_2_prog.c
lab11_3.sh
lab11_3.sh~
output.txt
lala.sh~
lab11_4.sh~
lab11_4.sh
[ngalacan@ngalacan lab11]$ ls
input.txt  lab11_2_prog  lab11_2.sh~  lab11_4.sh  lala.sh~
lab11_1.sh  lab11_2_prog.c  lab11_3.sh  lab11_4.sh~  output.txt
lab11_1.sh~  lab11_2.sh  lab11_3.sh~  lab11.tar
```

Рис. 4: Запуск программы №4



Изучены основы программирования в оболочке ОС UNIX. Приобретены навыки написания более сложных командных файлов с использованием логических управляющих конструкций и циклов.