

# Лабораторная работа №13

Операционные системы

---

Калашникова Ольга Сергеевна НПИбд-01-23

4 мая 2024

Российский университет дружбы народов, Москва, Россия

Цель данной лабораторной работы - изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Создаю файл с разрешением на исполнение (рис.1).

```
[oskalashnikova@oskalashnikova ~]$ touch l11.sh
[oskalashnikova@oskalashnikova ~]$ chmod +x l11.sh
[oskalashnikova@oskalashnikova ~]$ touch input.txt output.txt
[oskalashnikova@oskalashnikova ~]$ bash l11.sh -p curl -i input.txt -o o
utput.txt -c -n
[oskalashnikova@oskalashnikova ~]$
```

**Рис. 1:** Создание файла

## Выполнение лабораторной работы

Командный файл, с командами getopt и grep, который анализирует командную строку, а затем ищет в указанном файле нужные строки (рис.2).

```
#!/bin/bash

while getopt 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 Illegal option $optletter;;
  esac
done

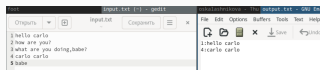
if ! test $cflag
then
  cf=-i
fi

if test $nflag
then
  nf=-n
fi

grep $cf $nf $pval $ival >> $oval
```

Рис. 2: Код программы

Результат работы программы в файле output.txt (рис.3).



**Рис. 3:** Результат работы программы

Создаю исполняемый файл для второй программы, также создаю файл 12.c для программы на Си (рис.4).

```
[oskalashnikova@oskalashnikova ~]$ touch 112.sh  
[oskalashnikova@oskalashnikova ~]$ chmod +x 112.sh  
[oskalashnikova@oskalashnikova ~]$ touch 12.cpp  
[oskalashnikova@oskalashnikova ~]$ bash 112.sh
```

**Рис. 4:** Создание файла

## Выполнение лабораторной работы

Пишу программу на языке Си, 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 (рис.5).

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

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

Командный файл должен вызывать эту программу и, проанализировав с помощью команды \$?, выдать сообщение о том, какое число было введено (рис.6).

```
#!/bin/bash

gcc -o cprog 12.cpp
./cprog
case $? in
0) echo "Число равно нулю";;
1) echo "Число больше нуля";;
2) echo "Число меньше нуля";;
esac
```

**Рис. 6:** Код программы



Программа работает корректно (рис.7).

```
[oskalashnikova@oskalashnikova ~]$ bash 112.sh
Введите число: 34
Число больше нуля
[oskalashnikova@oskalashnikova ~]$ bash 112.sh
Введите число: 0
Число равно нулю
[oskalashnikova@oskalashnikova ~]$ bash 112.sh
Введите число: -100
Число меньше нуля
[oskalashnikova@oskalashnikova ~]$
```

**Рис. 7:** Результат работы программы

Создаю исполняемый файл для третьей программы (рис.8).

```
[oskalashnikova@oskalashnikova ~]$ touch 113.sh  
[oskalashnikova@oskalashnikova ~]$ chmod +x 113.sh  
[oskalashnikova@oskalashnikova ~]$
```

**Рис. 8:** Создание файла

## Выполнение лабораторной работы

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) (рис.9).

```
#!/bin/bash
for((i=1; i<=$*; i++))
do
if test -f "$i".tmp
then rm "$i".tmp
else touch "$i.tmp"
fi
done
```

Рис. 9: Код программы

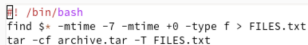
# Выполнение лабораторной работы

Проверяю, что программа создала файлы и удалила их при соответствующих запросах (рис.10).

```
[oskalashnikova@oskalashnikova ~]$ bash 113.sh 4
[oskalashnikova@oskalashnikova ~]$ ls
111.sh  12.cpp~  cprog    my_os    Документы
111.sh~ 1.tmp    feathers output.txt Загрузки
112.sh  2.tmp    file.txt play     Изображения
112.sh~ 3.tmp    input.txt reports  Музыка
113.sh  4.tmp    input.txt~ ski.plases Общедоступные
113.sh~ abcl    may      work     'Рабочий стол'
12.cpp  bin      monthly1 Видео     Шаблоны
[oskalashnikova@oskalashnikova ~]$ bash 113.sh 3
[oskalashnikova@oskalashnikova ~]$ ls
111.sh  12.cpp~  input.txt reports  Музыка
111.sh~ 4.tmp    input.txt~ ski.plases Общедоступные
112.sh  abcl    may      work     'Рабочий стол'
112.sh~ bin      monthly1 Видео     Шаблоны
113.sh  cprog    my_os    Документы
113.sh~ feathers output.txt Загрузки
12.cpp  file.txt play     Изображения
[oskalashnikova@oskalashnikova ~]$
```

Рис. 10: Результат работы программы

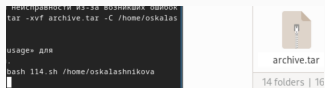
Создаю исполняемый файл для четвертой программы. Это командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`) (рис.11).

A terminal window with a dark background and light text. The prompt is a red icon followed by `/bin/bash`. The first command is `find * -mtime -7 -mtime +0 -type f > FILES.txt` and the second is `tar -cf archive.tar -T FILES.txt`.

```
/bin/bash
find * -mtime -7 -mtime +0 -type f > FILES.txt
tar -cf archive.tar -T FILES.txt
```

**Рис. 11:** Код программы

Проверяю работу программы (рис.12).



**Рис. 12:** Результат работы программы

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