

Лабораторная работа № 13

Операционные системы

Иванов Сергей Владимирович, НПИбд-01-23

29 апреля 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.

Выполнение работы

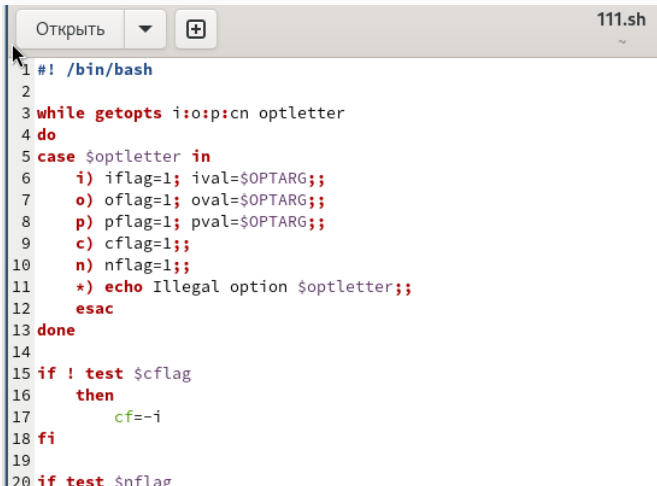
Создаю файл с разрешением на исполнение

```
[svivanov1@svivanov1 ~]$ touch 111.sh  
[svivanov1@svivanov1 ~]$ chmod +x 111.sh  
[svivanov1@svivanov1 ~]$ bash 111.sh -p улит -i input.txt -o output.txt -c -n
```

Рис. 1: Создание файла

Код программы

Командный файл, который анализирует командную строку с ключами а затем ищет в указанном файле нужные строки



```
111.sh
~

1 #!/bin/bash
2
3 while getopts i:o:p:cn optletter
4 do
5     case $optletter in
6         i) iflag=1; ival=$OPTARG;;
7         o) oflag=1; oval=$OPTARG;;
8         p) pflag=1; pval=$OPTARG;;
9         c) cflag=1;;
10        n) nflag=1;;
11        *) echo Illegal option $optletter;;
12    esac
13 done
14
15 if ! test $cflag
16 then
17     cf=-i
18 fi
19
20 if test $nflag
```

Результат работы программы в файле output.txt

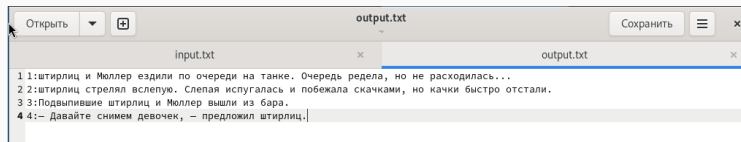


Рис. 3: Результат работы программы

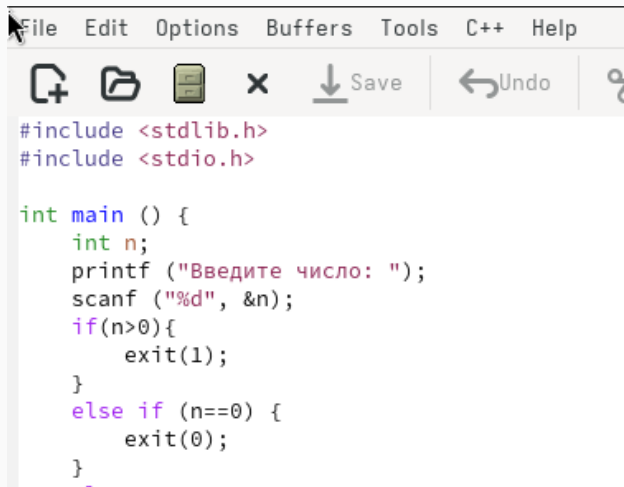
Создаю исполняемый файл для второй программы, также создаю файл 12.cpp для программы на C

```
[svivanov1@svivanov1 ~]$ touch 112.sh  
[svivanov1@svivanov1 ~]$ chmod +x 112.sh  
[svivanov1@svivanov1 ~]$ touch 12.cpp  
[svivanov1@svivanov1 ~]$ bash 112.sh
```

Рис. 4: Создание файла

Код программы на Си

Пишу программу на Си, которая вводит число и определяет, является ли оно больше нуля



```
File Edit Options Buffers Tools C++ Help
[Icons] Save Undo

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

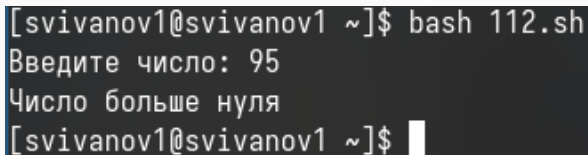
int main () {
    int n;
    printf ("Введите число: ");
    scanf ("%d", &n);
    if(n>0){
        exit(1);
    }
    else if (n==0) {
        exit(0);
    }
}
```


Код программы

Командный файл должен вызывать программу и, проанализировав, выдать сообщение о том, какое число было введено

```
1 #!/bin/bash  
2  
3 gcc -o cprog 12.cpp  
4 ./cprog  
5 case $? in  
6 0) echo "Число равно нулю";;  
7 1) echo "Число больше нуля";;  
8 2) echo "Число меньше нуля";;  
9 esac
```

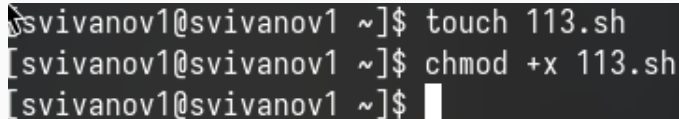
Программа работает корректно

A terminal window with a dark background and light gray text. The text shows a user running a script, entering a number, and receiving a confirmation message.

```
[svivanov1@svivanov1 ~]$ bash 112.sh
Введите число: 95
Число больше нуля
[svivanov1@svivanov1 ~]$
```

Рис. 7: Результат работы программы

Создаю исполняемый файл для третьей программы

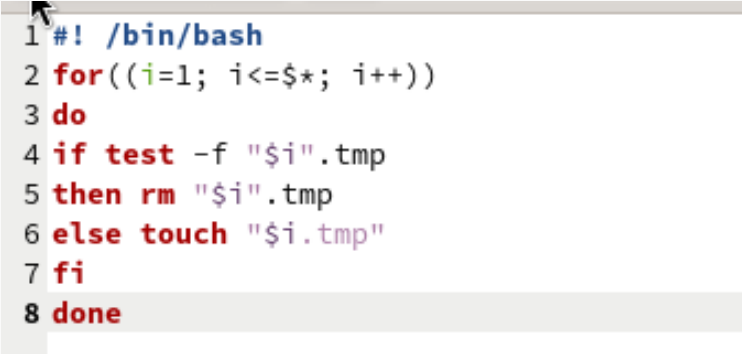
A terminal window with a dark background and light gray text. The prompt is 'svivanov1@svivanov1 ~]\$. The first command is 'touch 113.sh'. The second command is 'chmod +x 113.sh'. The third line shows the prompt again with a white cursor block.

```
svivanov1@svivanov1 ~]$ touch 113.sh  
[svivanov1@svivanov1 ~]$ chmod +x 113.sh  
[svivanov1@svivanov1 ~]$
```

Рис. 8: Создание файла

Код программы

Командный файл, создающий указанное число файлов, пронумерованных последовательно. Этот файл должен уметь удалять все созданные им файлы



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

Рис. 9: Код программы

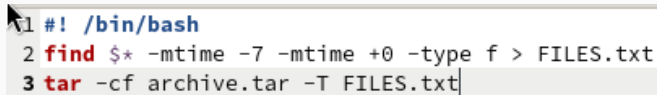
Результат работы программы

Проверяю, что программа создала файлы и удалила их при соответствующих запросах

```
[svivanov1@svivanov1 ~]$ bash 113.sh 4
[svivanov1@svivanov1 ~]$ ls
'##111.sh#'  1.tmp      bin          '##lab07.sh##'  prog1.sh  text.txt  work5      Общедоступные
111.sh       2.tmp      blog         '##lab07.sh#'  prog2.sh  touch     Видео      'Рабочий стол'
112.sh       3.tmp      cprog       '##lab07.sh#~'  prog3.sh  work      Документы  Шаблоны
113.sh       4.tmp      fun         my_os          prog4.sh  work2     Загрузки
12.cpp       australia  git-extended  output.txt     ski.plases work3     Изображения
12.cpp~      backup     input.txt    package.json   spring    work4     Музыка
[svivanov1@svivanov1 ~]$ bash 113.sh 4
[svivanov1@svivanov1 ~]$ ls
'##111.sh#'  australia  git-extended  output.txt     ski.plases  work3     Изображения
111.sh       backup     input.txt    package.json   spring       work4     Музыка
112.sh       bin        '##lab07.sh##'  prog1.sh      text.txt    work5     Общедоступные
113.sh       blog      '##lab07.sh#'  prog2.sh      touch       Видео     'Рабочий стол'
12.cpp       cprog     '##lab07.sh#~'  prog3.sh      work        Документы Шаблоны
12.cpp~      fun       my_os          prog4.sh      work2       Загрузки
[svivanov1@svivanov1 ~]$
```

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.



```
1 #!/bin/bash
2 find $* -mtime -7 -mtime +0 -type f > FILES.txt
3 tar -cf archive.tar -T FILES.txt
```

Рис. 11: Код программы

Результат работы программы

Проверяю работу программы

```
[svivanov1@svivanov1 ~]$ bash 114.sh /home/svivanov1  
tar: Удаляется начальный '/' из имен объектов  
tar: Удаляются начальные '/' из целей жестких ссылок  
[svivanov1@svivanov1 ~]$
```

Рис. 12: Результат работы программы

Вывод

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

<https://esystem.rudn.ru/mod/page/view.php?id=1098933>