

# Лабораторная работа №13

Операционные системы

---

Луангсуваннавонг Сайпхачан

6 мая 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Луангсуваннавонг Сайпхачан
- Студент из группы НКАбд-01-24
- Российский университет дружбы народов
- <https://sayprachanh-lsvnv.github.io>

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

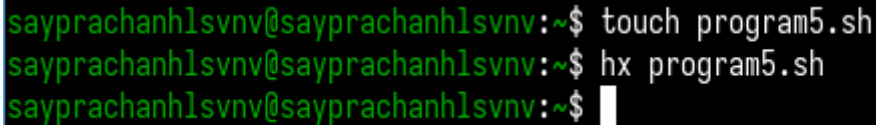
1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: `-i`inputfile — прочитать данные из указанного файла; `-o`outputfile — вывести данные в указанный файл; `-r`шаблон — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-r`.
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. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

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).
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`).

## Выполнение лабораторной работы

---

Я создаю файл program5.sh и открываю его для редактирования в текстовом редакторе. (рис. 1)

A terminal window with a black background and green text. It shows three lines of commands and their execution. The first line is 'touch program5.sh', the second is 'hx program5.sh', and the third is a prompt with a cursor. The terminal text is: sayprachanh1svnv@sayprachanh1svnv:~\$ touch program5.sh  
sayprachanh1svnv@sayprachanh1svnv:~\$ hx program5.sh  
sayprachanh1svnv@sayprachanh1svnv:~\$ █

```
sayprachanh1svnv@sayprachanh1svnv:~$ touch program5.sh
sayprachanh1svnv@sayprachanh1svnv:~$ hx program5.sh
sayprachanh1svnv@sayprachanh1svnv:~$ █
```

Рис. 1: Создание файла

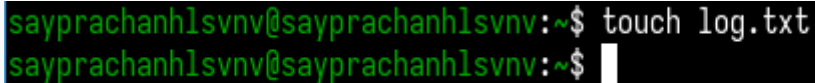


## Выполнение лабораторной работы

Я добавляю код программы, которая будет искать нужные строки в указанном файле, используя команды `getopts` и `grep`. (рис. 2)

```
1  #!/bin/bash
2  while getopts i:o:p:Cn optletter
3  do
4  case $optletter in
5      i) iflag=1; ival=$OPTARG;;
6      o) oflag=1; oval=$OPTARG;;
7      p) pflag=1; pval=$OPTARG;;
8      C) Cflag=1;;
9      n) nflag=1;;
10     *) echo Illegal option $optletter
11     esac
12 done
13
14 if ! test $Cflag
15 then
16     cf=-i
17 fi
18
19 if test $nflag
20 then
21     nf=-n
22 fi
23
24 if test $oflag
25 then
```

Я также создаю текстовый файл log.txt для использования с опцией -o, чтобы сохранять результат работы программы в файл. (рис. 3)

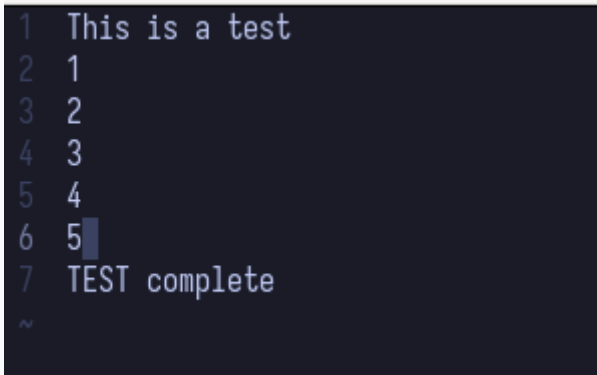
A terminal window with a black background and green text. The prompt is 'sayprachanh1svnv@sayprachanh1svnv:~\$'. The command 'touch log.txt' has been entered and executed. The prompt is now 'sayprachanh1svnv@sayprachanh1svnv:~\$' followed by a white cursor block.

```
sayprachanh1svnv@sayprachanh1svnv:~$ touch log.txt  
sayprachanh1svnv@sayprachanh1svnv:~$
```

Рис. 3: Создание файла

## Выполнение лабораторной работы

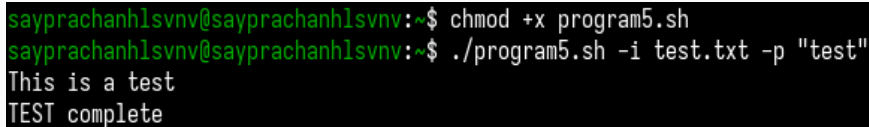
Я создаю тестовый файл test.txt для проверки работы программы. Внутри файла я записываю некоторый текст. (рис. 4)

A screenshot of a terminal window with a dark background and light blue text. It shows a sequence of commands and their outputs. Line 1: 'This is a test'. Line 2: '1'. Line 3: '2'. Line 4: '3'. Line 5: '4'. Line 6: '5' followed by a cursor. Line 7: 'TEST complete'. Line 8: '~'.

```
1 This is a test
2 1
3 2
4 3
5 4
6 5
7 TEST complete
~
```

Рис. 4: Создание файла

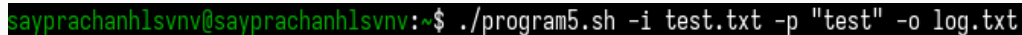
Я даю файлу разрешение на выполнение и запускаю программу. Сначала я запускаю программу и выбираю вывод результата на экран. (рис. 5)

A terminal window with a black background and green text. The prompt is 'sayprachanhlsnv@sayprachanhlsnv:~\$'. The first command is 'chmod +x program5.sh'. The second command is './program5.sh -i test.txt -p "test"'. The output consists of two lines: 'This is a test' and 'TEST complete'.

```
sayprachanhlsnv@sayprachanhlsnv:~$ chmod +x program5.sh
sayprachanhlsnv@sayprachanhlsnv:~$ ./program5.sh -i test.txt -p "test"
This is a test
TEST complete
```

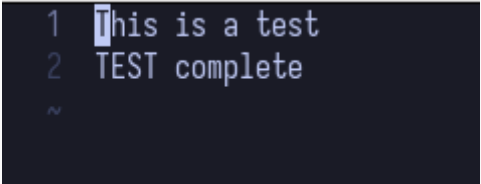
Рис. 5: Запуск программы

Затем я выбираю вывод результата в созданный текстовый файл, добавляя опцию -o, за которой следует имя файла (log.txt). Я проверяю результат команды — в файле log.txt есть строки, значит программа работает корректно. (рис. 6 и рис. 7)

A terminal window with a black background and green text. The prompt is 'sayprachanhlsvnnv@sayprachanhlsvnnv:~\$'. The command entered is './program5.sh -i test.txt -p "test" -o log.txt'.

```
sayprachanhlsvnnv@sayprachanhlsvnnv:~$ ./program5.sh -i test.txt -p "test" -o log.txt
```

Рис. 6: Запуск программы

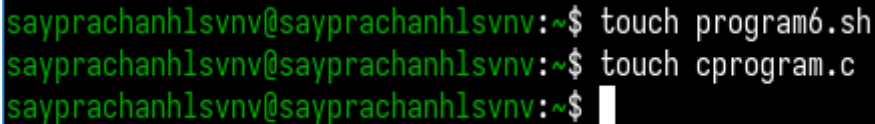


```
1 This is a test
2 TEST complete
~
```

A screenshot of a terminal window with a dark background. It shows two lines of text: "1 This is a test" and "2 TEST complete". A light blue cursor is positioned at the start of the first line. Below the second line, there is a tilde symbol (~).

Рис. 7: Создание файла

Я создаю bash-файл program6.sh и файл с программой на C, затем открываю их для редактирования в текстовом редакторе. (рис. 8)

A terminal window with a black background and green text. It shows three lines of commands being executed in a shell. The first two lines are 'touch program6.sh' and 'touch cprogram.c', both of which have been executed successfully. The third line shows the prompt 'sayprachanh1svnv@sayprachanh1svnv:~\$' followed by a cursor, indicating the next command is yet to be entered.

```
sayprachanh1svnv@sayprachanh1svnv:~$ touch program6.sh
sayprachanh1svnv@sayprachanh1svnv:~$ touch cprogram.c
sayprachanh1svnv@sayprachanh1svnv:~$
```

Рис. 8: Создание файлов

## Выполнение лабораторной работы


В файле С-программы я ввожу код, который запрашивает число, определяет, больше ли оно нуля, меньше или равно нулю, и завершает программу с помощью функции `exit`. (рис. 9)

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     int n;
6     printf("Enter the number: ");
7     scanf("%d", &n);
8
9     if(n > 0){
10         exit(1);
11     }else if(n == 0){
12         exit(0);
13     }else{
14         exit(2);
15     }
```

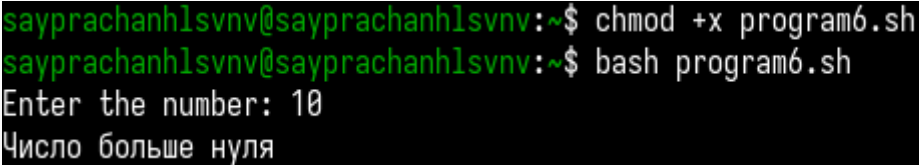


## Выполнение лабораторной работы

В bash-файл `program6.sh` я добавляю команду для компиляции С-программы, вызова программы и вывода сообщения на экран. (рис. 10)

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

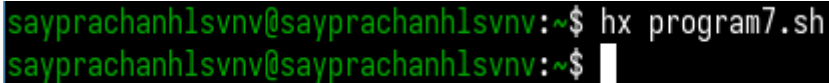
Я даю файлу разрешение на выполнение и запускаю программу. Ввожу число больше нуля, и программа корректно отображает сообщение. (рис. 11)

A terminal window with a black background and green text. The prompt is 'sayprachanhlsvnnv@sayprachanhlsvnnv:~\$'. The first command is 'chmod +x program6.sh'. The second command is 'bash program6.sh'. The program outputs 'Enter the number: 10' and then 'Число больше нуля'.

```
sayprachanhlsvnnv@sayprachanhlsvnnv:~$ chmod +x program6.sh
sayprachanhlsvnnv@sayprachanhlsvnnv:~$ bash program6.sh
Enter the number: 10
Число больше нуля
```

Рис. 11: Запуск программы

Я создаю bash-файл program7.sh и открываю его в текстовом редакторе. (рис. 12)

A terminal window with a black background and green text. The prompt is 'sayprachanh1svn@sayprachanh1svn:~\$'. The command 'hx program7.sh' has been entered. The next line shows the prompt 'sayprachanh1svn@sayprachanh1svn:~\$' followed by a white cursor block.

```
sayprachanh1svn@sayprachanh1svn:~$ hx program7.sh  
sayprachanh1svn@sayprachanh1svn:~$
```

Рис. 12: Создание файла

## Выполнение лабораторной работы

Я добавляю код в файл `program7.sh`. Программа создаёт новые файлы и удаляет уже существующие, созданные этой же программой. Количество файлов определяется числом, переданным в аргументе командной строки. (рис. 13)

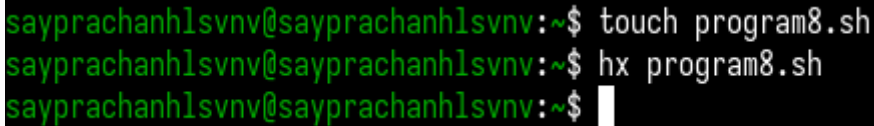
```
1  #!/bin/bash
2  for((i = 1; i <= $*; i++))
3  do
4  if test -f "$i".tmp
5  then rm "$i.tmp"
6
7  else touch "$i.tmp"
8
9  fi
10 done
```

## Выполнение лабораторной работы

Я запускаю программу. Ввожу число 3 в качестве аргумента. В результате программа создаёт 3 файла, и если я снова запускаю команду, то программа удаляет все 3 файла. (рис. 14)

```
sayprachanhlsnv@sayprachanhlsnv:~$ ./program7.sh 3
sayprachanhlsnv@sayprachanhlsnv:~$ ls
1.tmp      bin          feathers     lab07.sh~
2.tmp      conf.txt     file.txt     LICENSE
3.tmp      cprogram    git-extended log.txt
abc1       cprogram.c  git-pass     LOG.txt
australia  Documents   HelloWorld   may
backup     Downloads   lab07.sh     monthly
sayprachanhlsnv@sayprachanhlsnv:~$ ./program7.sh 3
sayprachanhlsnv@sayprachanhlsnv:~$ ls
abc1       cprogram.c  git-pass     LOG.txt
australia  Documents   HelloWorld   may
```

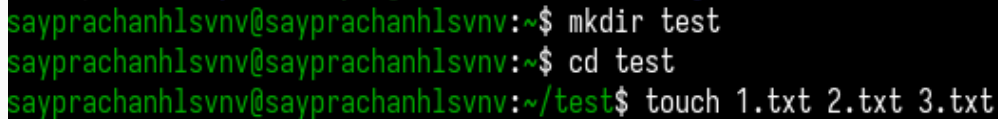
Затем я создаю bash-файл program8.sh и открываю его в текстовом редакторе. (рис. 15)

A terminal window with a black background and green text. It shows three lines of commands being executed in a shell. The first line creates a file named 'program8.sh' using the 'touch' command. The second line opens the file in a text editor named 'hx'. The third line shows the prompt with a cursor, indicating the editor is ready for input.

```
sayprachanh1svnv@sayprachanh1svnv:~$ touch program8.sh
sayprachanh1svnv@sayprachanh1svnv:~$ hx program8.sh
sayprachanh1svnv@sayprachanh1svnv:~$
```

Рис. 15: Создание файла

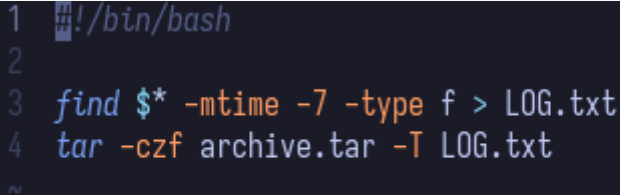
Я также создаю тестовую директорию и несколько текстовых файлов внутри этой директории. (рис. 16)

A terminal window with a black background and green text. It shows three lines of commands being executed. The first line creates a directory named 'test'. The second line changes the current directory to 'test'. The third line creates three text files named '1.txt', '2.txt', and '3.txt' using the 'touch' command. The prompt changes from '~\$' to '~/test\$' after the directory change.

```
sayprachanhlsnv@sayprachanhlsnv:~$ mkdir test  
sayprachanhlsnv@sayprachanhlsnv:~$ cd test  
sayprachanhlsnv@sayprachanhlsnv:~/test$ touch 1.txt 2.txt 3.txt
```

Рис. 16: Создание каталога и файлов

Я добавляю код в файл program8.sh. Программа создаёт архив в указанной директории и включает в него только те файлы, которые были изменены менее недели назад. (рис. 17)



```
1  #!/bin/bash
2
3  find $* -mtime -7 -type f > LOG.txt
4  tar -czf archive.tar -T LOG.txt
~
```

Рис. 17: Код программы



Я даю файлу разрешение на выполнение и запускаю программу. (рис. 18) Ввожу путь к указанной директории (тестовой директории), и программа создаёт архив этой директории и файлов. Программа работает корректно. (рис. 19) В файле LOG.txt содержится список файлов, включённых в архив. (рис. 20)

```
sayprachanh1svnv@sayprachanh1svnv:~$ chmod +x program8.sh  
sayprachanh1svnv@sayprachanh1svnv:~$ ./program8.sh /home/sayprachanh1svnv/test
```

Рис. 18: Запуск программы



Рис. 19: Созданный архив

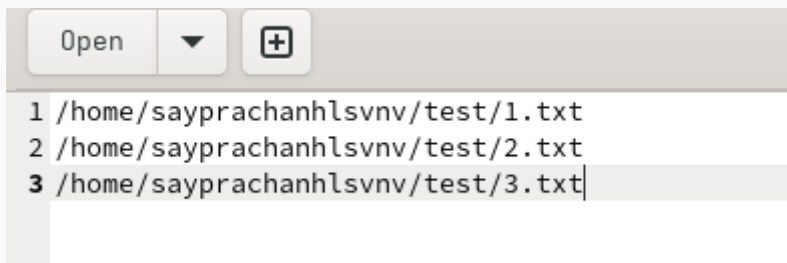


Рис. 20: Список файлов

Во время этой лабораторной работы я изучил основы программирования в оболочке UNIX. Научился писать более сложные пакетные файлы, используя логические структуры управления и циклы.

Спасибо за внимание

---