

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 12

дисциплина: *Операционные системы*

Студент: Юрченко Артём Алексеевич Группа: НФИбд-02-20

МОСКВА

2021 г.

Цель работы

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

Выполнение лабораторной работы

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

```
1#!/bin/bash
2while getopts i:o:p:Cn oletter
3do case $opletter in
4 i) iflag=1; ival=$OPTARG;;
5 o) oflag=1; oval=$OPTARG;;
6 p) pflag=1; pval=$OPTARG;;
7 C) Cflag=1;;
8 n) nflag=1;;
9 *) echo Ключи и параметры заданы неправильно!
10 esac
11 done
12
13 if [[ $Cflag == 1 ]]; then
14     if [[ $nflag == 1 ]]; then
15         grep $pflag -n $ival
16         if [[ $oflag == 1 ]]; then
17             grep $pval -n $ival > $oval
18             echo Данные были успешно записаны в файл
19         fi
20     else
21         grep $pval $ival
22         if [[ $oflag == 1 ]]; then
23             grep $pval -n -i $ival > $oval
24             echo Данные были успешно записаны в файл
25         fi
26     fi
27 else
28     if [[ $nflag == 1 ]]; then
29         grep $pval -n -i $ival
30         if [[ $oflag == 1 ]]; then
31             grep $pval -n -i $ival > $oval
32             echo Данные были успешно записаны в файл
33         fi
34     else
35         grep $pval -i $ival
36         if [[ $oflag == 1 ]]; then
37             grep $pval -i $ival > $oval
```

```
aaurchenko@aaau:~$ bash lab12 1.sh -i als -o als2 -pHey -C -n
```

```
aaurchenko@aaau:~$ bash lab12 1.sh -i als -o als2 -pHey -C -n
```

2. Пишем 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 вызывает эту программу и, проанализировав с помощью команды `$?` , выдаёт сообщение о том, какое число было введено.

```
Текстовый редактор Чт, 28 октября 21:45
lab12_2.cpp
Открыть Сохранить

1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     cout << "Введите число";
7     int n;
8     cin >> n;
9     if (n > 0) { exit(1); };
10    if (n < 0) { exit(2); };
11    if (n == 0) { exit(0); };
12    return 0;
13 }
14 }
```

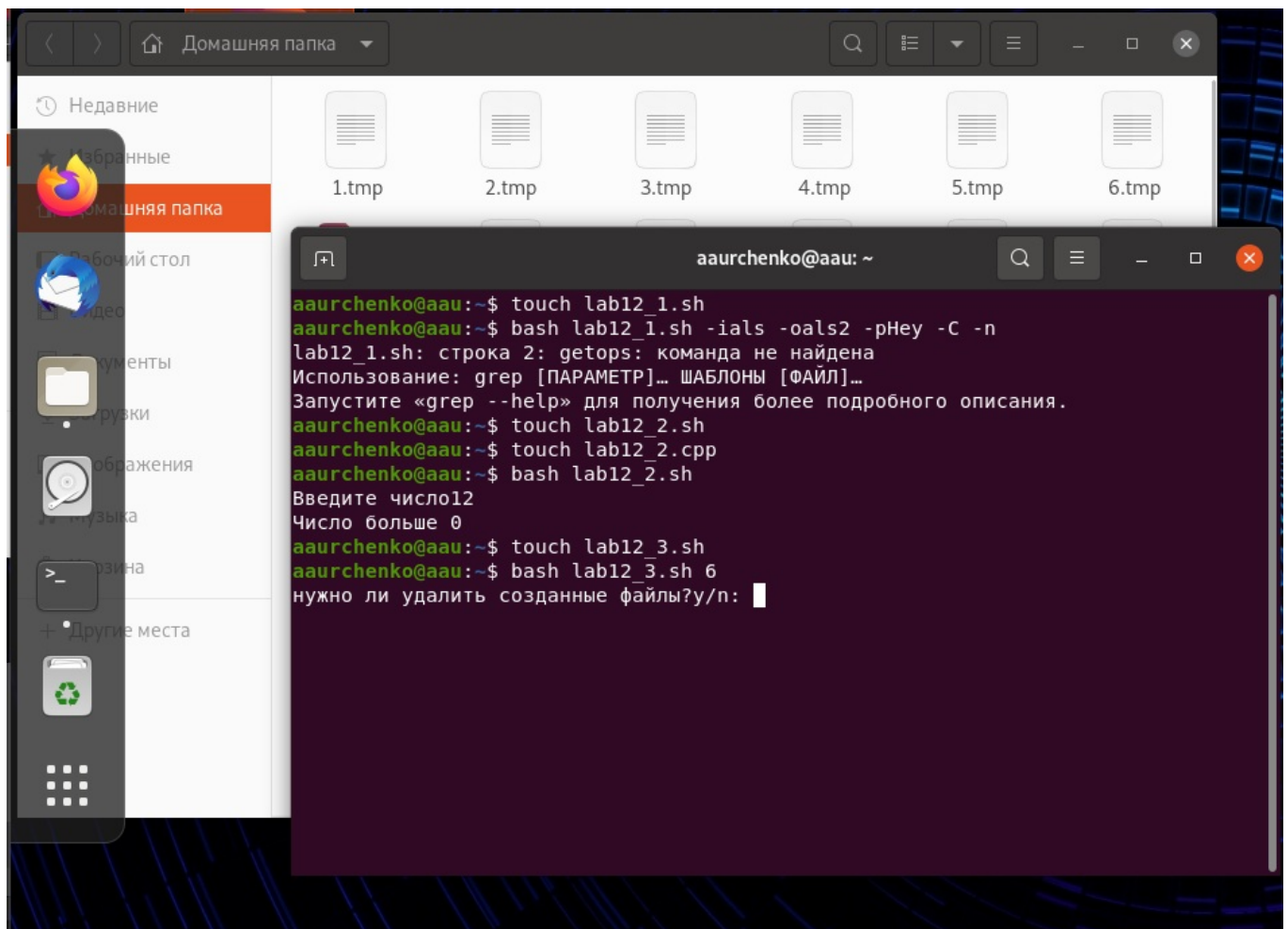
```
*lab12_2.sh
Открыть

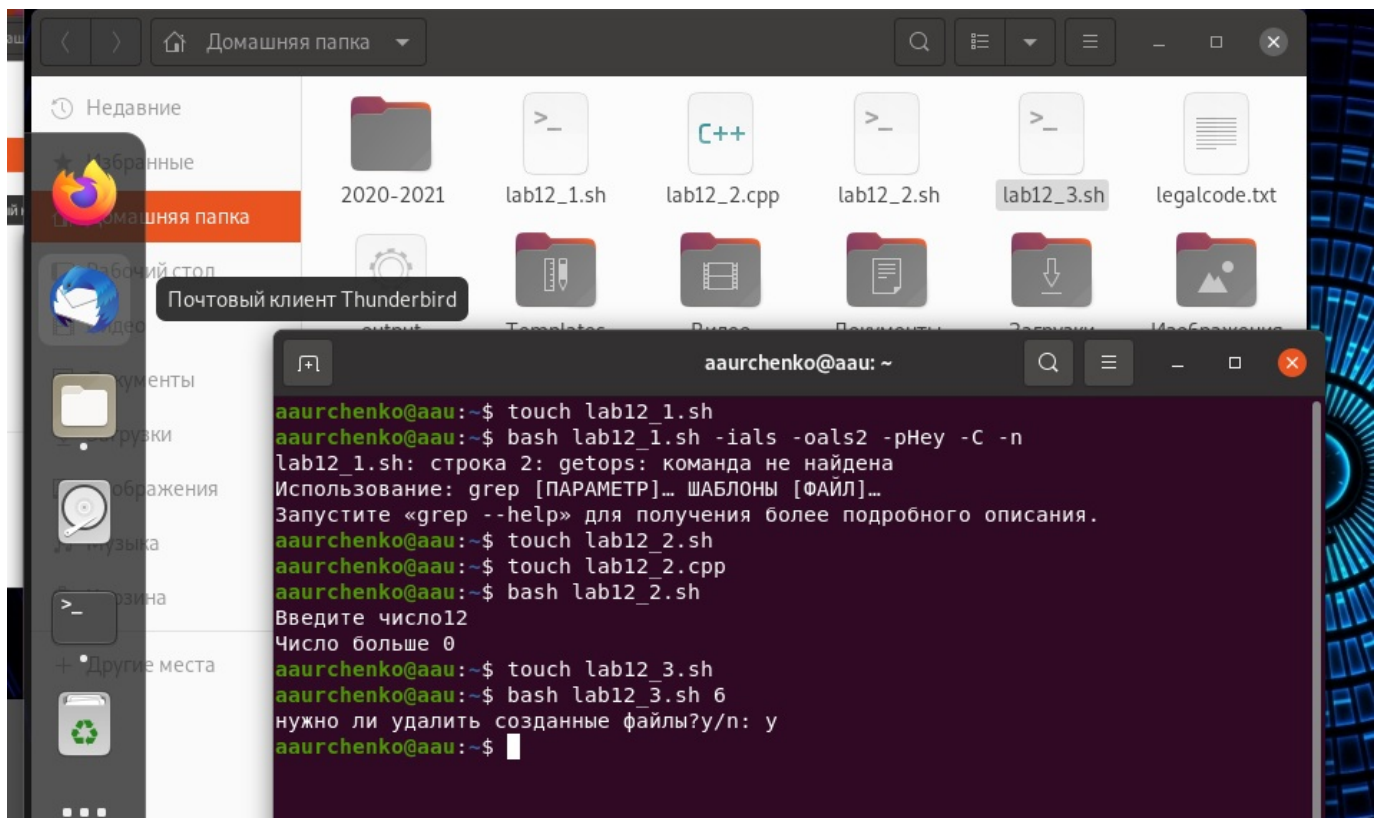
1 #!/bin/bash
2 g++ lab12_2.cpp -o output
3 ./output
4 case $? in
5 0) echo Число равно 0 ;;
6 1) echo Число больше 0 ;;
7 2) echo Число меньше 0 ;;
8 esac
```

```
aaurchenko@aa:~$ touch lab12_2.sh
aaurchenko@aa:~$ touch lab12_2.cpp
aaurchenko@aa:~$
```

3. Пишем командный файл, который создаст указанное число файлов, пронумерованных последовательно от 0 до N (например 0.tmp, 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 умеет удалять все созданные им файлы.

```
Открыть *lab12_3.sh Сохранить
1 #!/bin/bash
2 for (( i = 1; i <= $1; i++ ))
3     do
4         touch ${i}.tmp
5     done
6
7 read -p 'нужно ли удалить созданные файлы?y/n: ' marker
8 if [[ $marker == 'y' ]]; then
9     for (( i = 1; i <= $1; i++ ))
10        do
11            rm ${i}.tmp
12        done
13 fi
```





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

Выводы

Мы изучили основы программирования в оболочке ОС UNIX. Закрепили знания, полученные в прошлых работах. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Термины

- Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера.
- POSIX (Portable Operating System Interface for Computer Environments) — набор стандартов описания интерфейсов взаимодействия операционной системы и прикладных программ.
- Последовательность команд может быть помещена в текстовый файл. Такой файл называется командным.
- Флаги — это опции командной строки, обычно помеченные знаком минус; Например, для команды ls флагом может являться -F.
- Каталог, он же директория, (от английского Directory) — это объект в ФС (файловой системе), необходимый для того, чтобы упростить работу с файлами.
- Домашний каталог - каталог, предназначенный для хранения собственных данных пользователя Linux. Как правило, является текущим непосредственно после регистрации пользователя в системе.
- Команда - записанный по специальным правилам текст (возможно с аргументами), представляющий собой указание на выполнение какой-либо функций (или действий) в операционной системе.