

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 8

дисциплина: Операционные системы

Студент: Бешкуров Михаил

Группа: НК-101

МОСКВА

2019 г.

1. Цель работы

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

2. Описание процесса выполнения задания

1. Написал командный файл

```
#!/bin/bash
grep -i $2 $1
```

2. Мы написали 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 о коде завершения в оболочку. Командный файл должен вызывать эту программу и проанализировав с помощью команды `$?`. Код возврата, выдает сообщение о том, какое число было введено.

```
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $ ls
c.cpp LabPDF lab.sh
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $ chmod +x lab.sh
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $ g++ -Wall -o c c.cpp
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $ ./lab.sh
4
Больше нуля4
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $
```

```
#include<iostream>
#include<stdio.h>
#include<stdlib.h>
using namespace std;

int main(){
    int a;
    cin >> a;
    if (a>0)
        cout << "Больше нуля";
    else if (a<0)
        cout << "Меньше нуля";
    else
        cout << "Равно нулю";
    exit(a);
    return a;
}
```

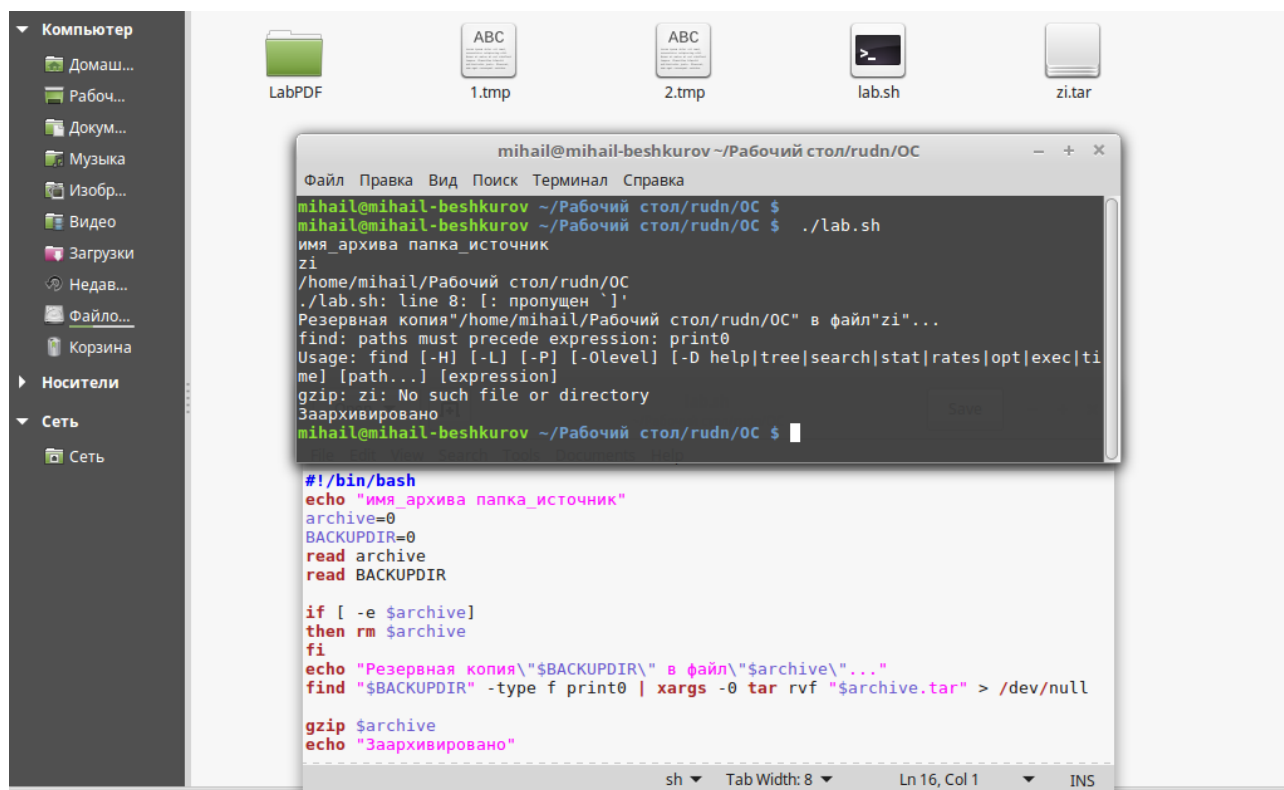
3. Мы написали командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (например, 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.). Число файлов, которые необходимо создать, передается в аргумент командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).



```
#!/bin/bash
name=1
k=0
read k
while (($name!=($k+1)))
do
    for i in $name
    do
        touch $i.tmp
    done
    ((name+=1))
done
echo "Файлы созданы, удалить их? (0/1)"
name=1
r=0
read r
if (r==1)
then
    while (($name!=($k + 1)))
    do
        for i in $name
        do
            rm $i.tmp
        done
        ((name+=1))
    done
fi
```

1,1 Наверху

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



```
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC
Файл Правка Вид Поиск Терминал Справка
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $ ./lab.sh
имя_архива папка_источник
zi
/home/mi hail/Рабочий стол/rudn/OC
./lab.sh: line 8: [: пропущен `]'
Резервная копия"/home/mi hail/Рабочий стол/rudn/OC" в файл"zi"...
find: paths must precede expression: print0
Usage: find [-H] [-L] [-P] [-Olevel] [-D help|tree|search|stat|rates|opt|exec|ti
me] [path...] [expression]
gzip: zi: No such file or directory
Заархивировано
mi hail@mi hail-beshkurov ~/Рабочий стол/rudn/OC $

#!/bin/bash
echo "имя_архива папка_источник"
archive=0
BACKUPDIR=0
read archive
read BACKUPDIR

if [ -e $archive]
then rm $archive
fi
echo "Резервная копия\"$BACKUPDIR\" в файл\"$archive\"..."
find "$BACKUPDIR" -type f print0 | xargs -0 tar rvf "$archive.tar" > /dev/null

gzip $archive
echo "Заархивировано"
```

3. Вывод

Изучил основы программирования в оболочке ОС UNIX/Linux.

Научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

4. Контрольные вопросы

1. Каково предназначение команды getoptс?

Команда getoptс используется для разбора позиционных параметров и проверки опций на допустимость.

2. Какое отношение метасимволы имеют к генерации имён файлов?

Метасимволы отвечают за параметры выдачи файлов.

3. Какие операторы управления действиями вы знаете?

for, case, while, until, break, continue, if

4. Какие операторы используются для прерывания цикла?

break

5. Для чего нужны команды false и true?

Аналог логических значений (0/1) для функций.

7. Объясните различия между конструкциями while и until.

В конструкции while проверка условия выхода выполняется вначале, а не в конце цикла, если условие не удовлетворяется до начала выполнения цикла, то управление передается оператору стоящему сразу за телом цикла. Условие выхода удовлетворяется, если выражение, определяющее условие выхода, ложно.

Между зарезервированными словами until может размещаться несколько операторов не применяя операторные скобки, когда как в конструкции while только один. Условие выхода удовлетворяется, если выражение, определяющее условие выхода, истинно.