

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 12 «Программирование в командном процессоре ОС UNIX. Ветвления и циклы»

дисциплина: *Операционные системы*

Студент: Юсупов Шухратджон Фирдавсович

Группа: НПИбд-02-20

МОСКВА 2021 г.

Цель работы: Изучить основы программирования в оболочке ОС UNIX/Linux.
Научиться писать небольшие командные файлы.

Ход работы

1. Используя команды `getopts` `grep`, напомним командный файл, который анализирует командную строку с ключами:
 - i — прочитать данные из указанного файла;
 - o — вывести данные в указанный файл;
 - r — указать шаблон для поиска;
 - C — различать большие и малые буквы (по умолчанию включен)
 - n — выдавать номера строк. (Рис.1-2)

```
#!/bin/bash

t_=""
o_=""
p_=""
C_=0
n_=0

while getopts "t:o:p:C:n" opt
do
case $opt in
t) t_="$OPTARG";;
o) o_="$OPTARG";;
p) p_="$OPTARG";;
C) C_=$((C_+1));;
n) n_=$((n_+1));;
esac
done
if (($C_ + $n_ == 2))
then
grep -t -n "$p_" "$t_" > "$o_"
elif (($C_ == 1))
then
grep -n "$p_" "$t_" > "$o_"
fi
```

рис. 1

```
[root@localhost laboratory]# cd lab12
-bash: cd: lab12: Нет такого файла или каталога
[root@localhost laboratory]# mkdir lab12
[root@localhost laboratory]# cd lab12
[root@localhost lab12]# ./sch1.sh -i from.txt -o to.txt -p abc
-bash: ./sch1.sh: Нет такого файла или каталога
[root@localhost lab12]# cat
```

рис. 2

2. Написал 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. (Рис. 3-4)

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
int main(){
    int j;
    scanf("%d > 0 \n", j);
    if (j>0) cout<<j<<">0">>endl;
    else if (j<0) cout<<j<<"<0"<<endl;
    else printf("0 = 0\n");
    exit(j);
    return j;
}
```

рис. 3

```
#!/bin/bash

./sch2.sh
echo $?
```

рис. 4

3. Написал 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

```
#!/bin/bash
r_=0
n_=0
name=1
while getopts "r:n:" opt
do
case opt in
r)r_="$OPTARG";;
n)n_="$OPTARG";;
esac
done
if [ "$r_" -eq "0" ]
then
while(( $name != (( $n_ + 1 ))) )
do
for i in $name
do
touch $i.tmp
done
((name+=1))
done
else
name=1
while(( $name != (( $n_ + 1 ))) )
do
for i in $name
do
rm $i.tmp
done
((name+=1))
done
fi
fi
```

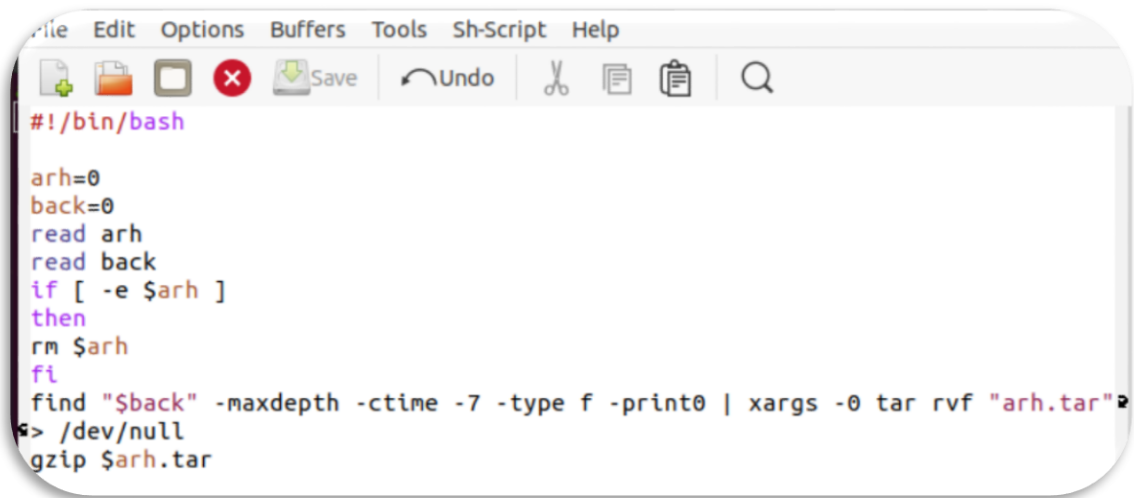
существуют). (Рис. 5-6)

рис. 5

```
[root@localhost lab12]# ls
[root@localhost lab12]# ./sch3.sh -r 0 -n 3
-bash: ./sch3.sh: Нет такого файла или каталога
[root@localhost lab12]# ls
[root@localhost lab12]#
```

рис. 6

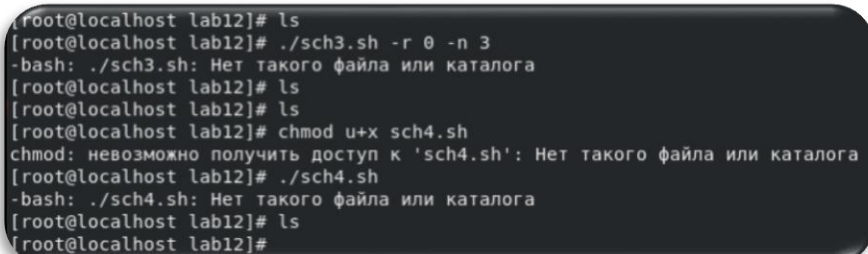
4. Напишем командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. модифицируем его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад. (Рис. 7-8)



```
#!/bin/bash

arh=0
back=0
read arh
read back
if [ -e $arh ]
then
rm $arh
fi
find "$back" -maxdepth -ctime -7 -type f -print0 | xargs -0 tar rvf "arh.tar"
> /dev/null
gzip $arh.tar
```

рис.



```
[root@localhost lab12]# ls
[root@localhost lab12]# ./sch3.sh -r 0 -n 3
-bash: ./sch3.sh: Нет такого файла или каталога
[root@localhost lab12]# ls
[root@localhost lab12]# ls
[root@localhost lab12]# chmod u+x sch4.sh
chmod: невозможно получить доступ к 'sch4.sh': Нет такого файла или каталога
[root@localhost lab12]# ./sch4.sh
-bash: ./sch4.sh: Нет такого файла или каталога
[root@localhost lab12]# ls
[root@localhost lab12]#
```

7

рис. 8

ВЫВОД: В ходе работы мы изучили основы программирования в оболочке ОС UNIX. Научились писать более сложные командные файлы с использованием логических управляющих инструкций и циклов.