

Лабораторная работа No 12

**Программирование в командном процессоре ОС UNIX. Ветвления и
циклы**

Кеан Путхеаро НПИбд-01-20

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	7
4	Вывод	14
5	Библиография	15

Список иллюстраций

3.1	Первый скрипт	8
3.2	Проверка работы скрипта	8
3.3	Проверка работы скрипта	9
3.4	Проверка работы скрипта	9
3.5	создание файла program2.c	9
3.6	Файл program2.c	10
3.7	Файл program2.sh	10
3.8	результат работы командного файла	10
3.9	создание файла Program3.sh	11
3.10	результат работы командного файла	11
3.11	результат работы командного файла	12
3.12	создание файла Program4.sh	12
3.13	результат работы	13
3.14	результат работы	13

1 Цель работы

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

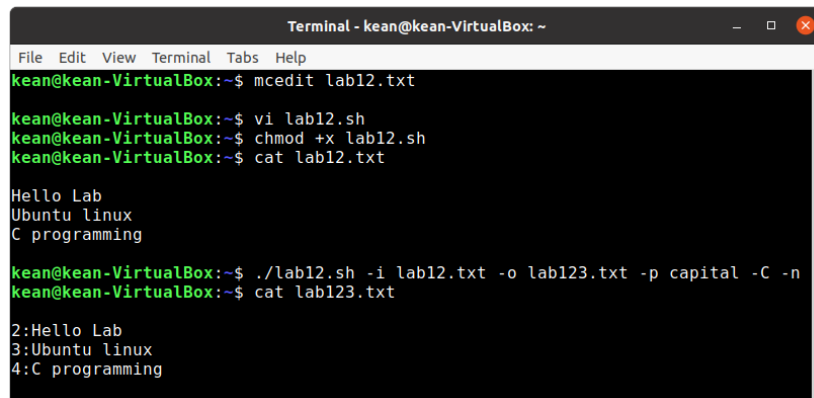
2 Задание

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами:
 - inputfile— прочитать данные из указанного файла;
 - outputfile— вывести данные в указанный файл; --ршаблон— указать шаблон для поиска;
 - C— различать большие и малые буквы;
 - n— выдавать номера строк.а затем ищет в указанном файле нужные строки, определяемые ключом-р.
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 код 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

вархив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад(использовать командуfind).

3 Выполнение лабораторной работы

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: `-iinputfile` — прочитать данные из указанного файла; `-ooutputfile` — вывести данные в указанный файл; `-rшаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-p`.



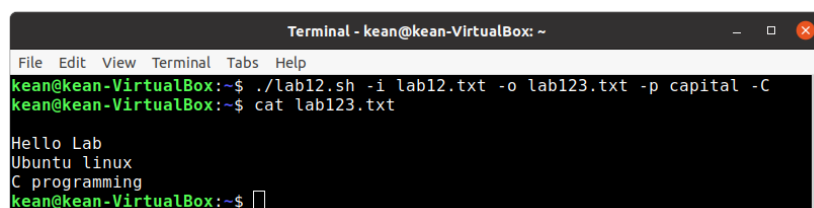
```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ mcedit lab12.txt
kean@kean-VirtualBox:~$ vi lab12.sh
kean@kean-VirtualBox:~$ chmod +x lab12.sh
kean@kean-VirtualBox:~$ cat lab12.txt

Hello Lab
Ubuntu linux
C programming

kean@kean-VirtualBox:~$ ./lab12.sh -i lab12.txt -o lab123.txt -p capital -C -n
kean@kean-VirtualBox:~$ cat lab123.txt

2:Hello Lab
3:Ubuntu linux
4:C programming
```

Рис. 3.3: Проверка работы скрипта

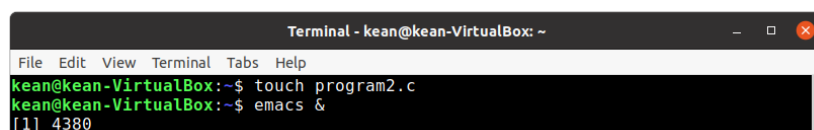


```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ ./lab12.sh -i lab12.txt -o lab123.txt -p capital -C
kean@kean-VirtualBox:~$ cat lab123.txt

Hello Lab
Ubuntu linux
C programming
kean@kean-VirtualBox:~$
```

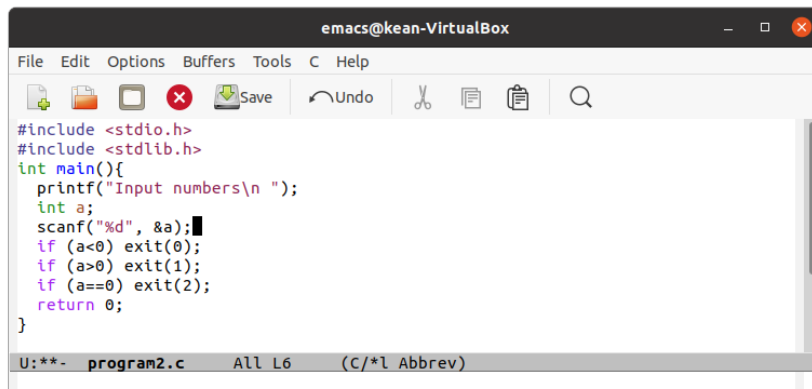
Рис. 3.4: Проверка работы скрипта

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.



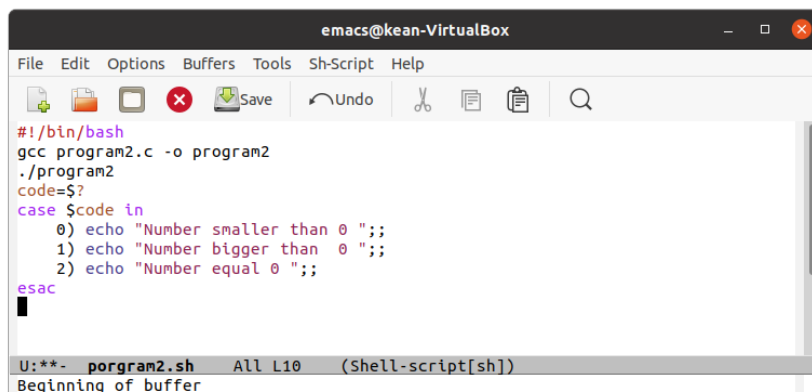
```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ touch program2.c
kean@kean-VirtualBox:~$ emacs &
[1] 4380
```

Рис. 3.5: создание файла program2.c



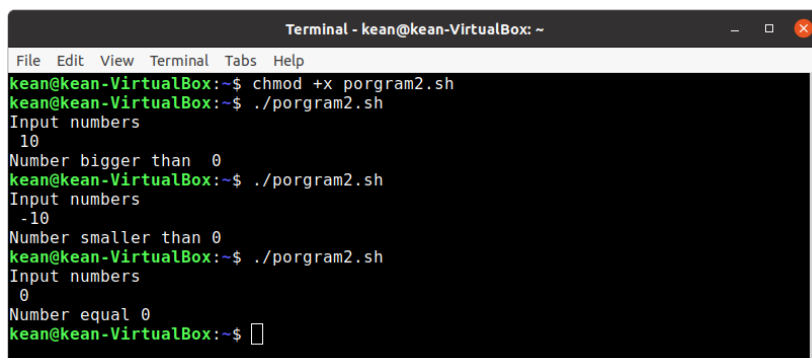
```
emacs@kean-VirtualBox
File Edit Options Buffers Tools C Help
[Icons] Save Undo [Icons]
#include <stdio.h>
#include <stdlib.h>
int main(){
    printf("Input numbers\n ");
    int a;
    scanf("%d", &a);
    if (a<0) exit(0);
    if (a>0) exit(1);
    if (a==0) exit(2);
    return 0;
}
U:**- program2.c All L6 (C/*l Abbrev)
```

Рис. 3.6: Файл program2.c



```
emacs@kean-VirtualBox
File Edit Options Buffers Tools Sh-Script Help
[Icons] Save Undo [Icons]
#!/bin/bash
gcc program2.c -o program2
./program2
code=$?
case $code in
    0) echo "Number smaller than 0 ";;
    1) echo "Number bigger than 0 ";;
    2) echo "Number equal 0 ";;
esac
█
U:**- program2.sh All L10 (Shell-script[sh])
Beginning of buffer
```

Рис. 3.7: Файл program2.sh

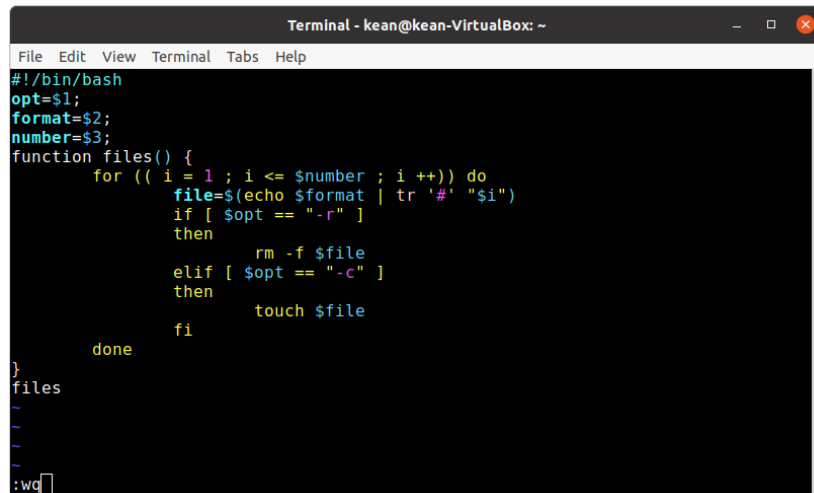


```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ chmod +x program2.sh
kean@kean-VirtualBox:~$ ./program2.sh
Input numbers
10
Number bigger than 0
kean@kean-VirtualBox:~$ ./program2.sh
Input numbers
-10
Number smaller than 0
kean@kean-VirtualBox:~$ ./program2.sh
Input numbers
0
Number equal 0
kean@kean-VirtualBox:~$ █
```

Рис. 3.8: результат работы командного файла

3. Написал командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и

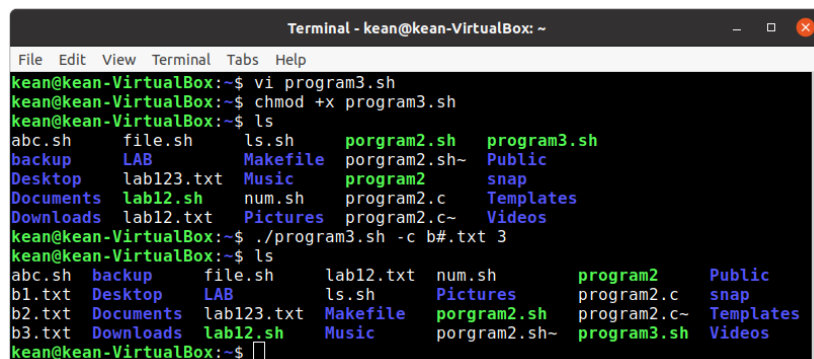
т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).



```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
#!/bin/bash
opt=$1;
format=$2;
number=$3;
function files() {
    for (( i = 1 ; i <= $number ; i ++)) do
        file=$(echo $format | tr '#' '$i')
        if [ $opt == "-r" ]
        then
            rm -f $file
        elif [ $opt == "-c" ]
        then
            touch $file
        fi
    done
}
files
:wc
```

Рис. 3.9: создание файла Program3.sh

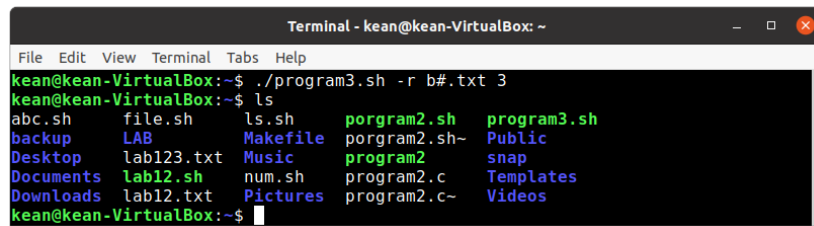
файлы b1.txt b2.txt b3.txt созданы.



```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ vi program3.sh
kean@kean-VirtualBox:~$ chmod +x program3.sh
kean@kean-VirtualBox:~$ ls
abc.sh  file.sh  ls.sh  program2.sh  program3.sh
backup  LAB      Makefile  program2.sh~  Public
Desktop lab123.txt Music  program2      snap
Documents lab12.sh num.sh  program2.c  Templates
Downloads lab12.txt Pictures program2.c~  Videos
kean@kean-VirtualBox:~$ ./program3.sh -c b#.txt 3
kean@kean-VirtualBox:~$ ls
abc.sh  backup  file.sh  lab12.txt  num.sh  program2  Public
b1.txt  Desktop LAB      ls.sh     Pictures  program2.c  snap
b2.txt  Documents lab123.txt Makefile  program2.sh  program2.c~  Templates
b3.txt  Downloads lab12.sh  Music    program2.sh~ program3.sh  Videos
kean@kean-VirtualBox:~$
```

Рис. 3.10: результат работы командного файла

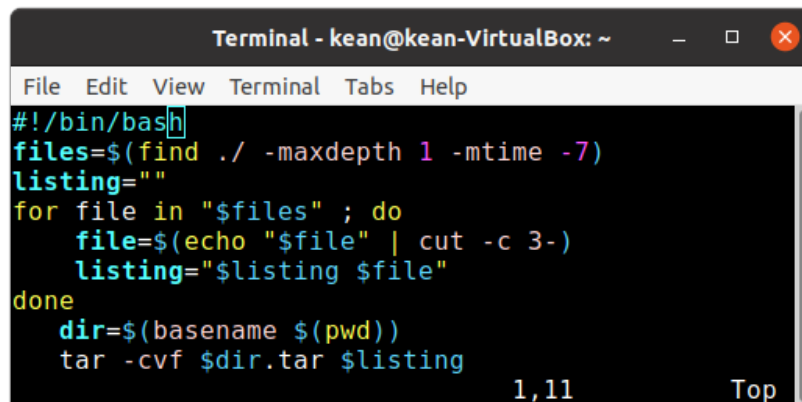
файлы b1.txt b2.txt b3.txt удалены.



```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~$ ./program3.sh -r b#.txt 3
kean@kean-VirtualBox:~$ ls
abc.sh      file.sh      ls.sh        program2.sh  program3.sh
backup      LAB          Makefile     program2.sh~ Public
Desktop     lab123.txt   Music        program2     snap
Documents   lab12.sh     num.sh       program2.c   Templates
Downloads   lab12.txt    Pictures     program2.c~  Videos
kean@kean-VirtualBox:~$
```

Рис. 3.11: результат работы командного файла

4. Написал 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).



```
Terminal - kean@kean-VirtualBox: ~
File Edit View Terminal Tabs Help
#!/bin/bas[h]
files=$(find ./ -maxdepth 1 -mtime -7)
listing=""
for file in "$files" ; do
    file=$(echo "$file" | cut -c 3-)
    listing="$listing $file"
done
dir=$(basename $(pwd))
tar -cvf $dir.tar $listing
1,11 Top
```

Рис. 3.12: создание файла Program4.sh

```
Terminal - kean@kean-VirtualBox: ~/New
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~/New$ chmod +x program4.sh
kean@kean-VirtualBox:~/New$ ls -l
total 68
-rw-rw-r-- 1 kean kean 95 мая 28 15:41 abc.sh
-rw-rw-r-- 1 kean kean 187 мая 28 17:14 file.sh
-rw-rw-r-- 1 kean kean 38 мая 28 23:22 lab123.txt
-rwxrwxr-x 1 kean kean 762 мая 28 23:17 lab12.sh
-rw-rw-r-- 1 kean kean 39 мая 28 22:25 lab12.txt
-rw-rw-r-- 1 kean kean 141 мая 28 16:09 ls.sh
-rw-rw-r-- 1 kean kean 118 мая 28 15:47 num.sh
-rwxrwxr-x 1 kean kean 187 мая 28 23:29 porgram2.sh
-rwxrwxr-x 1 kean kean 16848 мая 28 23:31 program2
-rw-rw-r-- 1 kean kean 184 мая 28 23:27 program2.c
-rwxrwxr-x 1 kean kean 248 мая 28 23:35 program3.sh
-rwxrwxr-x 1 kean kean 210 мая 28 23:48 '#program4.sh#'
-rwxrwxr-x 1 kean kean 219 мая 28 23:56 program4.sh
kean@kean-VirtualBox:~/New$ ./program4.sh
program2.c
program4.sh
lab123.txt
abc.sh
#program4.sh#
porgram2.sh
program3.sh
num.sh
file.sh
ls.sh
lab12.txt
lab12.sh
program2
kean@kean-VirtualBox:~/New$
```

Рис. 3.13: результат работы

Итак, New.tar создан.

```
Terminal - kean@kean-VirtualBox: ~/New
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~/New$ ls
abc.sh      lab12.sh    New.tar     program2    '#program4.sh#'
file.sh     lab12.txt  num.sh      program2.c  program4.sh
lab123.txt  ls.sh     porgram2.sh program3.sh
kean@kean-VirtualBox:~/New$
```

Рис. 3.14: результат работы

4 Вывод

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

5 Библиография

1. (Лабораторная работа №12) https://esystem.rudn.ru/pluginfile.php/1142380/mod_resource/content/1/lab_shell_prog_2.pdf
2. (stackexchange) <https://vi.stackexchange.com/questions/10209/execute-current-buffer-as-bash-script-from-vim>
3. (BASH: функция getopt — используем опции в скриптах) https://esystem.rudn.ru/pluginfile.php/1142380/mod_resource/content/1/lab_shell_prog_2.pdf
4. (stackoverflow) <https://stackoverflow.com/questions/16483119/an-example-of-how-to-use-getopts-in-bash>