Отчёт по лабораторной работе

Лабораторная работа № 12.

Lukashov Nikita

Содержание

List of Tables

List of Figures

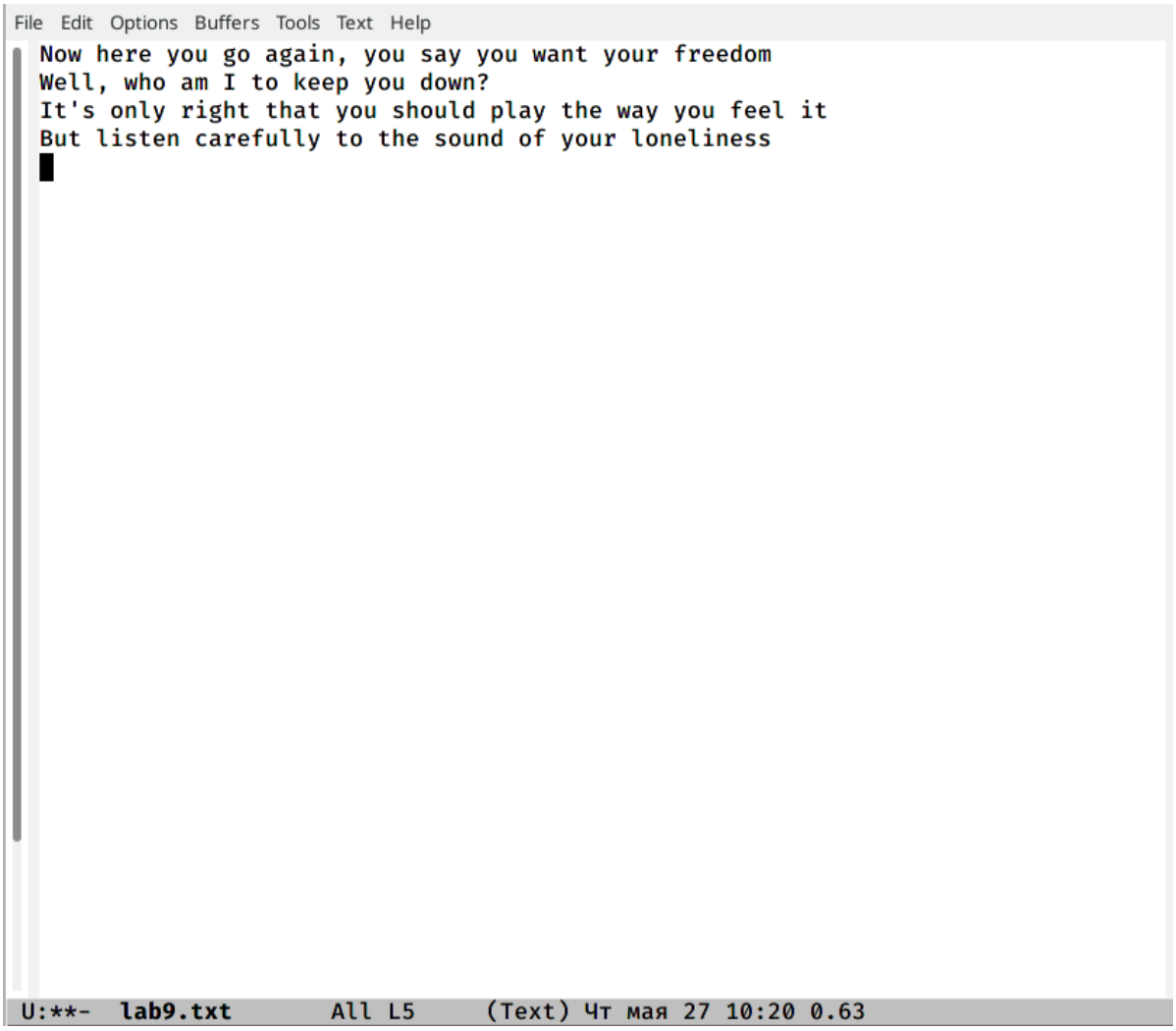
1 Цель работы

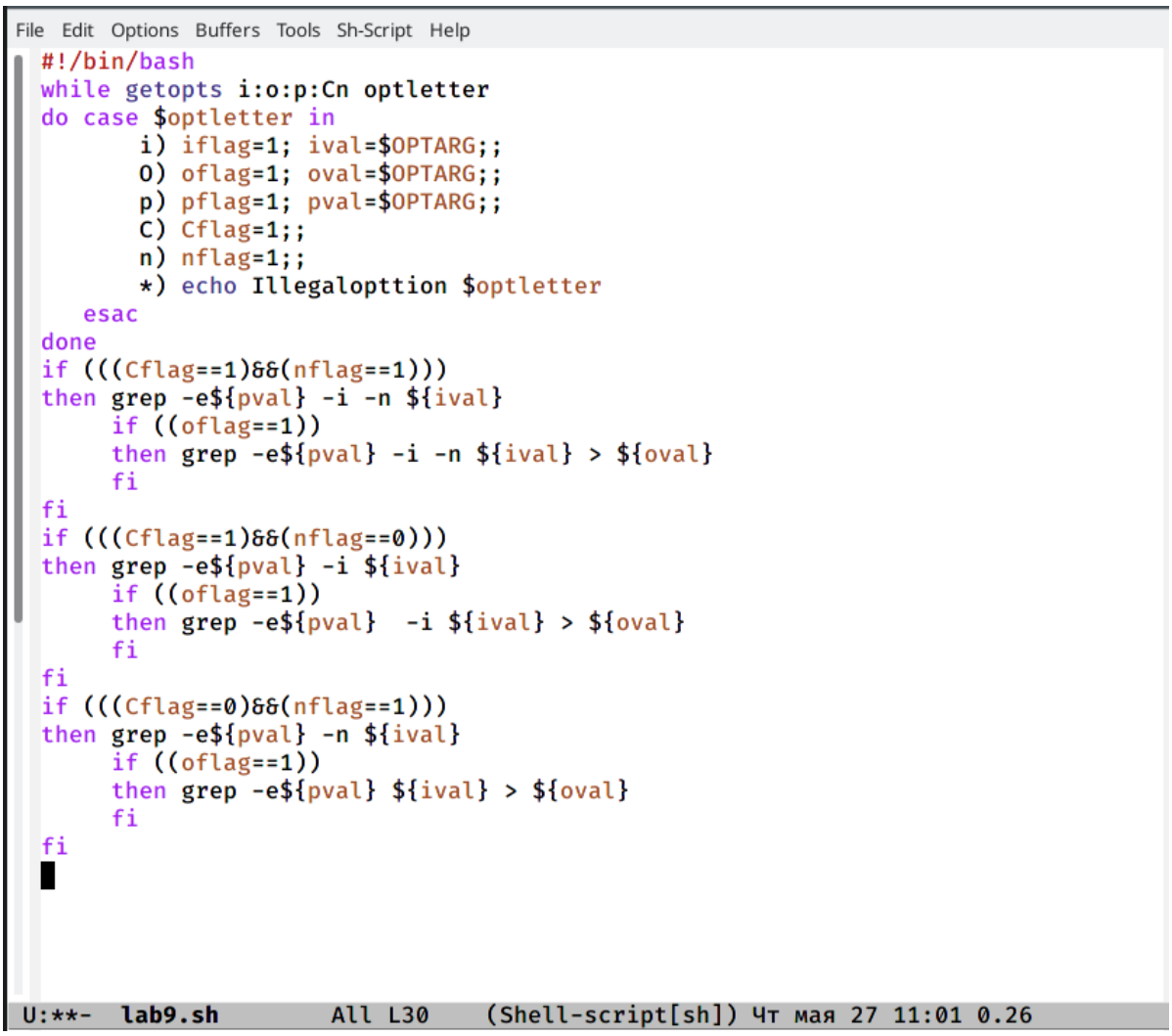
Изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

2 Выполнение лабораторной работы

1. Используя команды getopts grep, написал командный файл, который анализирует командную строку с ключами:

– -iinputfile — прочитать данные из указанного файла;
– -ooutputfile — вывести данные в указанный файл;
– -pшаблон — 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.


File Edit Options Buffers Tools Text Help
Now here you go again, you say you want your freedom
Well, who am I to keep you down?
It's only right that you should play the way you feel it
But listen carefully to the sound of your loneliness
U:**- lab9.txt All L5 (Text) Чт мая 27 10:20 0.63



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.

```c
#include <stdio.h>
#include <stdlib.h>

int main(){
  int a;
  printf("input:");
  scanf("%i",&a);
  if (a==0) exit (0);
  else if (a<0) exit (1);
  else if (a>0) exit (2);
  return (3);
}
```

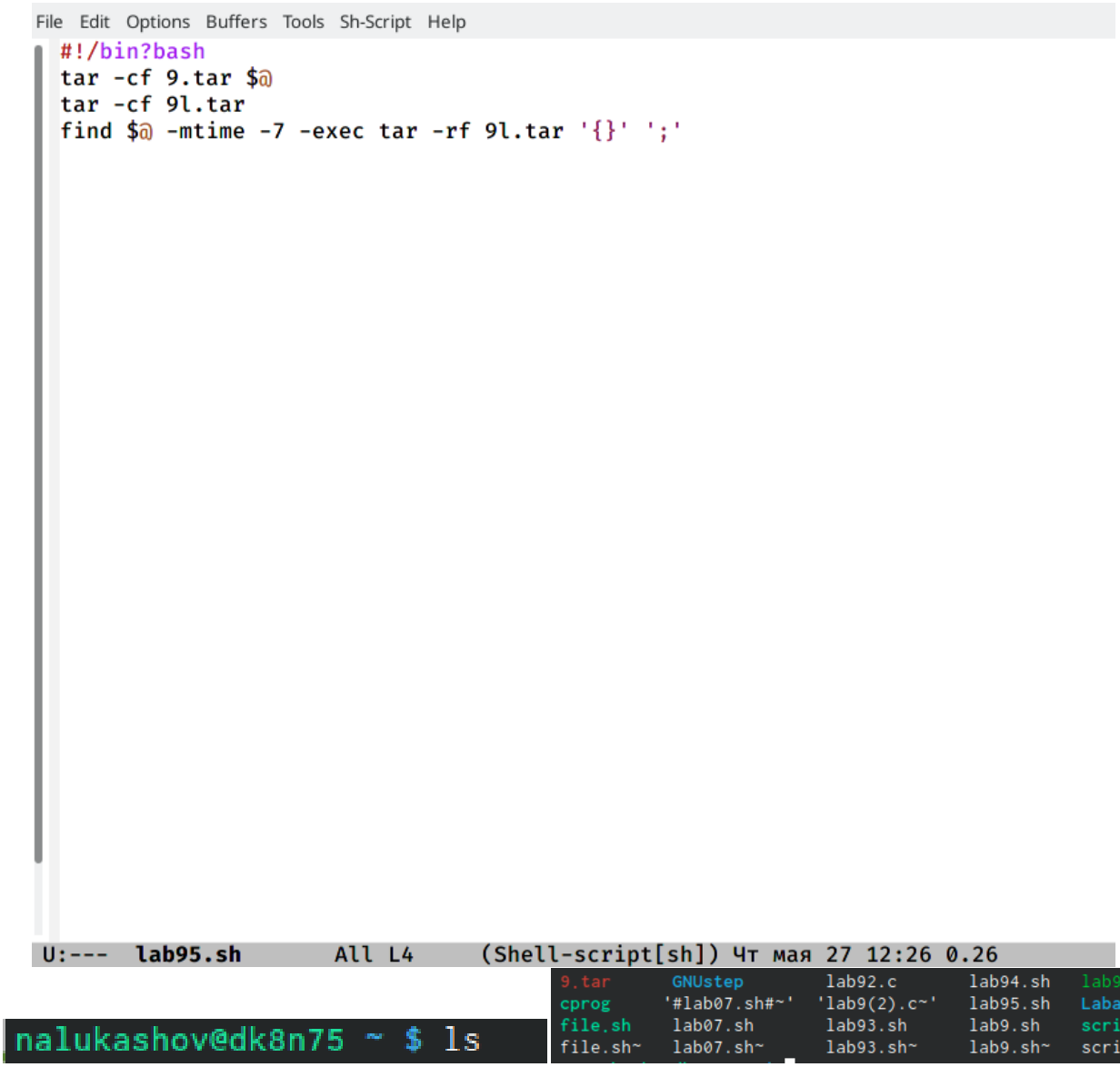
```
File  Edit  Options  Buffers  Tools  Sh-Script  Help
#!/bin/bash
gcc -o cprog lab92.c
./cprog
case $? in
    0)echo 'input number is equal to 0';;
    1)echo 'input number is smaller then 0';;
    2)echo 'input number is bigger then 0';;
esac
-:---  lab93.sh       All L2     (Shell-script[bash]) Чт мая 27 12:05 0.57
```

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).

```
File  Edit  Options  Buffers  Tools  Sh-Script  Help
#!/bin/bash
let dflag=0;
while getopts a:d optletter
do case $optletter in
       a) aflang=1; aval=$OPTARG;;
       d) dflag=1;;
       *) echo Illegaloptтion $optletter
   esac
done
#echo ${aval}
if ((dflag==0))
then for ((i=1;i<=aval;i++))
     do touch ${i}.txt
     done
fi
if ((dflag==1))
then for ((i=1;i<=aval;i++))
     do rm ${i}.txt
     done
fi
-:---  lab94.sh        All L1     (Shell-script[bash]) Чт мая 27 12:20 1.13
```

```
-rw-rw-r-- 1 nalukashov studsci       0 мая 13 12:31  abc1
-rwxr-xr-x 1 nalukashov studsci    1112 сен 24  2020  asdfg
-rw-r--r-- 1 nalukashov studsci     328 сен 24  2020  asdfg.asm
drwxr--r-- 2 nalukashov studsci    2048 мая 13 12:39  australia
drwxr-xr-x 3 nalukashov studsci    2048 апр 23 14:08  c++
-rw-rw-r-- 1 nalukashov studsci       0 мая 13 12:37  feathers
drwxr-xr-x 3 nalukashov studsci    2048 апр 23 13:51  GNUstep
-rw-r--r-- 1 nalukashov studsci    1112 окт  1  2020  lab03-1.asm
-rw-r--r-- 1 nalukashov studsci    1488 окт 21  2020  lab07
-rwxr-xr-x 1 nalukashov studsci    1077 окт 21  2020  lab07.asm
-rw-r--r-- 1 nalukashov studsci     187 мая 20 11:53  lab07.sh
-rw-r--r-- 1 nalukashov studsci     100 мая 20 10:52  lab07.sh~
-rw-r--r-- 1 nalukashov studsci     268 сен 17  2020  lab2.asm
drwxr-xr-x 3 nalukashov studsci    2048 апр 29 10:42  laboratory
-rw-r--r-- 1 nalukashov studsci     154 окт  1  2020  makefile
-rw-r--r-- 1 nalukashov studsci       0 мая 13 12:04  may
drwx--x--x 2 nalukashov studsci    2048 мая 13 11:57  monthly
-r-xr--r-- 1 nalukashov studsci       0 мая 13 12:37  my_os
drwx--x--x 3 nalukashov studsci    2048 мая 13 12:50  play
```

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



#Контрольные вопросы

1. Команда getopts является встроенной командой командной оболочки bash, предназначенной для разбора параметров сценариев. Она обрабатывает исключительно однобуквенные параметры как с аргументами, так и без

них и этого

вполне достаточно для передачи сценариям любых входных данных.

2. При генерации имен используют метасимволы:

- произвольная (возможно пустая) последовательность символов;

? один произвольный символ;

[...] любой из символов, указанных в скобках перечислением и/или с указанием диапазона;

cat f* выдаст все файлы каталога, начинающиеся с “f”;

cat *f* выдаст все файлы, содержащие “f”;

cat program.? выдаст файлы данного каталога с однобуквенными расширениями, скажем “program.c” и “program.o”, но не выдаст “program.com”;

cat [a-d]* выдаст файлы, которые начинаются с “a”, “b”, “c”, “d”. Аналогичный эффект дадут и команды “cat [abcd]*” u ”cat [bdac]*”.

3. Операторы && и || являются управляющими операторами. Если в командной строке стоит command1 && command2, то command2 выполняется в том, и только в том случае, если статус выхода из команды command1 равен нулю, что говорит об успешном ее завершении. Аналогично, если командная строка имеет вид command1 || command2, то команда command2 выполняется тогда, и только тогда, когда статус выхода из команды command1 отличен от нуля.

4. Оператор break завершает выполнение ближайшего включающего цикла или условного оператора, в котором он отображается.

5. Команда true всегда возвращает ноль в качестве выходного статуса для индикации успеха. Команда false всегда возвращает не-ноль в качестве выходного статуса для индикации неудачи. Во всех управляющих конструкциях в качестве логического значения используется код возврата из программы,

указанной в качестве условия. Код возврата 0 – истина, любое другое значение – ложь. Программа true – всегда завершается с кодом 0, false – всегда завершается с кодом 1.

6. Введенная строка означает условие существования файла man$s$/i.$s

7. Цикл While выполняется до тех пор, пока указанное в нем условие истинно. Когда указанное условие становится ложным - цикл завершается. Цикл Until выполняется до тех пор, пока указанное в нем условие ложно.

3 Выводы

Я изучил основы программирования в оболочке ОС UNIX, научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.