

Отчёт по лабораторной работе 11

Программирование в командном процессоре ОС Unix. Ветвление и
циклы.

Фомичева Маргарита Романовна

Содержание

1	Цель работы	6
2	Ход работы	7
	Используя команды <code>getops</code> <code>grep</code> , пишу командный файл, который анализирует командную строку с ключами:	7
	а затем ищет в указанном файле нужные строки, определяемые ключом <code>-p</code>	8
	Пишу на языке <code>c</code> программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции <code>exit(n)</code> , передавая информацию о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды <code>&?</code> , выдать сообщение о том, какое число было введено	11
	Пишу командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до <code>N</code> . Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).	13

Написала командный файл, который с помощью команды <code>tar</code> запаковывает в архив все файлы в указанной директории. Модифицирую командный файл так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использую команду <code>find</code>)	14
3 Вывод	16
4 Ответы на контрольные вопросы	17
Какого предназначение команды <code>getopts</code> ?	17
Какое отношение метасимволы имеют к генерации имён файлов? . .	17
Какие операторы управления действиями вы знаете?	18
Какие операторы используются для прерывания цикла?	18
Для чего нужны команды <code>false</code> и <code>true</code> ?	18
Что означает строка <code>if test -f mans/i</code> , встреченная в командной файле?	18
Объясните различия между конструкциями <code>while</code> и <code>until</code> ?	19

Список таблиц

Список иллюстраций

2.1	1	8
2.2	2	9
2.3	3	10
2.4	4	10
2.5	5	11
2.6	6	11
2.7	7	12
2.8	8	13
2.9	9	14
2.10	10	14
2.11	11	15
2.12	12	15

1 Цель работы

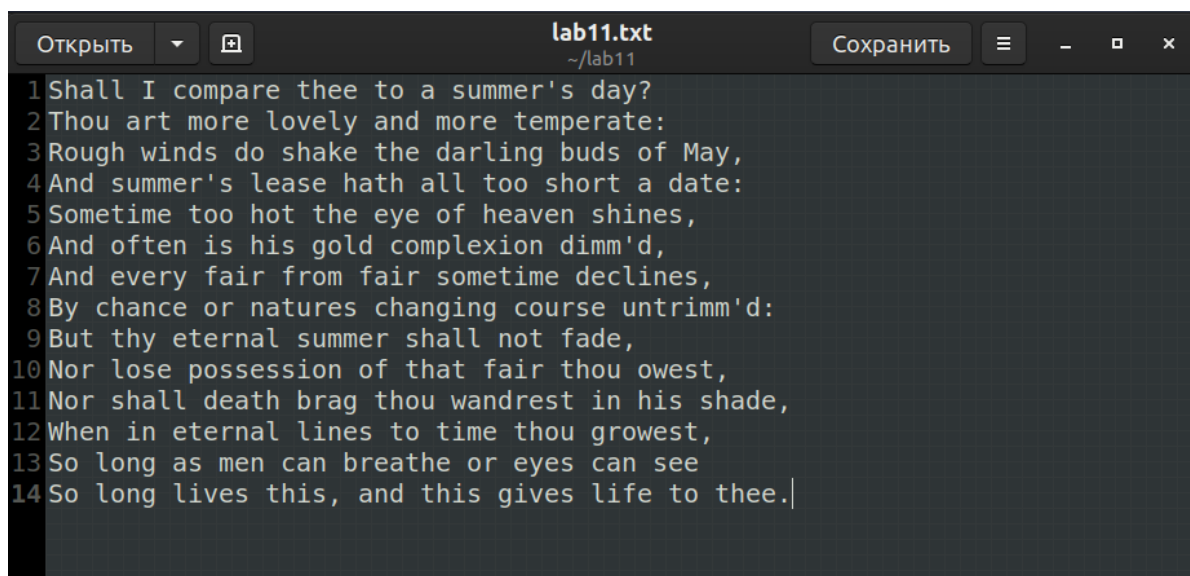
- Изучить основы программирования в оболочке ОС Unix. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

2 Ход работы

Используя команды `getops` `grep`, пишу 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,

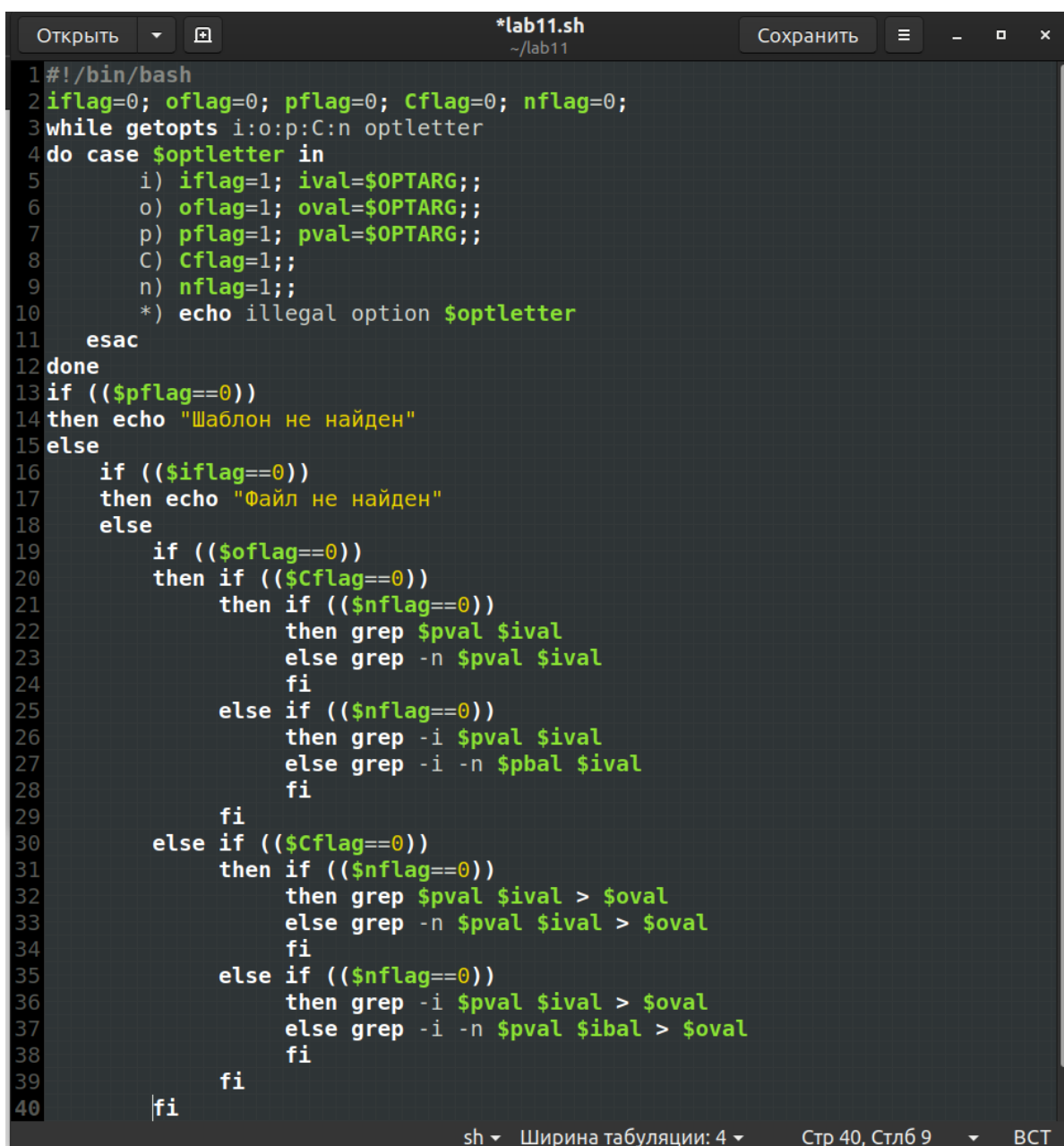
а затем ищет в указанном файле нужные строки, определяемые ключом -p



The image shows a screenshot of a text editor window. The title bar at the top reads "lab11.txt" and "~ /lab11". On the left side of the title bar are buttons for "Открыть" (Open) and a file icon. On the right side are buttons for "Сохранить" (Save), a menu icon (three horizontal lines), and standard window control icons (minimize, maximize, close). The main area of the window contains 14 lines of text, which is a sonnet by William Shakespeare. Each line is numbered from 1 to 14 on the left margin. The text is as follows:

```
1 Shall I compare thee to a summer's day?  
2 Thou art more lovely and more temperate:  
3 Rough winds do shake the darling buds of May,  
4 And summer's lease hath all too short a date:  
5 Sometime too hot the eye of heaven shines,  
6 And often is his gold complexion dimm'd,  
7 And every fair from fair sometime declines,  
8 By chance or natures changing course untrimm'd:  
9 But thy eternal summer shall not fade,  
10 Nor lose possession of that fair thou owest,  
11 Nor shall death brag thou wandrest in his shade,  
12 When in eternal lines to time thou growest,  
13 So long as men can breathe or eyes can see  
14 So long lives this, and this gives life to thee.
```

Рис. 2.1: 1



```
1#!/bin/bash
2iflag=0; oflag=0; pflag=0; cflag=0; nflag=0;
3while getopts i:o:p:C:n optletter
4do case $optletter in
5    i) iflag=1; ival=$OPTARG;;
6    o) oflag=1; oval=$OPTARG;;
7    p) pflag=1; pval=$OPTARG;;
8    C) cflag=1;;
9    n) nflag=1;;
10   *) echo illegal option $optletter
11   esac
12done
13if (($pflag==0))
14then echo "Шаблон не найден"
15else
16    if (($iflag==0))
17    then echo "Файл не найден"
18    else
19        if (($oflag==0))
20        then if (($cflag==0))
21            then if (($nflag==0))
22                then grep $pval $ival
23                else grep -n $pval $ival
24                fi
25            else if (($nflag==0))
26                then grep -i $pval $ival
27                else grep -i -n $pval $ival
28                fi
29            fi
30        else if (($cflag==0))
31            then if (($nflag==0))
32                then grep $pval $ival > $oval
33                else grep -n $pval $ival > $oval
34                fi
35            else if (($nflag==0))
36                then grep -i $pval $ival > $oval
37                else grep -i -n $pval $ival > $oval
38                fi
39            fi
40    fi
```

Рис. 2.2: 2

```
35         else if (($nflag==0))
36             then grep -i $pval $ival > $oval
37             else grep -i -n $pval $ibal > $oval
38             fi
39         fi
40     fi
41 fi
42 fi
```

sh ▾ Ширина табуляции: 4 ▾ Стр 40, Стлб 9 ▾ ВСТ

Рис. 2.3: 3

```
→ lab11 bash lab11.sh -ilab11.txt -olab11-1.txt -pice
→ lab11 cat lab11.txt
Shall I compare thee to a summer's day?
Thou art more lovely and more temperate:
Rough winds do shake the darling buds of May,
And summer's lease hath all too short a date:
Sometime too hot the eye of heaven shines,
And often is his gold complexion dimm'd,
And every fair from fair sometime declines,
By chance or natures changing course untrimm'd:
But thy eternal summer shall not fade,
Nor lose possession of that fair thou owest,
Nor shall death brag thou wandrest in his shade,
When in eternal lines to time thou growest,
So long as men can breathe or eyes can see
So long lives this, and this gives life to thee.
→ lab11 ./lab11.sh -i lab11.txt -o lab11-1.txt -p And -n
→ lab11 cat lab11-1.txt
4:And summer's lease hath all too short a date:
6:And often is his gold complexion dimm'd,
7:And every fair from fair sometime declines,
→ lab11 ./lab11.sh -i lab11.txt -C -n
Шаблон не найден
→ lab11 ./lab11.sh -o lab11-1.txt -p And -n
Файл не найден
→ lab11
```

Рис. 2.4: 4

Пишу на языке с 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 `&?`, выдать сообщение о том, какое число было введено



```
lab11-1.c
~/lab11
Открыть Сохранить
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     printf("Введите число: ");
6     int a;
7     scanf("%d",&a);
8     if (a<0) exit(0);
9     if (a>0) exit(1);
10    if (a==0) exit(2);
11    return 0;
12 }
```

Рис. 2.5: 5



```
lab11-2.sh
~/lab11
Открыть Сохранить
1 #!/bin/bash
2
3 gcc lab11-1.c -o lab11
4 ./lab11
5 code=$?
6 case $code in
7     0) echo "Число меньше 0";;
8     1) echo "Число больше 0";;
9     2) echo "Число равно 0";;
10 esac
```

Рис. 2.6: 6

```
→ lab11 touch lab11-1.c lab11-2.sh
→ lab11 chmod +x *.sh
→ lab11 ./lab11-2.sh
Введите число: 7
Число больше 0
→ lab11 ./lab11-2.sh
Введите число: 0
Число равно 0
→ lab11 ./lab11-2.sh
Введите число: -3
Число меньше 0
→ lab11
```

Рис. 2.7: 7

Пишу командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N. Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).



```
1#!/bin/bash
2
3opt=$1;
4form=$2;
5num=$3;
6function Files() {
7    for ((i=1; i<=$num; i++)) do
8        file=$(echo $form | tr '#' "$i")
9        if [ $opt == "-r" ]
10        then
11            rm -f $file
12        elif [ $opt == "-c" ]
13        then
14            touch $file
15        fi
16    done
17}
18Files|
```

Рис. 2.8: 8

```

→ lab11 touch lab11-3.sh
→ lab11 chmod +x *.sh
→ lab11 ls
lab11 lab11-1.c lab11-1.txt lab11-2.sh lab11-3.sh lab11.sh lab11.txt
→ lab11 ./lab11-3.sh -c text#.txt 8
→ lab11 ls
lab11 lab11-2.sh lab11.txt text3.txt text6.txt
lab11-1.c lab11-3.sh text1.txt text4.txt text7.txt
lab11-1.txt lab11.sh text2.txt text5.txt text8.txt
→ lab11 ./lab11-3.sh -r text#.txt 8
→ lab11 ls
lab11 lab11-1.c lab11-1.txt lab11-2.sh lab11-3.sh lab11.sh lab11.txt
→ lab11

```

Рис. 2.9: 9

Написала командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории.

Модифицирую командный файл так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использую команду find)

```

Открыть  *lab11-4.sh  Сохранить
~/lab11
1 #!/bin/bash
2
3 files=$(find ./ -maxdepth 1 -mtime -7)
4 listing=""
5 for file in "$files" ; do
6     file=$(echo "$file" | cut -c 3-)
7     listing="$listing $file"
8 done
9 dir=$(basename $(pwd))
10 tar -cvf $dir.tar $listing

```

Рис. 2.10: 10

```

→ lab11 touch lab11-4.sh
→ lab11 chmod +x *.sh
→ lab11 ls -l
итого 48
-rwxrwxr-x 1 bakha bakha 16848 мая 26 15:58 lab11
-rw-rw-r-- 1 bakha bakha 196 мая 26 15:57 lab11-1.c
-rw-rw-r-- 1 bakha bakha 137 мая 26 15:27 lab11-1.txt
-rwxrwxr-x 1 bakha bakha 195 мая 26 15:57 lab11-2.sh
-rwxrwxr-x 1 bakha bakha 235 мая 26 16:22 lab11-3.sh
-rwxrwxr-x 1 bakha bakha 210 мая 26 16:30 lab11-4.sh
-rwxrwxr-x 1 bakha bakha 1090 мая 26 14:56 lab11.sh
-rw-rw-r-- 1 bakha bakha 618 мая 26 15:25 lab11.txt

```

Рис. 2.11: 11

```

→ lab11 ./lab11-4.sh
lab11-1.txt
lab11.txt
lab11-2.sh
lab11-3.sh
lab11.sh
lab11-1.c
lab11-4.sh
lab11
→ lab11 tar -tf lab11.tar
lab11-1.txt
lab11.txt
lab11-2.sh
lab11-3.sh
lab11.sh
lab11-1.c
lab11-4.sh
lab11
→ lab11

```

Рис. 2.12: 12

3 Вывод

- Я изучила основы программирования в оболочке ОС Unix более подробно. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

4 Ответы на контрольные вопросы

Какого предназначение команды `getopts`?

- Команда `getopts` - это встроенная команда оболочки `bash`, которая предназначена для разбора параметров сценария. Команда обрабатывает только однобуквенные параметры и с аргументами, и без них.

Какое отношение метасимволы имеют к генерации имён файлов?

- 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*` выдаёт все файлы каталога, начинающиеся с `"f"`
- `cat f` выдаёт все файлы, содержащие `"f"`
- `cat program.?` выдаёт файлы данного каталога с однобуквенными расширениями, скажем `"program.c"` и `"program.o"`, но не выдаст `"program.com"`
- `cat [a-d]*` выдаёт файлы, которые начинаются с `"a"`, `"b"`, `"c"`, `"d"`. Такой же результат выдадут и команды `"cat [abcd]"` и `"cat [bdac]"`.

Какие операторы управления действиями вы знаете?

- Операторы `&&` и `||` это управляющие операторы. Если в командной строке стоит `command1 && command2`, то `command2` выполняется в том случае, если статус выхода из команды `command1` равен нулю, что говорит о ее успешном завершении. Так же, если командная строка имеет вид `command1 || command2`, то команда `command2` выполняется тогда, когда статус выхода из команды `command1` отличен от нуля.

Какие операторы используются для прерывания цикла?

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

Для чего нужны команды `false` и `true`?

- 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.

Что означает строка `if test -f mans/i`, встреченная в командной файле?

- Данная строка строка означает условие существования файла `mans/i.$s`

Объясните различия между конструкциями while и until?

- 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.