

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Чернов Иван Сергеевич¹

18 апреля, 2023, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

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

Задачи лабораторной работы

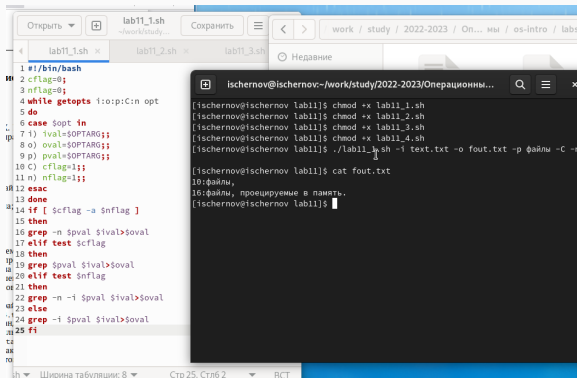
1 Выполнить 4 задания

Процесс выполнения лабораторной работы

1. Используя команды `getopts` `grep` напишем командный файл, который анализирует командную строку с ключами и выполним его: `-i inputfile` — прочитать данные из указанного файла; `-o outputfile` — вывести данные в указанный файл; `-r шаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк;

а затем ищет в указанном файле нужные строки

Выполнение работы



The image shows a code editor window with a file named `lab11_1.sh` open. The script contains a `while` loop that reads command-line options and a `case` statement that sets flags based on the options. The script also includes `grep` commands to check for specific flags. A terminal window is overlaid on the code editor, showing the execution of the script. The terminal output shows the script being run with various flags, and the output of the `cat` command is displayed.

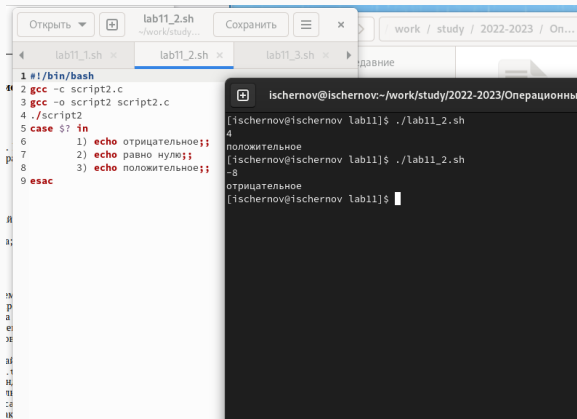
```
1 #!/bin/bash
2 cflag=0;
3 nflag=0;
4 while getopts i:o:p:C:n opt
5 do
6 case $opt in
7 i) ival=$OPTARG;;
8 o) oval=$OPTARG;;
9 p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $ival>$oval
17 elif test $cflag
18 then
19 grep $pval $ival>$oval
20 elif test $nflag
21 then
22 grep -n -i $pval $ival>$oval
23 else
24 grep -i $pval $ival>$oval
25 fi
```

```
[ischernov@ischernov lab11]$ chmod +x lab11_1.sh
[ischernov@ischernov lab11]$ chmod +x lab11_2.sh
[ischernov@ischernov lab11]$ chmod +x lab11_3.sh
[ischernov@ischernov lab11]$ chmod +x lab11_4.sh
[ischernov@ischernov lab11]$ ./lab11_1.sh -i text.txt -o fout.txt -p файлы -C -n
[ischernov@ischernov lab11]$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
[ischernov@ischernov lab11]$
```

Рис. 1: Задание 1

2. Напишем сначала на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем завершим программу при помощи функции `exit(n)`, передавая информацию о коде завершения в оболочку. Командный файл вызовет эту программу и, проанализировав с помощью команды `$?`, выдаст сообщение о том, какое число было введено

Выполнение работы



The image shows a code editor window with a file named `lab11_2.sh` and a terminal window running the script.

Code Editor Content:

```
1 #!/bin/bash
2 gcc -c script2.c
3 gcc -o script2 script2.c
4 ./script2
5 case $? in
6     1) echo отрицательное;;
7     2) echo равно нулю;;
8     3) echo положительное;;
9 esac
```

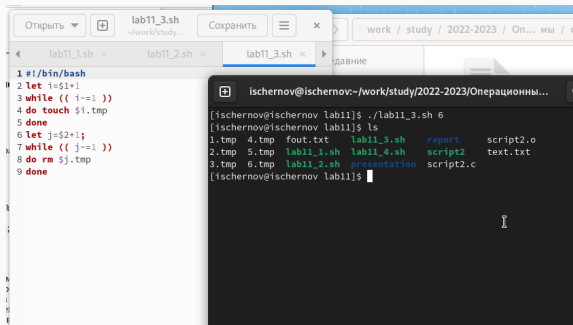
Terminal Content:

```
ischernov@ischernov:~/work/study/2022-2023/Операционны
[ischernov@ischernov lab11]$ ./lab11_2.sh
4
положительное
[ischernov@ischernov lab11]$ ./lab11_2.sh
-8
отрицательное
[ischernov@ischernov lab11]$
```

Рис. 2: Задание 2

3. Напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N

Выполнение работы



The image shows a code editor window with a file named `lab11_3.sh` and a terminal window running the script. The code editor has tabs for `lab11_1.sh`, `lab11_2.sh`, and `lab11_3.sh`. The terminal window title is `ischernov@ischernov:~/work/study/2022-2023/Операционны...`.

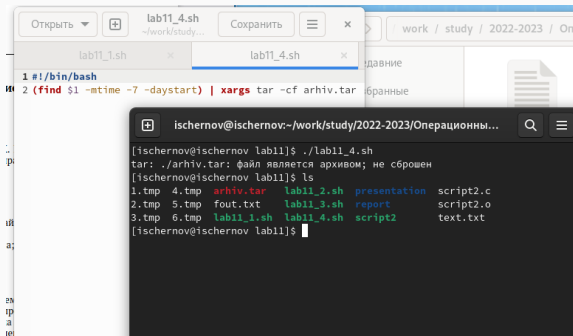
```
1 #!/bin/bash
2 let i=$1+1
3 while (( i-->0 ))
4 do touch $i.tmp
5 done
6 let j=$2+1;
7 while (( j-->0 ))
8 do rm $j.tmp
9 done
```

```
[ischernov@ischernov lab11]$ ./lab11_3.sh 6
[ischernov@ischernov lab11]$ ls
1.tmp 4.tmp  fout.txt  lab11_3.sh  report      script2.o
2.tmp 5.tmp  lab11_1.sh  lab11_4.sh  script2     text.txt
3.tmp 6.tmp  lab11_2.sh  presentation  script2.c
```

Рис. 3: Задание 3

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.

Выполнение работы



The screenshot shows a terminal window with a light gray title bar. The title bar contains the text "lab11_4.sh" and the path "~/work/study...". Below the title bar, the terminal displays the following commands and output:

```
1 #!/bin/bash
2 (find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar
```

The terminal window is overlaid on a file manager window. The file manager window shows a directory structure with the path "work / study / 2022-2023 / Операционны...". The file manager window also shows a list of files and folders, including "архивные" and "бронные".

The terminal window shows the following output:

```
[ischernov@ischernov lab11]$ ./lab11_4.sh
tar: ./arhiv.tar: файл является архивом; не сброшен
[ischernov@ischernov lab11]$ ls
1.tmp  4.tmp  arhiv.tar  lab11_2.sh  presentation  script2.c
2.tmp  5.tmp  fout.txt  lab11_3.sh  report        script2.o
3.tmp  6.tmp  lab11_1.sh  lab11_4.sh  script2       text.txt
[ischernov@ischernov lab11]$
```

Рис. 4: Задание 4

Выводы по проделанной работе

В данной работе мы изучили основы программирования в оболочке ОС UNIX и писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.