

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Павел Межинский¹

21 апреля, 2024, 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

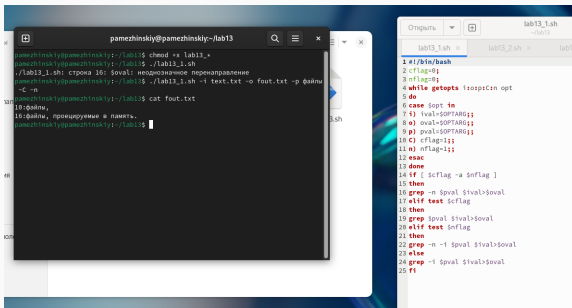
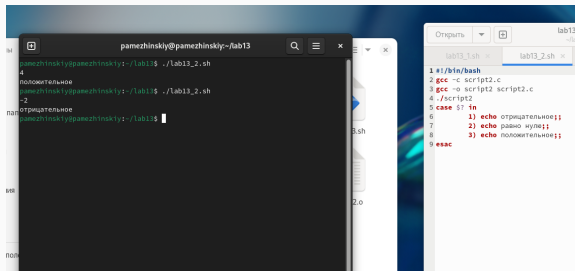


Рис. 1: Задание 1

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 terminal window and a code editor. The terminal window, titled 'pamezhinskiy@pamezhinskiy:~/lab13', shows the execution of a script named 'lab13_2.sh'. The script's output is as follows:

```
pamezhinskiy@pamezhinskiy:~/lab13$ ./lab13_2.sh
4
положительное
pamezhinskiy@pamezhinskiy:~/lab13$ ./lab13_2.sh
-2
отрицательное
pamezhinskiy@pamezhinskiy:~/lab13$
```

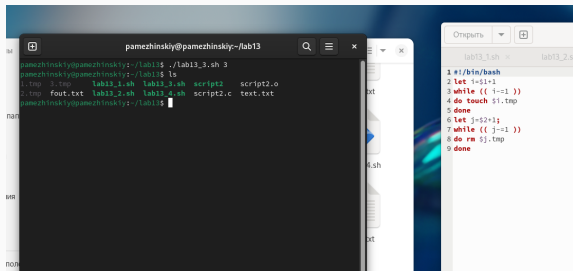
The code editor, titled 'lab13_2.sh', shows the source code of the script:

```
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
```

Рис. 2: Задание 2

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

Выполнение работы



The image shows a terminal window and a file explorer. The terminal window, titled 'pamezhinskiy@pamezhinskiy:~/lab13', displays the following commands and output:

```
pamezhinskiy@pamezhinskiy:~/lab13$ ./lab13_3.sh 3
pamezhinskiy@pamezhinskiy:~/lab13$ ls
1.tmp 3.tmp lab13_1.sh lab13_3.sh script2 script2.o
2.tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
pamezhinskiy@pamezhinskiy:~/lab13$
```

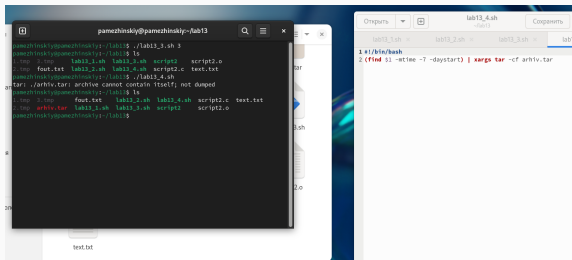
The file explorer shows two files: 'lab13_1.sh' and 'lab13_2.sh'. The content of 'lab13_1.sh' is as follows:

```
1 #!/bin/bash
2 let i=51+1
3 while (( i-->1 ))
4 do touch $i.tmp
5 done
6 let j=52+1;
7 while (( j-->1 ))
8 do rm $j.tmp
9 done
```

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 image displays two terminal windows from a Linux environment. The left window, titled 'pamezhinskiy@pamezhinskiy:~/lab13', shows a series of commands and their outputs. It starts with running './lab13_3.sh 3', followed by 'ls' which lists files in the current directory. Then, it runs a loop: '1. tmp 2. tmp lab13_1.sh lab13_2.sh script3 script2.o', followed by 'fout.txt lab13_3.sh lab13_4.sh script2.c text.txt'. This is followed by another loop: 'pamezhinskiy@pamezhinskiy:~/lab13\$./lab13_4.sh', which outputs 'tar: ./arhiv.tar: archive cannot contain itself; not dumped'. Finally, it runs '1. tmp 2. tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt' and '2. tmp arhiv.tar lab13_1.sh lab13_3.sh script2 script2.o', resulting in 'pamezhinskiy@pamezhinskiy:~/lab13\$'. The right window, titled 'lab13_4.sh', shows a prompt '1 vi/bin/bash' and a command '2 (find \$1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar'.

```
pamezhinskiy@pamezhinskiy:~/lab13$ ./lab13_3.sh 3
pamezhinskiy@pamezhinskiy:~/lab13$ ls
1. tmp 2. tmp lab13_1.sh lab13_2.sh script3 script2.o
fout.txt lab13_3.sh lab13_4.sh script2.c text.txt
pamezhinskiy@pamezhinskiy:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
pamezhinskiy@pamezhinskiy:~/lab13$ ls
1. tmp 2. tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
2. tmp arhiv.tar lab13_1.sh lab13_3.sh script2 script2.o
pamezhinskiy@pamezhinskiy:~/lab13$

lab13_4.sh
1 vi/bin/bash
2 (find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar
```

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.