

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Алина Тарасова¹

19 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

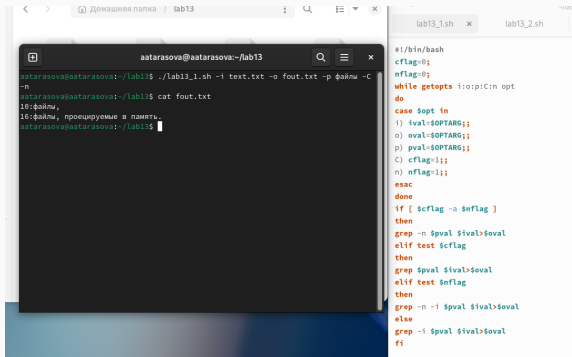
1 Выполнить 4 задания

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



The image shows a terminal window and a script file. The terminal window, titled 'aatarasova@aatarasova:~/lab13', displays the execution of a script named 'lab13_1.sh'. The script takes 'text.txt' as input and produces 'fout.txt' as output. The output of the script is displayed in the terminal, showing the contents of 'fout.txt' which are: '10:файлы,' and '16:файлы, проецируемые в память.' The script file, titled 'lab13_1.sh', is shown on the right. It is a bash script that sets flags 'cflag' and 'nflag' to 0. It then enters a 'while' loop that reads options from 'text.txt' and sets flags accordingly. The script then checks if the flags are set and performs a series of 'grep' operations on 'fout.txt' based on the flags.

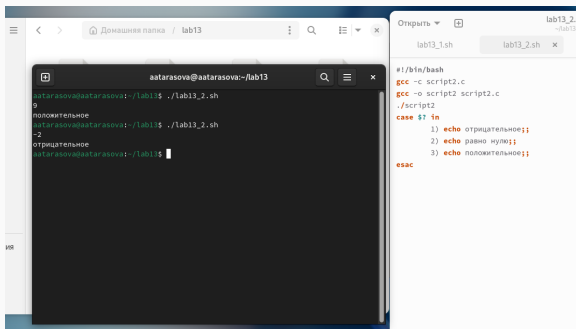
```
aatarasova@aatarasova:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -c -n
aatarasova@aatarasova:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
aatarasova@aatarasova:~/lab13$
```

```
#!/bin/bash
cflag=0;
nflag=0;
while getopts i:o:p:C:n opt
do
case $opt in
i) ival=$OPTARG;;
o) oval=$OPTARG;;
p) pval=$OPTARG;;
C) cflag=1;;
n) nflag=1;;
esac
done
if [ $cflag -a $nflag ]
then
grep -n $pval $ival>$oval
elif test $cflag
then
grep $pval $ival>$oval
elif test $nflag
then
grep -n -i $pval $ival>$oval
else
grep -i $pval $ival>$oval
fi
```

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 desktop environment with a terminal window and a code editor. The terminal window, titled 'aatarasova@aatarasova: ~/lab13', displays the execution of a script 'lab13_2.sh' which outputs '0' (positive) and '-2' (negative). The code editor, titled 'lab13_2.sh', shows the script's content, which includes a case statement for checking if a number is negative, zero, or positive.

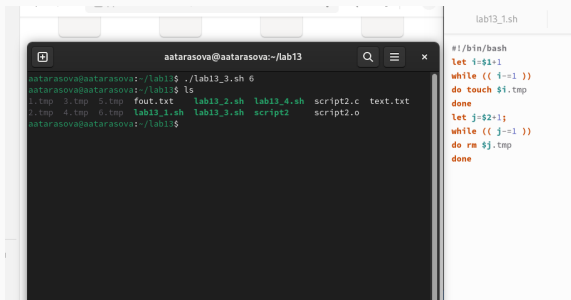
```
aatarasova@aatarasova: ~/lab13
aatarasova@aatarasova: ~/lab13$ ./lab13_2.sh
0
положительное
aatarasova@aatarasova: ~/lab13$ ./lab13_2.sh
-2
отрицательное
aatarasova@aatarasova: ~/lab13$
```

```
#!/bin/bash
gcc -c script2.c
gcc -o script2 script2.c
./script2
case $? in
  1) echo отрицательное;;
  2) echo равно нулю;;
  3) echo положительное;;
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 code editor. The terminal window, titled 'aatarasova@aatarasova:~/lab13', displays the execution of a script 'lab13_3.sh' with argument '6'. The output of the script is a directory listing of files in the current directory. The code editor, titled 'lab13_1.sh', shows the source code of the script, which is a bash script that iterates over a range of numbers and performs file operations.

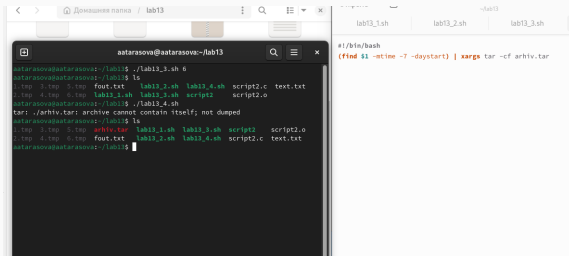
```
aatarasova@aatarasova:~/lab13$ ./lab13_3.sh 6
aatarasova@aatarasova:~/lab13$ ls
1.tmp  3.tmp  5.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
2.tmp  4.tmp  6.tmp  lab13_1.sh  lab13_3.sh  script2    script2.o
aatarasova@aatarasova:~/lab13$
```

```
#!/bin/bash
let i=$1+1
while (( i--=1 ))
do touch $i.tmp
done
let j=$2+1;
while (( j--=1 ))
do rm $j.tmp
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 shows a terminal window and a file manager side-by-side. The terminal window, titled 'aatarasova@aatarasova: ~/lab13', displays the following commands and output:

```
aatarasova@aatarasova:~/lab13$ ./lab13_3.sh 6
aatarasova@aatarasova:~/lab13$ ls
1.tsp 3.tsp 5.tsp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
2.tsp 4.tsp 6.tsp lab13_1.sh lab13_3.sh script2 script2.o
aatarasova@aatarasova:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
aatarasova@aatarasova:~/lab13$ ls
1.tsp 3.tsp 5.tsp arhiv.tar lab13_1.sh lab13_3.sh script2 script2.o
2.tsp 4.tsp 6.tsp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
aatarasova@aatarasova:~/lab13$
```

The file manager window, titled 'Домашняя папка / lab13', shows three tabs: 'lab13_1.sh', 'lab13_2.sh', and 'lab13_3.sh'. The 'lab13_3.sh' tab is active and displays the following code:

```
#!/bin/bash
(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.