

Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Ярослав Шищенко

18 апреля 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

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

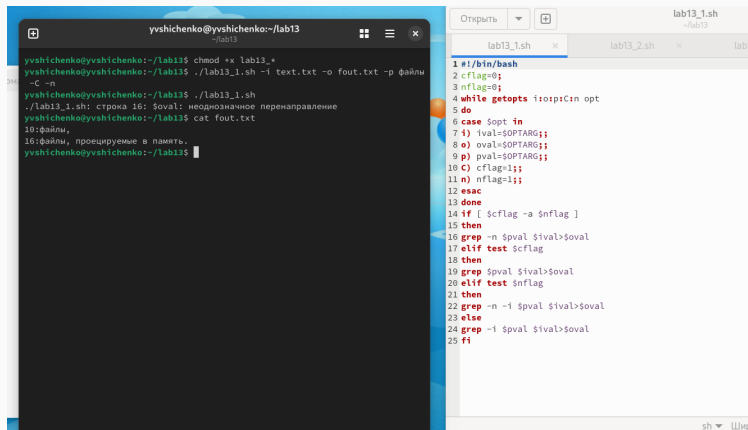
1 Выполнить 4 задания

Процесс выполнения лабораторной работы

1. Используя команды `getopts` `grep` напомним 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; `-p 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 on the left and a code editor on the right. The terminal window, titled 'yvshichenko@yvshichenko:~/lab13', shows the execution of a script 'lab13_1.sh'. The user runs 'chmod +x lab13_*', then './lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n'. The script outputs '16: строка 16: \$oval: неоднозначное перенаправление' and '16: файлы, проецируемые в память.' before returning to the prompt. The code editor, titled 'lab13_1.sh', shows the script's content, which is a shell script using 'getopts' to parse command-line options and 'grep' to search for patterns in input files.

```
yvshichenko@yvshichenko:~/lab13$ chmod +x lab13_*
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы
-C -n
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
yvshichenko@yvshichenko:~/lab13$ cat fout.txt
16:файлы,
16:файлы, проецируемые в память.
yvshichenko@yvshichenko:~/lab13$
```

```
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
```

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

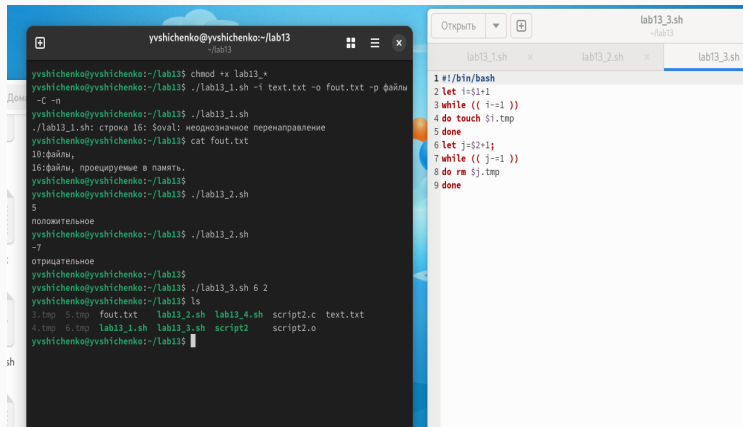
The image shows a terminal window on the left and a code editor on the right. The terminal window, titled 'yvshichenko@yvshichenko:~/lab13', shows the execution of a shell script 'lab13_1.sh' which sets up a directory 'fout.txt' and runs 'cat fout.txt'. The output of 'cat' shows a list of files and their sizes. The terminal then shows the execution of 'lab13_2.sh', which outputs '5', 'положительное', and '-7'. The code editor, titled 'lab13_2.sh', shows the source code of the script, which includes comments in Russian and a case statement for handling input.

```
yvshichenko@yvshichenko:~/lab13$ chmod +x lab13_*
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы
-C -n
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
yvshichenko@yvshichenko:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
5
положительное
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
-7
отрицательное
yvshichenko@yvshichenko:~/lab13$
```

```
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



The image shows a terminal window on the left and a file editor on the right. The terminal window, titled 'yvshichenko@yvshichenko:~/lab13', displays the execution of several shell scripts: 'lab13_1.sh', 'lab13_2.sh', and 'lab13_3.sh'. The output includes file permissions, file creation, and directory listings. The file editor, titled 'lab13_3.sh', shows the source code of the 'lab13_3.sh' script, which is a bash script that uses 'let' and 'while' loops to create and remove temporary files.

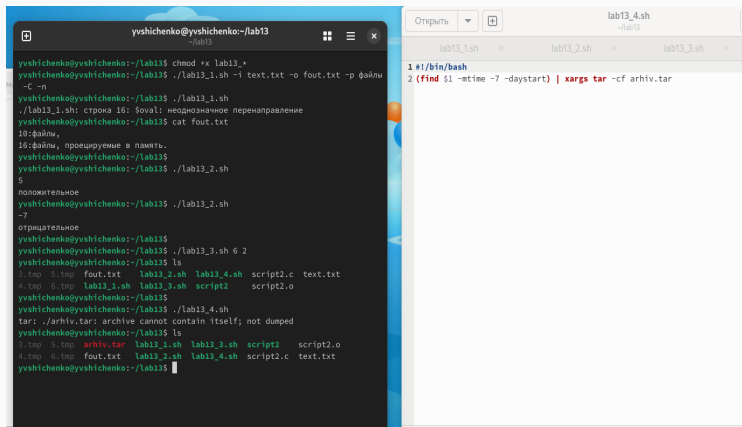
```
yvshichenko@yvshichenko:~/lab13$ chmod +x lab13_*
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы
-C -п
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
yvshichenko@yvshichenko:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
5
положительное
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
-7
отрицательное
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_3.sh 6 2
yvshichenko@yvshichenko:~/lab13$ ls
3.tmp 5.tmp fout.txt lab13_2.sh lab13_4.sh script2.c text.txt
4.tmp 6.tmp lab13_1.sh lab13_3.sh script2 script2.o
yvshichenko@yvshichenko:~/lab13$
```

```
#!/bin/bash
2 let i=$1+1
3 while (( i-=1 ))
4 do touch $i.tmp
5 done
6 let j=$2+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 shows a terminal window on the left and a file editor on the right. The terminal window, titled 'yvshichenko@yvshichenko:~/lab13', shows the execution of several shell scripts (lab13_1.sh, lab13_2.sh, lab13_3.sh, lab13_4.sh) and the creation of an archive named 'arhiv.tar'. The file editor, titled 'lab13_4.sh', shows the content of the script, which includes a shebang line and a command to create an archive using 'xargs tar'.

```
yvshichenko@yvshichenko:~/lab13
yvshichenko@yvshichenko:~/lab13$ chmod +x lab13.*
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh -f text.txt -o fout.txt -p файлы
-C -n
yvshichenko@yvshichenko:~/lab13$ ./lab13_1.sh
./lab13_1.sh: строка 16: $oval: неоднозначное перенаправление
yvshichenko@yvshichenko:~/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
5
положительное
yvshichenko@yvshichenko:~/lab13$ ./lab13_2.sh
-7
отрицательное
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_3.sh 6 2
yvshichenko@yvshichenko:~/lab13$ ls
3.tmp 5.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
4.tmp 6.tmp  lab13_1.sh  lab13_3.sh  script2    script2.o
yvshichenko@yvshichenko:~/lab13$
yvshichenko@yvshichenko:~/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
yvshichenko@yvshichenko:~/lab13$ ls
3.tmp 5.tmp  arhiv.tar  lab13_1.sh  lab13_3.sh  script2    script2.o
4.tmp 6.tmp  fout.txt  lab13_2.sh  lab13_4.sh  script2.c  text.txt
yvshichenko@yvshichenko:~/lab13$
```

```
lab13_4.sh
1 #!/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.