

Операционные системы

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Вишняков Родион Сергеевич

19 июня 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.

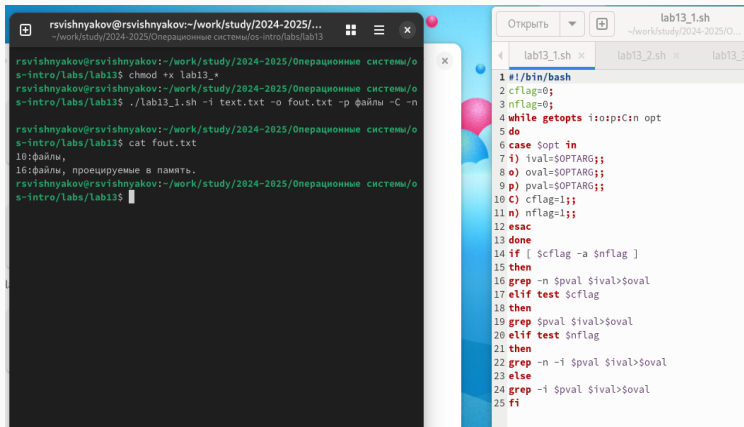
1 Выполнить 4 задания

Процесс выполнения лабораторной работы

1. Используя команды `getopts` `grep` 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; `-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 script file on the right. The terminal window has a title bar with the user 'rsvishnyakov' and the path '~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13'. The terminal output shows the user running 'chmod +x lab13_*', then './lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n', and finally 'cat fout.txt'. The output of the script is '10:файлы,' and '16:файлы, проецируемые в память.' The script file on the right is titled 'lab13_1.sh' and contains a shell script for processing command-line options.

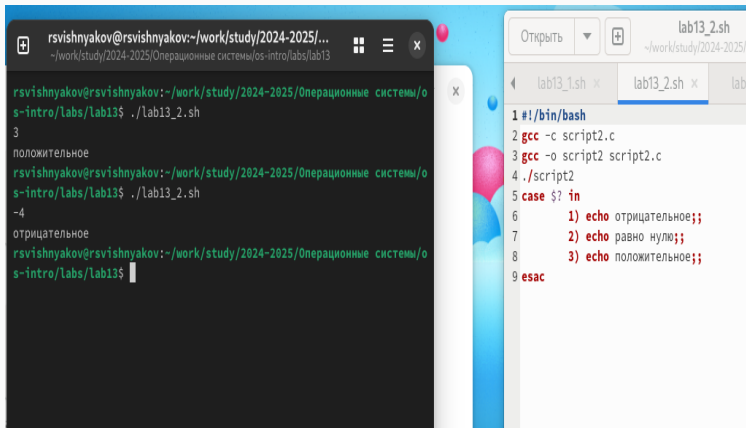
```
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ chmod +x lab13_*
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/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

Выполнение работы



The image shows a terminal window on the left and a code editor on the right. The terminal window displays the execution of a script named `lab13_2.sh`. The user runs the script twice, first with a positive argument and then with a negative argument. The code editor shows the source code of `lab13_2.sh`, which is a shell script that compiles a C program, runs it, and uses a `case` statement to print different messages based on the input argument.

```
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/...  
~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13  
  
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/o  
s-intro/labs/lab13$ ./lab13_2.sh  
3  
положительное  
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/o  
s-intro/labs/lab13$ ./lab13_2.sh  
-4  
отрицательное  
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/o  
s-intro/labs/lab13$
```

```
lab13_2.sh  
1#!/bin/bash  
2gcc -c script2.c  
3gcc -o script2 script2.c  
4./script2  
5case $? in  
6    1) echo отрицательное;;  
7    2) echo равно нулю;;  
8    3) echo положительное;;  
9esac
```

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 has a title bar with the user 'rsvishnyakov' and the path '~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13'. The terminal output shows the execution of 'lab13_3.sh' with argument '2', followed by a 'ls' command listing files in the current directory. The file editor on the right has a title bar for 'lab13_3.sh' and shows the contents of the script, which includes a loop to create and then remove temporary files.

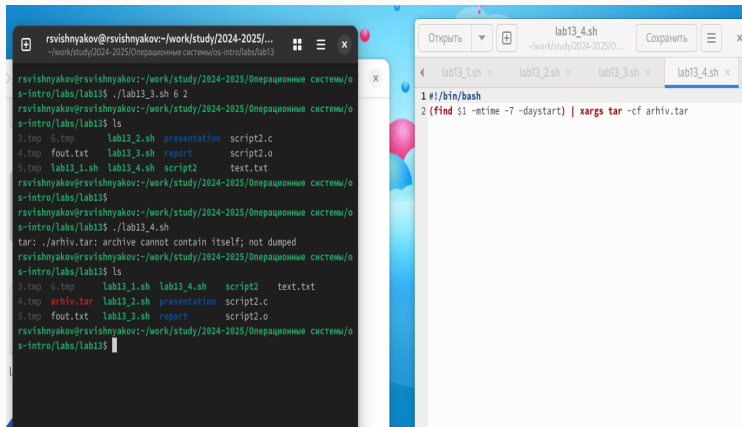
```
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ./lab13_3.sh 2
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$ ls
3.tmp  6.tmp      lab13_2.sh  presentation  script2.c
4.tmp  fout.txt   lab13_3.sh  report        script2.o
5.tmp  lab13_1.sh lab13_4.sh  script2      text.txt
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab13$
```

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

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 two overlapping windows from a Linux desktop environment. The background window is a terminal with the following content:

```
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$ ./lab13_3.sh 6 2
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$ ls
3.tmp  6.tmp      lab13_2.sh  presentation  script2.c
4.tmp  fout.txt    lab13_3.sh  report        script2.o
5.tmp  lab13_1.sh  lab13_4.sh  script2       text.txt
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$ ./lab13_4.sh
tar: ./arhiv.tar: archive cannot contain itself; not dumped
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$ ls
3.tmp  6.tmp      lab13_1.sh  lab13_4.sh  script2  text.txt
4.tmp  arhiv.tar  lab13_2.sh  presentation  script2.c
5.tmp  fout.txt  lab13_3.sh  report        script2.o
rsvishnyakov@rsvishnyakov:~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab13$
```

The foreground window is a text editor titled "lab13_4.sh" showing the script content:

```
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.