

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Гафоров Нурмухаммад Вомикович¹

20 июня, 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

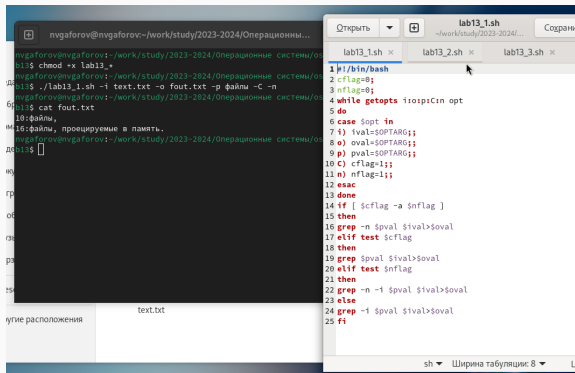
1 Выполнить 4 задания

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; `-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 and a code editor. The terminal window on the left displays the execution of a shell script named `lab13_1.sh`. The user `nvgafarov` is in the directory `~/work/study/2023-2024/Операционные системы/ос`. The script performs the following actions:

- Creates a file `lab13_1` with `chmod +x lab13_1`.
- Runs `./lab13_1.sh -i text.txt -o fout.txt -p файлы -C -n`.
- Displays the contents of `fout.txt` using `cat fout.txt`.

The output of the script is:

```
10:файлы,  
16:файлы, проецируемые в память.  
nvgafarov@nvgafarov:~/work/study/2023-2024/Операционные системы/ос  
Д0:1$
```

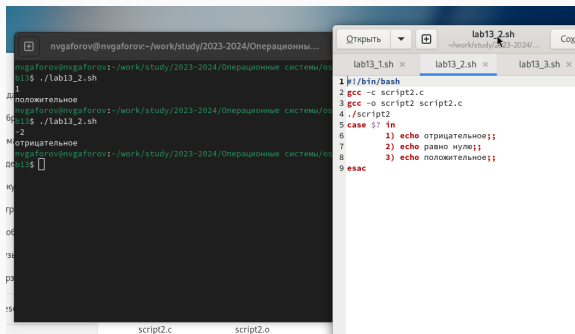
The code editor on the right shows the source code of `lab13_1.sh`. The code is a shell script that uses `getopts` to parse command-line options and `grep` to search for patterns in the input file.

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

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 'nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...', displays the execution of a script 'lab13_2.sh'. The script's output is as follows:

```
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...
lab13$ ./lab13_2.sh
1
положительное
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...
lab13$ ./lab13_2.sh
-2
Отрицательное
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...
lab13$
```

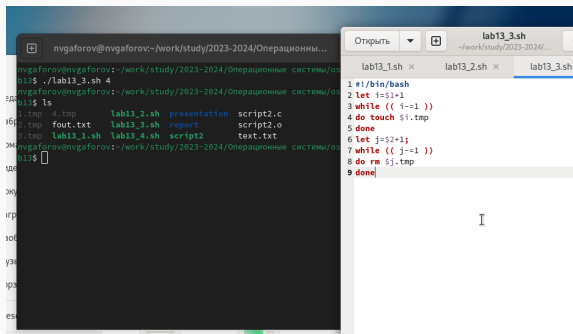
The code editor, titled 'lab13_2.sh', shows the script's source code:

```
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 code editor. The terminal window, titled 'nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...', displays the execution of a shell script 'lab13_3.sh'. The script's output is as follows:

```
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны...
b13$ ./lab13_3.sh 4
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны/о...
b13$ ls
1.tmp      4.tmp      lab13_2.sh  presentation  script2.c
2.tmp      fout.txt   lab13_3.sh  report        script2.o
3.tmp      lab13_1.sh lab13_4.sh  script2       text.txt
nvgaforov@nvgaforov:~/work/study/2023-2024/Операционны/о...
b13$
```

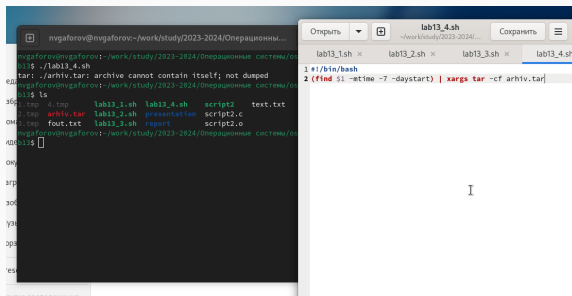
The code editor, titled 'lab13_3.sh', shows the following script code:

```
1 #!/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 two overlapping windows. The background window is a terminal with the prompt `nvgaforov@nvgaforov:~/work/study/2023-2024/Операционные системы/оп`. It displays the output of several commands: `ls` (listing files in the current directory), `tar: ./arhiv.tar: archive cannot contain itself; not dumped`, and `ls` (listing files in the `arhiv` directory). The foreground window is a file manager titled `lab13_4.sh` with the path `~/work/study/2023-2024/...`. It shows a list of files: `lab13_1.sh`, `lab13_2.sh`, `lab13_3.sh`, and `lab13_4.sh`. The `lab13_4.sh` file is selected, and its contents are displayed: `1 #!/bin/bash` and `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.