

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Камолов Жахонгир Джафарович¹

19 мая, 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

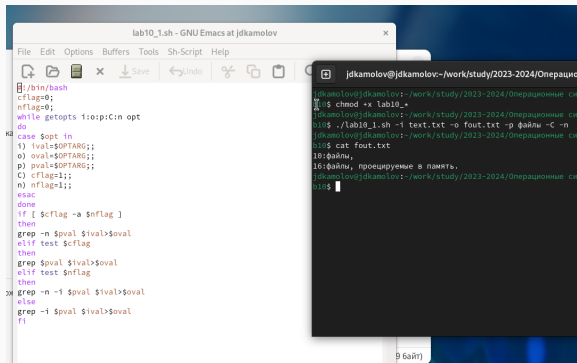
1 Выполнить 4 задания

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 GNU Emacs editor window titled "lab10_1.sh - GNU Emacs at jdkamolov" with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Help) and a toolbar. The editor contains a shell script with the following content:

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

Overlaid on the Emacs window is a terminal window titled "jdkamolov@jdkamolov:~/work/study/2023-2024/Операционные системы". The terminal shows the following commands and output:

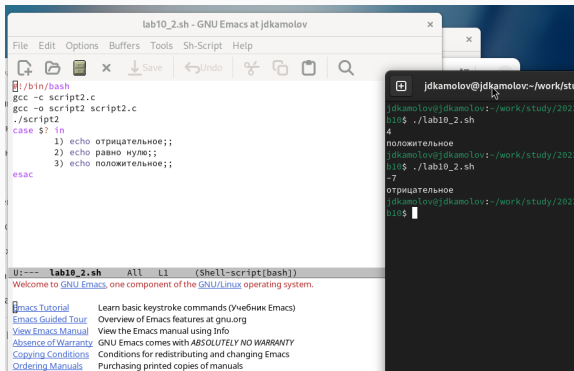
```
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционные системы$
jdkamolov$ chmod +x lab10_1.sh
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционные системы$ ./lab10_1.sh -i text.txt -o fout.txt -p файлы -C -n
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционные системы$ cat fout.txt
10:файлы,
16:файлы, проецируемые в память.
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционные системы$
```

At the bottom right of the terminal window, a status bar indicates "9 байт".

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 two overlapping windows. The background window is GNU Emacs, titled 'lab10_2.sh - GNU Emacs at jdkamolov'. It contains a shell script with the following content:

```
#!/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
```

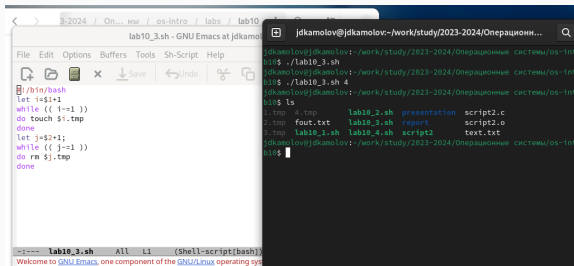
The foreground window is a terminal titled 'jdkamolov@jdkamolov:~/work/stu'. It shows the execution of the script 'lab10_2.sh' and its output:

```
jdkamolov@jdkamolov:~/work/study/2023
b10$ ./lab10_2.sh
4
положительное
jdkamolov@jdkamolov:~/work/study/2023
b10$ ./lab10_2.sh
-7
отрицательное
jdkamolov@jdkamolov:~/work/study/2023
b10$
```

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 an Emacs editor. The terminal window, titled 'jdkamolov@jdkamolov:~/work/study/2023-2024/Операционн...', displays the execution of a shell script 'lab10_3.sh'. The script's output is as follows:

```
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционн...$ ./lab10_3.sh
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционн...$ ls
1.tmp  4.tmp      lab10_2.sh  presentation  script2.c
2.tmp  fout.txt   lab10_3.sh  report        script2.o
3.tmp  lab10_1.sh lab10_4.sh  script2       text.txt
jdkamolov@jdkamolov:~/work/study/2023-2024/Операционн...$
```

The Emacs editor, titled 'lab10_3.sh - GNU Emacs at jdkamolov', shows the source code of the script 'lab10_3.sh' in the 'lab10' directory. The code is a shell script that uses a while loop to create files and a for loop to remove them.

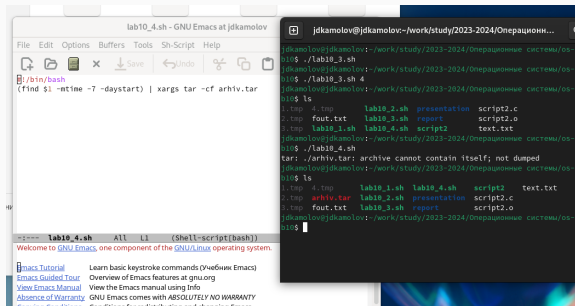
```
#!/bin/bash
let i=$1+1
while (( i-- ))
do touch $i.tmp
done
let j=$2+1;
while (( j-- ))
do rm $j.tmp
done
```

The status bar at the bottom of the Emacs window indicates the file is 'lab10_3.sh', it is 'All' (no changes), and it is a 'Shell-script(bash)'.

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 terminal windows. The left window is titled 'lab10_4.sh - GNU Emacs at jdkamolv' and shows the execution of a shell script. The script runs `(find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar` in a `bin/bash` environment. Below the script execution, there is a footer with links to Emacs documentation and a disclaimer: 'GNU Emacs comes with ABSOLUTELY NO WARRANTY'.

```
lab10_4.sh - GNU Emacs at jdkamolv
File Edit Options Buffers Tools Sh-Script Help
[Icons] Save Undo [Icons]
~/bin/bash
(find $1 -mtime -7 -daystart) | xargs tar -cf arhiv.tar

--== lab10_4.sh All LI (Shell-script{bash})
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

[Icons] Emacs Tutorial Learn basic keystroke commands (Учебник Emacs)
[Icons] Emacs Guided Tour Overview of Emacs features at gnu.org
[Icons] View Emacs Manual View the Emacs manual using Info
[Icons] Absence of Warranty GNU Emacs comes with ABSOLUTELY NO WARRANTY
[Icons] Emacs Conditions Conditions for redistribution and changing Emacs
```

The right window is titled 'jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...' and shows the output of the script. It lists the contents of the 'arhiv.tar' archive, which includes files from 'lab10_1.sh' through 'lab10_4.sh'.

```
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$ ./lab10_3.sh
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$ ./lab10_3.sh 4
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$ ls
1.tmp 4.tmp lab10_2.sh presentation script2.c
2.tmp fout.txt lab10_3.sh report script2.o
3.tmp lab10_1.sh lab10_4.sh script2 text.txt
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$ tar:
tar: ./arhiv.tar: archive cannot contain itself; not dumped
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$ ls
1.tmp 4.tmp lab10_1.sh lab10_4.sh script2 text.txt
2.tmp arhiv.tar lab10_2.sh presentation script2.c
3.tmp fout.txt lab10_3.sh report script2.o
jdkamolv@jdkamolv:~/work/study/2023-2024/Операционн...$
```

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.