

Презентация к лабораторной работе №11

Егорова Юлия Владимировна

¹RUDN University, Moscow, Russian Federation

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

1)Используя команды `getopts` `grep`, я написала командный файл, который анализирует командную строку с ключами:

-iinputfile — прочитать данные из указанного файла;

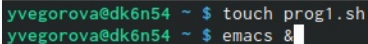
-ooutputfile - вывести данные в указанный файл;

-ршаблон — указать шаблон для поиска;

-C — различать большие и малые буквы;—

-n — выдавать номера строк.

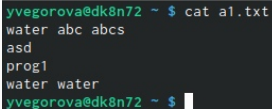
Затем нашла в указанном файле нужные строки, определяемые ключом -p, и создада файл prog1.sh, написав соответствующий скрипт:

A terminal window with a dark background and light green text. The prompt is 'yvegorova@dk6n54 ~ \$'. The first command is 'touch prog1.sh' and the second is 'emacs &'.

```
yvegorova@dk6n54 ~ $ touch prog1.sh  
yvegorova@dk6n54 ~ $ emacs &
```

Figure 1: Создание файла

2)Проверила работу написанного скрипта, предварительно создав файлы a1.txt и a2.txt. В a1.txt записала любой набор слов, а также дала доступ на исполнение файла:

A terminal window with a dark background. The prompt is 'yvegorova@dk8n72 ~ \$'. The command 'cat a1.txt' has been entered and executed. The output consists of five lines: 'water abc abcs', 'asd', 'prog1', 'water water', and a blank line. The prompt is now 'yvegorova@dk8n72 ~ \$' followed by a cursor.

```
yvegorova@dk8n72 ~ $ cat a1.txt
water abc abcs
asd
prog1
water water
yvegorova@dk8n72 ~ $
```

Figure 3: Содержимое a1.txt

```
yvegorova@dk8n72 ~ $ cat a2.txt  
water water water  
abs  
abs abc  
water  
water 1232
```

Figure 4: Содержимое a2.txt

```
water 1232
yvegorova@dk8n72 ~ $ ./prog1.sh -i a1.txt -o a2.txt -p water -n
yvegorova@dk8n72 ~ $ cat a2.txt
1:water abc abcs
4:water water
```

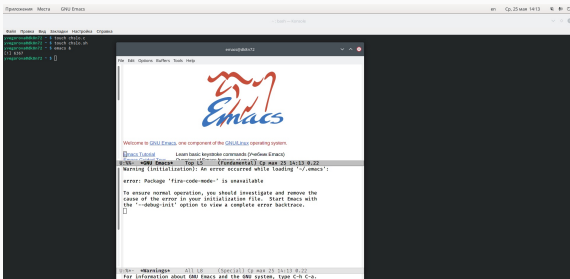
Figure 5: Проверка работы программы.


```
yvegorova@dk8n72 ~ $ ./prog1.sh -i a1.txt -o a2.txt -p water -C -n
yvegorova@dk8n72 ~ $ cat a2.txt
1:water abc abcs
4:water water
```

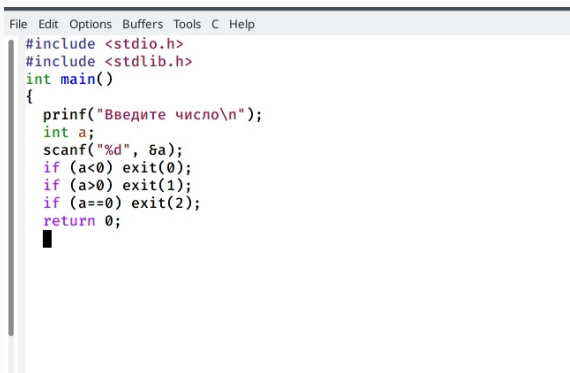
Figure 6: Проверка работы программы.

3) Написала на языке С программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

Для этого создала два файла: `chslo.c` и `chslo.sh`



Записала соответствующие скрипты:



```
File Edit Options Buffers Tools C Help
#include <stdio.h>
#include <stdlib.h>
int main()
{
    printf("Введите число\n");
    int a;
    scanf("%d", &a);
    if (a<0) exit(0);
    if (a>0) exit(1);
    if (a==0) exit(2);
    return 0;
}
```

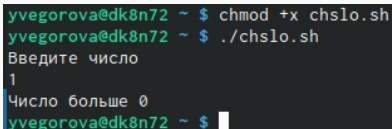
Figure 8: Скрипт файла chslo.c.

File Edit Options Buffers Tools Sh-Script Help

```
#!/bin/bash
gcc chislo.c -o chislo
./chislo
code=$?
case $code in
    0) echo "Число меньше 0" ;;
    1) echo "Число больше 0" ;;
    2) echo "Число равно 0"
esac
```

Figure 9: Скрипт файла chslo.sh.

И проверила работу программы, предварительно открыв доступ на исполнение файла:

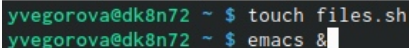
A terminal window with a dark background and light green text. The prompt is 'yvegorova@dk8n72 ~ \$'. The first command is 'chmod +x chslo.sh'. The second command is './chslo.sh'. The output is 'Введите число' followed by '1' on the next line. The next line shows 'Число больше 0'. The prompt returns to 'yvegorova@dk8n72 ~ \$' with a cursor at the end.

```
yvegorova@dk8n72 ~ $ chmod +x chslo.sh
yvegorova@dk8n72 ~ $ ./chslo.sh
Введите число
1
Число больше 0
yvegorova@dk8n72 ~ $
```

Figure 10: Проверка работы программы.

3) После этого написала командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до n . Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).

Для этого я создала файл `files.sh` и записала соответствующий скрипт:

A terminal window with a dark background. The prompt is 'yvegorova@dk8n72 ~'. The first command is '\$ touch files.sh' and the second is '\$ emacs &'.

```
yvegorova@dk8n72 ~ $ touch files.sh
yvegorova@dk8n72 ~ $ emacs &
```

Figure 11: Создание файла.

```
#!/bin/bash
opt=$1;
format=$2;
number=$3;
function Files()
{
    for (( i=1; i>=number; i++ )) do
        file=$(echo $format | tr 'a' "$i")
        if [ $opt == "-r" ]
        then
            rm -f $file
        elif [ $opt == "-c" ]
        then
            touch $file
        fi
    done
}
Files
```

U:--- files.sh All L18 (Shell-script[sh]) Cp man 25 14:43 0.18

Warning (initialization): An error occurred while loading '~/.emacs':

error: Package 'fira-code-mode-' is unavailable

To ensure normal operation, you should investigate and remove the cause of the error in your initialization file. Start Emacs with the '--debug-init' option to view a complete error backtrace.

□

U:%- *Warnings* All L8 (Special) Cp man 25 14:43 0.18

Wrote /afs/dk.sci.pfu.edu.ru/home/y/v/yvegorova/files.sh

Figure 12: Скрипт №3

Далее проверила работу написанного скрипта, добавив право на исполнение:

[illegible]

Figure 13: Проверка работы скрипта №3

4) Написала командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`).

Создала файл `prog4.sh` и записала в нем соответствующий скрипт:



Figure 15: Создание файла.

```

File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
files=$(find ./ -maxdepth 1 -mtime -7)
listing=""
for file in "$files" ; do
    file=$(echo "$file" | cit -c 3-)
    listing="$listing $file"
done
dir=$(basename $(pwd))
tar -cvf $dir.tar $listing
█

U:--- prog4.sh All L10 (Shell-script[sh]) Ср мая 25 15:29 1.06
Warning (initialization): An error occurred while loading '~/.emacs':

error: Package 'fira-code-mode-' is unavailable

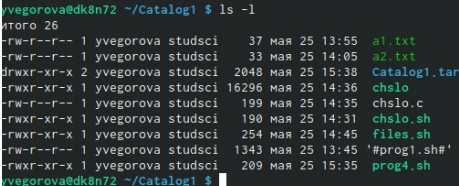
To ensure normal operation, you should investigate and remove the
cause of the error in your initialization file. Start Emacs with
the '--debug-init' option to view a complete error backtrace.
█

U:%*- *Warnings* All L8 (Special) Ср мая 25 15:29 1.06
Wrote /afs/.dk.sci.pfu.edu.ru/home/v/v/vvegorova/prog4.sh

```

Figure 16: Скрипт №4

А затем проверила работу написанного скрипта, предварительно добавив право на исполнение файла и создав отдельный каталог Catalog1 с несколькими файлами.



```
yvegorova@dk8n72 ~/Catalog1 $ ls -l
итого 26
-rw-r--r-- 1 yvegorova studsci   37 мая 25 13:55 a1.txt
-rw-r--r-- 1 yvegorova studsci   33 мая 25 14:05 a2.txt
drwxr-xr-x 2 yvegorova studsci 2048 мая 25 15:38 Catalog1.tar
-rwxr-xr-x 1 yvegorova studsci 16296 мая 25 14:36 chslo
-rw-r--r-- 1 yvegorova studsci   199 мая 25 14:35 chslo.c
-rwxr-xr-x 1 yvegorova studsci   190 мая 25 14:31 chslo.sh
-rwxr-xr-x 1 yvegorova studsci   254 мая 25 14:45 files.sh
-rw-r--r-- 1 yvegorova studsci  1343 мая 25 13:45 '#prog1.sh#'
-rwxr-xr-x 1 yvegorova studsci   209 мая 25 15:35 prog4.sh
yvegorova@dk8n72 ~/Catalog1 $
```

Figure 17: Проверка работы скрипта №4

```
yvegorova@dk8n72 ~ $ cd Catalog1
yvegorova@dk8n72 ~/Catalog1 $ ~/prog4.sh
chsl0
chsl0.c
chsl0.sh
#prog1.sh#
a2.txt
a1.txt
yvegorova@dk8n72 ~/Catalog1 $ tar -tf Catalog1.tar
chsl0
chsl0.c
chsl0.sh
#prog1.sh#
a2.txt
a1.txt
yvegorova@dk8n72 ~/Catalog1 $
```

Figure 18: Проверка работы скрипта №4

В ходе выполнения лабораторной работы я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах