

Лабораторная работа 11

Куркина Евгения Вячеславовна

¹RUDN University, Moscow, Russian Federation

Лабораторная работа 11

Цель данной лабораторной работы— Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Затем сохранила файл (рис. 3). Создала два файла, заполнила их словами(рис. 4) (рис. 5).



```
prag1.sh Top 148 (Shell-script[sh]) Fri Mar 19 14:38 2.33
ls --debug-init' option to view a complete error backtrace.
ls
***Warnings*** Sat 18 (Special) Fri Mar 19 14:38 2.33
Write /afs/.ds.nci.pfu.edu.ru/home/n/s/ovkukina/prag1.sh
```

Figure 3: Сохранение файла



```
prag1.sh
1. файл
2. файл
3. файл
4. файл
```

Figure 4: Текст первого файла



```
prag2.sh
1. файл
2. файл
3. файл
4. файл
```

Figure 5: текст второго файла

Предоставила доступ на исполнение (рис. 6), проверила корректную работу скрипта (рис. 7).

```
evkurkina@dk3n53 ~$ touch prog1.sh
evkurkina@dk3n53 ~$ emacs &
[1] 17433
evkurkina@dk3n53 ~$ touch a1.txt a2.txt
evkurkina@dk3n53 ~$ chmod +x prog1.sh
evkurkina@dk3n53 ~$ cat a1.txt
evkurkina@dk3n53 ~$ cat a1.txt
Привет
Как дела ?
Все хорошо
абсдв
```

Figure 6: Права доступа

```
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -o a2.txt -p дела -C -n
evkurkina@dk3n53 ~$ cat a2.txt
2:Как дела ?
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -o a2.txt -p дела -n
evkurkina@dk3n53 ~$ cat a2.txt
2:Как дела ?
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -C -n
./prog1.sh: строка 13: ((: ==0: синтаксическая ошибка: ожидается оператор (неверный маркер ==0))
^C
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -o a2.txt -p дела -n
evkurkina@dk3n53 ~$ cat a2.txt
2:Как дела ?
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -p дела -C -n
evkurkina@dk3n53 ~$ cat a2.txt
2:Как дела ?
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -C -n
unknown: не найден
evkurkina@dk3n53 ~$ ./prog1.sh -i a1.txt -C -n
unknown: не найден
evkurkina@dk3n53 ~$ ./prog1.sh -o a2.txt -p дела -C -n
Badn: не найден
evkurkina@dk3n53 ~$
```

Figure 7: Проверка работы скрипта

Создала два новых файла, для второго скрипта (рис. 8). На языке C написала программу, которая определяет, введенное число меньше, больше или же равно нулю, затем код завершается с помощью функции `exit(i)` (рис. 9), сохранила его (рис. 10).

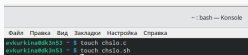


Figure 8: Созданы два файла

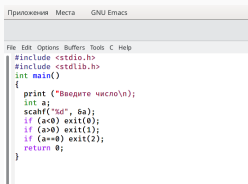
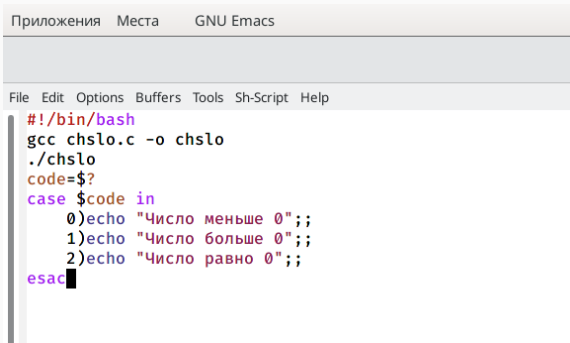


Figure 9: Код программы на языке C

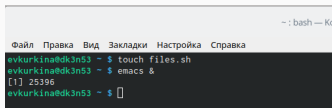
После командный файл вызывает данную программу и выдает сообщение о том, какое число было введено пользователем(рис. 11). Перешла в консоль, дала права доступа и проверила работу кода (рис. 12).



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

Figure 11: Код исполняемого файла

Создала новый файл для Зего скрипта (рис. 13). Написала код исполняемого файла, который создает указанное пользователем число файлов, которые пронумерованы от 1 до N. Число файлов передается аргументом в командной строке, также этот скрипт может удалить все созданные им файлы(рис. 14).



```
~: bash — Kc
Файл  Правка  Вид  Закладки  Настройка  Справка
avkurkina@dk3n53 ~$ touch files.sh
avkurkina@dk3n53 ~$ emacs &
[1] 25396
avkurkina@dk3n53 ~$
```

Figure 13: Создание файлов и переход в emacs



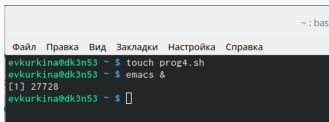
```
emacs@dk3n53
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
opt=$1;
format=$2;
number=$3;
function Files()
{
  for (( i=1; i<=$number; i++ )) do
    file=$(echo $format | tr '8' '$1')
    if [ $opt == "-r" ]
    then
      rm -f $file
    elif [ $opt == "-c" ]
    then
      touch $file
    fi
  done
}
Files
```

Далее, я предоставила доступ к исполнению, создала три файла, а затем удалила их, с помощью кода.(рис. 15)

```
evkurkina@dk3n53 ~ $ chmod +x files.sh
evkurkina@dk3n53 ~ $ ./files.sh -c abc#.txt 3
evkurkina@dk3n53 ~ $ ls
a1.txt      blog2.sh~  example.txt~  '#lab07.sh#'  prog2.sh      text.txt
a2.txt      chslo      feathers      lab07.sh      prog2.sh~    tmp
abc1        chslo.c~   file254      lab07.sh~    progals.sh    vozrast
abc1.txt    chslo.c~   file254.txt  'Matrisa(54)'  program1      work
abc2.txt    chslo.sh   file2.doc     may           program12     Видео
abc3.txt    chslo.sh~  file.doc     monthly      Programploshad  Документы
Architecture_PC conf.txt   file.pdf     my_os        programsum     Загрузки
australia  evkurkina.github.io  files.sh     os-intro     progls.sh      Изображения
backup     example2.txt~  files.sh~    play         program2       Музыка
backup.sh  example2.txt~  file.txt     poll         public         Общедоступные
backup.sh~ example3.txt  format.sh~   presentation.md  public_html    препрограмма
bin        example4.txt~  format.sh~   presentation.pdf  reports
blog       example4.txt~  GNUstep      prog1.sh      ski_places
blog2.sh   example.txt~  lab01        prog1.sh~     temp
evkurkina@dk3n53 ~ $ ./files.sh -r abc#.txt 3
evkurkina@dk3n53 ~ $ ls
a1.txt      chslo.c      feathers      '#lab07.sh#'  prog1.sh~    ski_places
a2.txt      chslo.c~     file254      lab07.sh      prog2.sh      temp
abc1        chslo.sh     file254.txt  lab07.sh~    prog2.sh~    text.txt
Architecture_PC chslo.sh~   file2.doc     'Matrisa(54)'  progals.sh    tmp
australia  evkurkina.github.io  file.doc     may           program1      vozrast
backup     evkurkina.github.io  file.pdf     monthly      program12     work
backup.sh  example2.txt~  files.sh     my_os        program12     Видео
backup.sh~ example2.txt~  files.sh~    os-intro     Programploshad  Документы
bin        example3.txt  file.txt     play         programsum     Загрузки
blog       example4.txt~  format.sh~   presentation.md  progls.sh      Изображения
blog2.sh   example4.txt~  format.sh~   presentation.pdf  public         Музыка
blog2.sh~  example.txt~  GNUstep      presentation.pdf  public_html    Общедоступные
chslo     example.txt~  lab01        prog1.sh      reports       препрограмма
evkurkina@dk3n53 ~ $
```

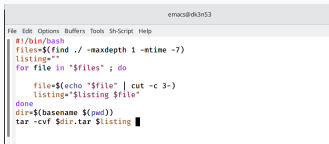
Figure 15: Права доступа и проверка работы скрипта

Создала файл для последнего 4го скрипта(рис. 16). Написала текст исполняемого файла, который с помощью команды tar собирает в архив все файлы, которые находятся в этой директории.(рис. 17).



```
~ : bash
Файл Правка Вид Закладки Настройка Справка
evkurkina@dk3n53 ~ $ touch prog4.sh
evkurkina@dk3n53 ~ $ emacs &
[1] 27728
evkurkina@dk3n53 ~ $
```

Figure 16: Создание файла для последнего скрипта



```
emacs@dk3n53
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" | cut -c 3-)
    listing="$listing $file"
done
dir=$(basename $(pwd))
tar -cvf $dir.tar $listing
```

Figure 17: Текст кода

Далее предоставила права доступа на выполнение, создала отдельный каталог, куда поместила файлы, а после проверила работу программы (рис. 18) (рис. 19).

```
evkurkina@dk3n53 ~$ chmod +x prog4.sh
evkurkina@dk3n53 ~$ mkdir Catalog1
evkurkina@dk3n53 ~$ cd ~/Catalog1
evkurkina@dk3n53 ~/Catalog1$ ls -l
итого 0
evkurkina@dk3n53 ~/Catalog1$ ls -l
итого 2
-rw-r--r-- 1 evkurkina studsci 63 мая 19 14:40 a1.txt
-rw-r--r-- 1 evkurkina studsci 21 мая 19 14:53 a2.txt
-rw-r--r-- 1 evkurkina studsci  0 мая  4 16:42 abc1
evkurkina@dk3n53 ~/Catalog1$ sudo ~/prog4.sh
```

Figure 18: Создание каталога и проверка работы скрипта

```
evkurkina@dk3n53 ~/Catalog1$ ~/prog4.sh
a1.txt
a2.txt
evkurkina@dk3n53 ~/Catalog1$ tar -tf Catalog1.tar
a1.txt
a2.txt
evkurkina@dk3n53 ~/Catalog1$
```

Figure 19: Результат работы

Во время выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

