

Отчет по лабораторной работе №11

Кашкин Иван Евгеньевич

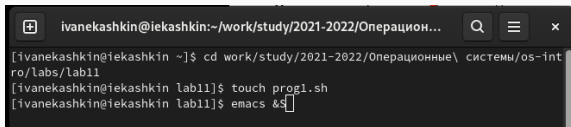
28 мая 2022 г.

РУДН, Москва, Россия

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

Прописать 4 скрипта заданных в лабораторной работе

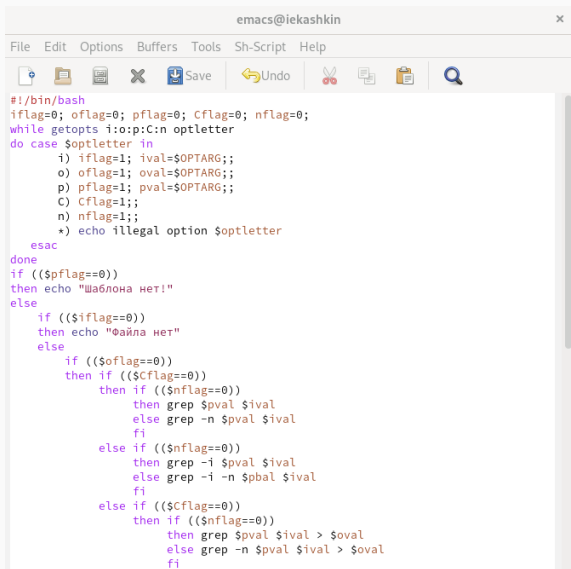
Сначала создаем файлы для скриптов и открываем эти файлы в emacs

A terminal window with a dark background and light text. The window title bar shows the user 'ivanekashkin' and the current directory '~/work/study/2021-2022/Операцион...'. The terminal content shows three lines of commands and their outputs: the first line changes the directory to 'work/study/2021-2022/Операционные системы/os-intro/labs/lab11'; the second line creates a file named 'prog1.sh' using the 'touch' command; the third line opens the file in the 'emacs' editor. The cursor is positioned at the end of the third line.

```
ivanekashkin@iekashkin: ~/work/study/2021-2022/Операцион...  
[ivanekashkin@iekashkin ~]$ cd work/study/2021-2022/Операционные\ системы/os-intro/labs/lab11  
[ivanekashkin@iekashkin lab11]$ touch prog1.sh  
[ivanekashkin@iekashkin lab11]$ emacs &[
```

Рис. 1: Создание первой программы

Первый скрипт

The image shows a screenshot of an Emacs editor window. The title bar at the top reads "emacs@iekashkin". Below the title bar is a menu bar with "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". Underneath the menu bar is a toolbar with icons for opening files, saving, undo, redo, and search. The main area of the window contains a shell script written in Bash. The script initializes flags (iflag, oflag, pflag, cflag, nflag) to 0 and enters a while loop using getopts to process command-line options. It handles options 'i', 'o', 'p', 'C', and 'n' by setting their respective values to \$OPTARG. An asterisk (*) indicates an illegal option. After the loop, it checks if any flags were set. If no flags are set, it echoes "Шаблона нет!". If flags are set, it checks if the file exists. If the file exists, it uses grep to search for the flag's value in the file. If the file does not exist, it echoes "Файла нет!". The script uses nested if-then-else-fi statements to handle the various flags and their corresponding grep commands.

```
#!/bin/bash
iflag=0; oflag=0; pflag=0; cflag=0; nflag=0;
while getopts i:o:p:C:n optletter
do case $optletter in
  i) iflag=1; ival=$OPTARG;;
  o) oflag=1; oval=$OPTARG;;
  p) pflag=1; pval=$OPTARG;;
  C) cflag=1;;
  n) nflag=1;;
  *) echo illegal option $optletter
    esac
done
if (($pflag==0))
then echo "Шаблона нет!"
else
  if (($iflag==0))
  then echo "Файла нет"
  else
    if (($oflag==0))
    then if (($cflag==0))
        then if (($nflag==0))
            then grep $pval $ival
            else grep -n $pval $ival
            fi
        else if (($nflag==0))
            then grep -i $pval $ival
            else grep -i -n $pval $ival
            fi
        else if (($cflag==0))
            then if (($nflag==0))
                then grep $pval $ival > $oval
                else grep -n $pval $ival > $oval
                fi
            fi
    fi
  fi
fi
```

Рис. 2: Программа №1

```

[ivanekashkin@iekashkin lab11]$ chmod +x *.sh
[1]+  Завершён          emacs
[ivanekashkin@iekashkin lab11]$ cat 1.txt
"1984" is John Orwell's book
"Dead Souls" is Gogol's book
Show
"War and Peace" is Leo Tolstoy's book
HELL
"The Cherry Orchard" is Chekhov's book
[ivanekashkin@iekashkin lab11]$ ./progl.sh -i 1.txt -o 2.txt -p book -C -n
[ivanekashkin@iekashkin lab11]$ cat 2.txt
"1984" is John Orwell's book
"Dead Souls" is Gogol's book
"War and Peace" is Leo Tolstoy's book
"The Cherry Orchard" is Chekhov's book
[ivanekashkin@iekashkin lab11]$ ./progl.sh -i 1.txt -o 2.txt -p book -n
[ivanekashkin@iekashkin lab11]$ cat 2.txt
1:"1984" is John Orwell's book
2:"Dead Souls" is Gogol's book
4:"War and Peace" is Leo Tolstoy's book
6:"The Cherry Orchard" is Chekhov's book
[ivanekashkin@iekashkin lab11]$ ./progl.sh -i 1.txt -C -n
Шаблона нет!
[ivanekashkin@iekashkin lab11]$ ./progl.sh -o 2.txt -p book -n
Файла нет

```

Рис. 3: Ввод программы

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Напишите число: ");
    int a;
    scanf("%d",&a);
    if (a<0) exit(0);
    if (a>0) exit(1);
    if (a==0) exit(2);
    return 0;
}
```

Рис. 4: Си

```
#!/bin/bash

gcc prog2.c -o prog2
./prog2
code=$?
case $code in
    0) echo "Меньше 0";;
    1) echo "Больше 0";;
    2) echo "Равно 0";;
esac
□
```

Рис. 5: sh


```
[ivanekashkin@iekashkin lab11]$ chmod +x *.sh
[2]+  Завершён      emacs
[ivanekashkin@iekashkin lab11]$ ./prog2.sh
Напишите число: 4
Больше 0
[ivanekashkin@iekashkin lab11]$ ./prog2.sh
Напишите число: -1
Меньше 0
[ivanekashkin@iekashkin lab11]$ ./prog2.sh
Напишите число: 0
Равно 0
```

Рис. 6: Ввод программы

```
#!/bin/bash

opt=$1;
form=$2;
num=$3;
function Files(){
    for ((i=1; i<=$num;i++)) do
        file=$(echo $form | tr '#' "$i")
        if [ $opt == "-r" ]
        then
            rm -f $file
        elif [ $opt == "-c" ]
        then
            touch $file
        fi
    done
}
Files
```

Рис. 7: Программа №3

```

[ivanekashkin@iekashkin lab11]$ chmod +x *.sh
[ivanekashkin@iekashkin lab11]$ ls
1.txt  presentation  prog1.sh~  prog2.c  prog2.sh  prog3.sh  report
2.txt  prog1.sh      prog2      prog2.c~  prog2.sh~  prog3.sh~
[ivanekashkin@iekashkin lab11]$ ./prog3.sh -c w#.txt 4
[ivanekashkin@iekashkin lab11]$ ls
1.txt      prog1.sh  prog2.c  prog2.sh~  report  w3.txt
2.txt      prog1.sh~ prog2.c~  prog3.sh  w1.txt  w4.txt
presentation  prog2      prog2.sh  prog3.sh~  w2.txt
[ivanekashkin@iekashkin lab11]$ ./prog3.sh -r w#.txt 4
[ivanekashkin@iekashkin lab11]$ ls
1.txt  presentation  prog1.sh~  prog2.c  prog2.sh  prog3.sh  report
2.txt  prog1.sh      prog2      prog2.c~  prog2.sh~  prog3.sh~

```

Рис. 8: Ввод программы

```
#!/bin/bash

file=$(find ./ -maxdepth 1 -mtime -7)
listing=""
for file in "$files"; do
    file=$(echo "$file" | cut -c 3-)
    listing="$listing $file"
done
dir=$(basename $(pwd))
tar -cvf $dir.tar $listing
█
```

Рис. 9: Программа №4

```

ivanekashkin@iekashkin lab11]$ chmod +x *.sh
[1]+  Завершён      emacs
ivanekashkin@iekashkin lab11]$ ls -l
итого 64
-rw-rw-r--. 1 ivanekashkin ivanekashkin 145 мая 25 13:22 1.txt
-rw-rw-r--. 1 ivanekashkin ivanekashkin 143 мая 25 13:45 2.txt
drwxrwxr-x. 1 ivanekashkin ivanekashkin   0 апр 22 12:06 presentation
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  976 мая 25 13:44 prog1.sh
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  967 мая 25 12:50 prog1.sh-
-rwxrwxr-x. 1 ivanekashkin ivanekashkin 25096 мая 25 13:58 prog2
-rw-rw-r--. 1 ivanekashkin ivanekashkin  200 мая 25 13:54 prog2.c
-rw-rw-r--. 1 ivanekashkin ivanekashkin   0 мая 25 13:49 prog2.c~
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  157 мая 25 13:57 prog2.sh
-rw-rw-r--. 1 ivanekashkin ivanekashkin   0 мая 25 13:49 prog2.sh~
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  233 мая 25 14:06 prog3.sh
-rw-rw-r--. 1 ivanekashkin ivanekashkin   0 мая 25 14:00 prog3.sh~
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  210 мая 25 14:33 prog4.sh
-rwxrwxr-x. 1 ivanekashkin ivanekashkin  209 мая 25 14:30 prog4.sh-
drwxrwxr-x. 1 ivanekashkin ivanekashkin   0 апр 23 21:50 report
ivanekashkin@iekashkin lab11]$ sudo ~/work/study/2021-2022/Операционные\системы
/os-intro/labs/lab11/prog4.sh
sudo: /home/ivanekashkin/work/study/2021-2022/Операционные\системы/os-intro/labs/
lab11/prog4.sh: command not found
ivanekashkin@iekashkin lab11]$ ./prog4.sh
2.txt
1.txt
prog1.sh~
prog1.sh
prog2.c~
prog2.c
prog2.sh~
prog2.sh
prog2
prog3.sh~
prog3.sh
prog4.sh~
prog4.sh
ivanekashkin@iekashkin lab11]$ tar -tf lab11.tar
tar: lab11.tar: Функция open завершилась с ошибкой: Нет такого файла или каталог
a
tar: Error is not recoverable: exiting now
ivanekashkin@iekashkin lab11]$ tar -tf Операционные.tar
2.txt
1.txt
prog1.sh~
prog1.sh

```

Рис. 10: Запуск программы

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

Спасибо за внимание!