

Лабораторная работа № 13

Операционные системы

Казазаев Д. М.

Российский университет дружбы народов, Москва, Россия

Информация

- Казазаев Даниил Михайлович
- Студент бакалавриата
- Российский университет дружбы народов
- [1132231427@rudn.ru]
- https://github.com/KazazaevDaniil/study_2023-2024_os-intro

Вводная часть

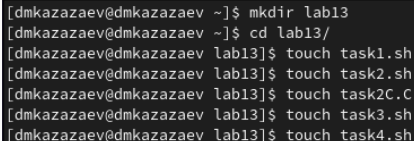
Изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

1. Выполнить задания лабораторной работы

Основная часть

Задание 1

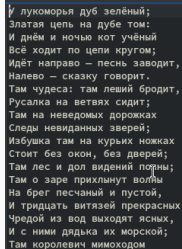
Создаю каталог и файлы для заданий. (рис. 1)

A screenshot of a terminal window with a dark background and light-colored text. It shows a series of commands being executed in a shell. The user is at the prompt [dmkazazaev@dmkazazaev ~]. The commands are: mkdir lab13, cd lab13/, touch task1.sh, touch task2.sh, touch task2C.C, touch task3.sh, and touch task4.sh. The prompt changes to [dmkazazaev@dmkazazaev lab13] after the cd command.

```
[dmkazazaev@dmkazazaev ~]$ mkdir lab13
[dmkazazaev@dmkazazaev ~]$ cd lab13/
[dmkazazaev@dmkazazaev lab13]$ touch task1.sh
[dmkazazaev@dmkazazaev lab13]$ touch task2.sh
[dmkazazaev@dmkazazaev lab13]$ touch task2C.C
[dmkazazaev@dmkazazaev lab13]$ touch task3.sh
[dmkazazaev@dmkazazaev lab13]$ touch task4.sh
```

Рис. 1: Создание каталога и файлов для заданий

оздаю еще два файла для первого задания и переношу в один из них текст. (рис. 2)



У лукоморья дуб зелёный;
Златая цепь на дубе том:
И днём и ночью кот учёный
Всё ходит по цепи кругом;
Идёт направо — песнь заводит,
Налево — сказку говорит.
Там чудеса: там леший бродит,
Русалка на ветвях сидит;
Там на неведомых дорожках
Следы невиданных зверей;
Избушка там на курьих ножках
Стоит без окон, без дверей;
Там лес и дол видений полны;
Там о заре прихлынут волны
На брег песчаный и пустой,
И тридцать витязей прекрасных
Чредой из вод выходят ясных,
И с ними дядька их морской;
Там королевич мимоходом

Рис. 2: Перенесенный текст

После переноса текста перехожу в файл с расширением .sh и пишу программу, которая соответствует заданию. (рис. 3)

```
1 |! ./bin/bash
2
3 while getopts t:op:cn optionletters
4 do
5     case $optionletters in
6         t) tflag=1; sval=$OPTARG;;
7         o) oflag=1; sval=$OPTARG;;
8         p) pflag=1; sval=$OPTARG;;
9         c) cflag=1;;
10        n) nflag=1;;
11    esac
12 done
13
14 if [ test $cflag
15 then
16     cf=-t
17 fi
18
19 if test $nflag
20 then
21     nf=-n
22 fi
23
24 trap {cf $nf $pval $tval} 0val
25
```

Рис. 3: Код программы

Выполнение лабораторной работы. Отчет по лабораторной работе № 13.

Листинг первой программы:

```
#!/bin/bash

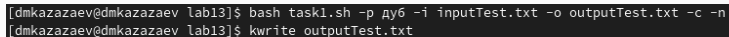
while getopts i:o:p:cn optionletters
do
    case $optionletters in
        i) iflag=1; ival=$OPTARG;;
        o) oflag=1; oval=$OPTARG;;
        p) pflag=1; pval=$OPTARG;;
        c) cflag=1;;
        n) nflag=1;;
    esac
done
```

```
if ! test $cflag
then
    cf=-i
fi
```

```
if test $nflag
then
    nf=-n
fi
```

```
grep $cf $nf $pval $ival >> $oval
```

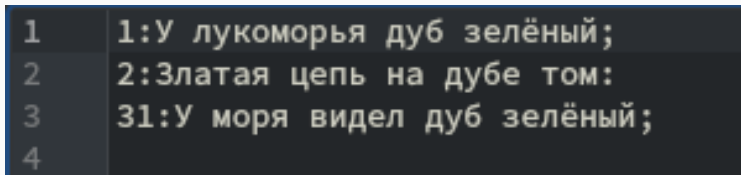
После чего запускаю программу. (рис. 4)



```
[dmkazazae@dmkazazae lab13]$ bash task1.sh -p дуб -i inputTest.txt -o outputTest.txt -c -n  
[dmkazazae@dmkazazae lab13]$ kwrite outputTest.txt
```

Рис. 4: Запуск программы

Открываю второй файл, чтобы проверить выполнение программы. (рис. 5)



```
1 1:У лукоморья дуб зелёный;  
2 2:Златая цепь на дубе том:  
3 31:У моря видел дуб зелёный;  
4
```

Рис. 5: Результат программы

Задание 2

В файле с расширением .c пишу программу, которая сравнивает число с нулем. (рис. 6)

```
1  #include <stdlib.h>
2  #include <stdio.h>
3
4  int main(){
5      int n;
6      printf ("Введите число: ");
7      scanf("%d", &n);
8      if(n > 0){
9          exit(1);
10     }
11     if(n == 0){
12         exit(0);
13     }
14     if(n < 0){
15         exit(2);
16     }
17 }
18
```

Рис. 6: Код программы на языке C

Листинг программы на языке C:

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

int main(){
    int n;
    printf ("Введите число: ");
    scanf("%d", &n);
    if(n > 0){
        exit(1);
    }
```



```
if(n == 0){  
    exit(0);  
}  
if(n < 0){  
    exit(2);  
}  
}
```

Выполнение лабораторной работы. Отчет по лабораторной работе № 13.

После чего перехожу в файл с расширением .sh и пишу программу, которая будет вызывать программу на языке С и, проанализировав с помощью \$?, выводить на экран результат сравнения. (рис. 7)

```
1  #! /bin/bash/
2
3  gcc -o lab13 task2C.c
4  ./lab13
5  case $? in
6  0) echo "Число равно нулю";;
7  1) echo "Число больше нуля";;
8  2) echo "Число меньше нуля";;
9  esac
10
```

Рис. 7: Код программы на языке Bash

Листинг программы на языке Bash:

```
#!/bin/bash/

gcc -o lab13 task2C.c
./lab13
case $? in
0) echo "Число равно нулю";;
1) echo "Число больше нуля";;
2) echo "Число меньше нуля";;
esac
```

Проверяю работу программы. (рис. 8)

```
[dmkazazaev@dmkazazaev lab13]$ bash task2.sh
Введите число: 2
Число больше нуля
[dmkazazaev@dmkazazaev lab13]$ bash task2.sh
Введите число: 13
Число больше нуля
[dmkazazaev@dmkazazaev lab13]$ bash task2.sh
Введите число: 0
Число равно нулю
[dmkazazaev@dmkazazaev lab13]$ bash task2.sh
Введите число: -1
Число меньше нуля
```

Рис. 8: Результат выполнения программы

Задание 3

Перехожу в третий файл и пишу программу, которая будет создавать n-ное число временных файлов, а если они уже существуют, что удалять их. (рис. 9)

```
1  #!/bin/bash
2
3  for((i=1; i<=$*; i++))
4  do
5  if test -f "$i".tmp
6  then rm "$i".tmp
7  else touch "$i".tmp
8  fi
9  done
10
```

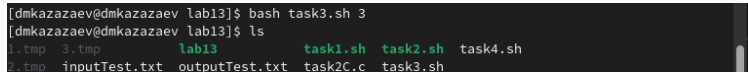
Рис. 9: Код третьей программы

Листинг третьей программы:

```
#!/bin/bash
```

```
for((i=1; i<=$*; i++))  
do  
if test -f "$i".tmp  
then rm "$i".tmp  
else touch "$i".tmp  
fi  
done
```

Запускаю программу и проверяю, создались ли файлы. (рис. 10)

A terminal window with a dark background. The prompt is [dmkazazae@dmkazazae lab13]. The first command is bash task3.sh 3. The second command is ls. The output of ls is displayed in two lines: 1.tmp 3.tmp lab13 task1.sh task2.sh task4.sh and 2.tmp inputTest.txt outputTest.txt task2C.c task3.sh. The file names are color-coded: lab13 is green, task1.sh and task2.sh are red, and the others are white.

```
[dmkazazae@dmkazazae lab13]$ bash task3.sh 3
[dmkazazae@dmkazazae lab13]$ ls
1.tmp  3.tmp  lab13  task1.sh  task2.sh  task4.sh
2.tmp  inputTest.txt  outputTest.txt  task2C.c  task3.sh
```

Рис. 10: Результат программы

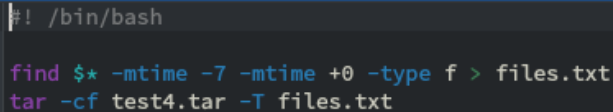
Запускаю парогрammu еще раз и проверяю, удалились ли файлы. (рис. 11)

```
[dmkazazaev@dmkazazaev lab13]$ bash task3.sh 3  
[dmkazazaev@dmkazazaev lab13]$ ls  
inputTest.txt  lab13  outputTest.txt  task1.sh  task2C.c  task2.sh  task3.sh  task4.sh
```

Рис. 11: Результат программы

Задание 4

Перехожу в четвертый файл и пишу программу, которая будет архивировать файлы, которые были изменены менее недели назад. (рис. 12)



```
#!/bin/bash

find $* -mtime -7 -mtime +0 -type f > files.txt
tar -cf test4.tar -T files.txt
```

Рис. 12: Код четвертой программы

Листинг четвертой программы:

```
#!/bin/bash
```

```
find $* -mtime -7 -mtime +0 -type f > files.txt
```

```
tar -cf test4.tar -T files.txt
```

Запускаю программу и проверяю результат выполнения. (рис. 13)

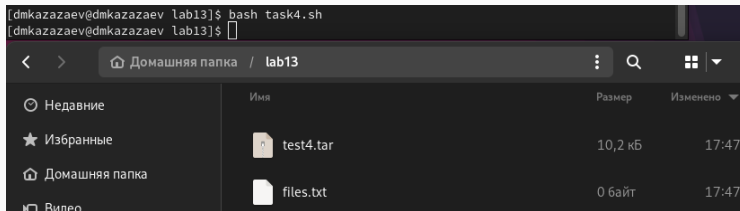


Рис. 13: Результат выполнения

Заключительная часть

- Получены практические программирования более сложных командных файлов с использованием логических управляющих конструкций и циклов.

В результате выполнения лабораторной работы я изучил основы программирования в оболочке ОС UNIX, научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.