Отчёт по лабораторной работе №13

Дисциплина: операционные системы

Казаазев Даниил Михайлович

Содержание

# 1 Цель работы

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

# 2 Задание

1. Выполнить задания лабораторной работы

# 3 Выполнение лабораторной работы

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

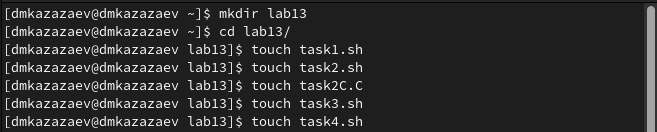


Рис. 1: Создание каталога и файлов для заданий

## 3.1 Задание 1

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

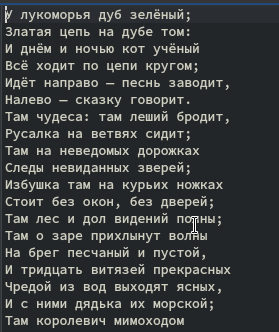


Рис. 2: Перенесенный текст

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

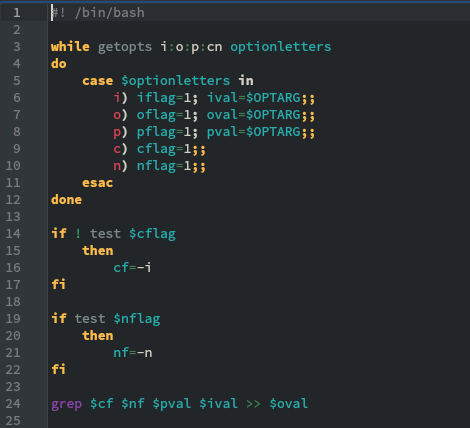


Рис. 3: Код программы

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)

Запуск программы

Рис. 4: Запуск программы

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

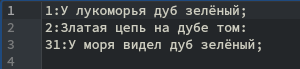


Рис. 5: Результат программы

# 4 Задание 2

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



Рис. 6: Код программы на языке С

Листинг программы на языке С:

#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);  
 }  
}

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)

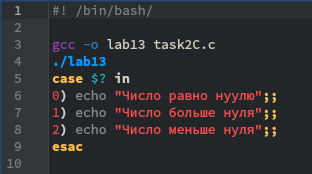


Рис. 7: Код программы на языке Bash

Листинг программы на языке Bash:

#! /bin/bash/  
  
gcc -o lab13 task2C.c  
./lab13  
case $? in  
0) echo "Число равно нуулю";;  
1) echo "Число больше нуля";;  
2) echo "Число меньше нуля";;  
esac

Проверяю работу программы. (рис. 8)

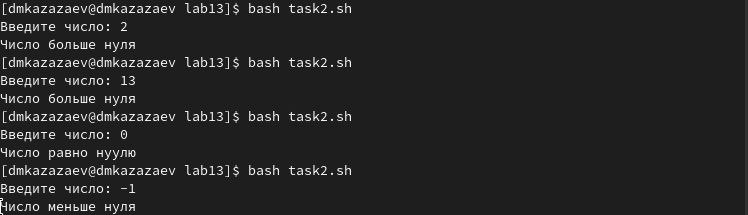


Рис. 8: Результат выполнения программы

# 5 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)

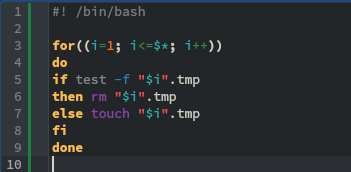


Рис. 9: Код третьей программы

Листинг тертьей 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 парограмму и проверяю, создались ли файлы. (рис. 10)

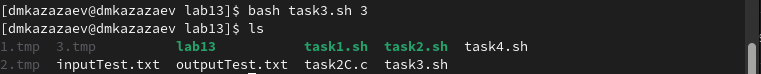


Рис. 10: Результат программы

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

Результат программы

Рис. 11: Результат программы

# 6 Задание 4

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

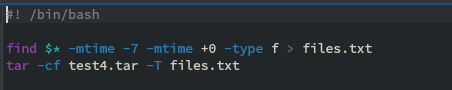


Рис. 12: Код четвертой программы

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)

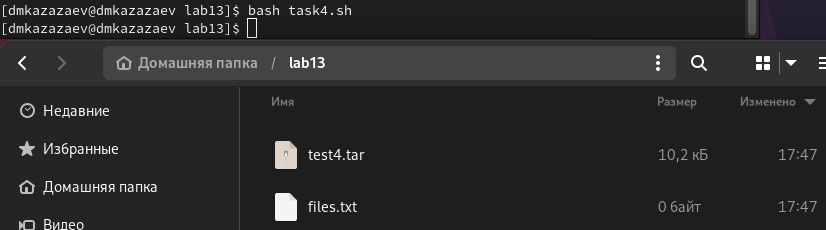


Рис. 13: Результат выполнения

# 7 Выводы

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.