

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ЛАБОРАТОРНАЯ РАБОТА №13

дисциплина: Операционные системы

Студент: Султанова Лейла

Группа: НБИбд-03-22


Москва

2023 г.

- **Цель работы**

Приобрести простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования С калькулятора с простейшими функциями

1. В домашнем каталоге создаем подкаталог
~/work/os/lab_prog.

A terminal window with a light blue title bar containing the text "Файл Правка Вид Терминал Выход Справка". The terminal content shows the command "lsultanova@lsultanova ~]\$ mkdir lab_prog" followed by a cursor. The rest of the terminal window is black.

```
lsultanova@lsultanova ~]$ mkdir lab_prog
```

Создаем в нём файлы: calculate.h, calculate.c, main.c. Это будет примитивнейший калькулятор, способный складывать, вычитать, умножать и делить, возводить число в степень, брать квадратный корень, вычислять sin, cos, tan. При запуске он будет запрашивать первое число, операцию, второе число. После этого программа выведет результат и остановится

```
lsultanova@lsultanova lab_prog]$ touch calculate.h
lsultanova@lsultanova lab_prog]$ touch calculate.c
lsultanova@lsultanova lab_prog]$ touch main.c
lsultanova@lsultanova lab_prog]$
```

Пишем скрипты в файлах calculate.h, calculate.c, main.c

```
calculate.c 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
```

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include "calculate.h"

float
Calculate(float Numeral, char Operation[4])
{
    float SecondNumeral;
    if(strncsp(Operation, "+", 1) == 0)
    {
        printf("Введите второе число: ");
        scanf("%f", &SecondNumeral);
        return(Numeral + SecondNumeral);
    }
    else if(strncsp(Operation, "-", 1) == 0)
    {
        printf("Введите второе число: ");
        scanf("%f", &SecondNumeral);
        return(Numeral - SecondNumeral);
    }
    else if(strncsp(Operation, "*", 1) == 0)
    {
        printf("Введите второе число: ");
        scanf("%f", &SecondNumeral);
        return(Numeral * SecondNumeral);
    }
    else if(strncsp(Operation, "/", 1) == 0)
    {
        printf("Введите второе число: ");
        scanf("%f", &SecondNumeral);
        if(SecondNumeral == 0)
        {
            printf("Ошибка: деление на ноль\n");
            return(0);
        }
        return(Numeral / SecondNumeral);
    }
    else
    {
        printf("Неизвестная операция\n");
        return(0);
    }
}
```

```
lab12-2.sh      [~---] 5 L [ 1+ 6 7/ 15] *(58 / 219b) 0032 0x020
#!/bin/bash
M=10
c=1
d=1
echo
echo "10 random words"
while (( $c != ($M+1) ))
do
    echo $(for((i=1; i<=10; i++)); do printf '%s' "${RANDOM:0:1}"; done) | tr '0-9' '[a-z]'
    echo $d
    ((c+=1))
    ((d+=1))
done
```

```
calculate.h      [-M--] 24 L [ 1+ 8 9/ 9] *(179 / 179b) <EOF>
////////////////////
// calculate.h

#ifndef CALCULATE_H_
#define CALCULATE_H_

float Calculate(float Numeral, char Operation[4]);

#endif /*CALCULATE_H_*/
```

4. Выполним компиляцию программы посредством gcc:

```
[lsultanova@lsultanova lab_prog]$ gcc -c calculate.c  
[lsultanova@lsultanova lab_prog]$ gcc -c main.c  
[lsultanova@lsultanova lab_prog]$ gcc calculate.o main.o -o calcul -lm
```

5. Создайте Makefile со следующим содержанием:

```
[lsultanova@lsultanova lab_prog]$ touch Makefile
[lsultanova@lsultanova lab_prog]$
```

```

Makefile: [-M-]  W 1  14:12  117/21  *lib  / 100%  0/0  0/0  0/0  0/0
#
# Makefile
#
CC = gcc
CFLAGS =
LIBS = -lm

calcul: calculate.o main.o
    gcc calculate.o main.o -o calcul $(LIBS)

calculate.o: calculate.c calculate.h
    gcc -c calculate.c $(CFLAGS)

main.o: main.c calculate.h
    gcc -c main.c $(CFLAGS)

clean:
    rm calcul *.o *~

# End Makefile

```


6. С помощью gdb выполним отладку программы calcul
(перед использованием gdb исправьте Makefile):

- Запустим отладчик GDB, загрузив в него программу для отладки:

```
GNU gdb (GDB) Fedora Linux 12.1-7.fc37
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-redhat-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>
```

Вывод

Я приобрела простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования C калькулятора с простейшими функциями.