

Российский университет дружбы народов

факультет физико-математических и естественных наук

Отчет по лабораторной работе № 14

дисциплина : Операционные системы

студент Блохин александр НКН

Москва

2021

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.

Ход работы

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

```
al@al-VirtualBox:~$ mkdir work
al@al-VirtualBox:~$ mkdir work/os
al@al-VirtualBox:~$ mkdir work/os/lab_prog
al@al-VirtualBox:~$ cd work/os/lab_prog
al@al-VirtualBox:~/work/os/lab_prog$ pwd
/home/al/work/os/lab_prog
al@al-VirtualBox:~/work/os/lab_prog$
```

2. Создайте 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.

```
al@al-VirtualBox:~/work/os/lab_prog$ touch calculate.c
al@al-VirtualBox:~/work/os/lab_prog$ touch calculate.h
al@al-VirtualBox:~/work/os/lab_prog$ touch main.c
```

```

#include <stdio.h>
#include <math.h>
#include <string.h>
#include "calculate.h"
float
Calculate(float Numeral, char Operation[4])
{
    float SecondNumeral;
    if(strncmp(Operation, "+", 1) == 0)
    {
        printf("Второе слагаемое: ");
        scanf("%f",&SecondNumeral);
        return(Numeral + SecondNumeral);
    }
    else if(strncmp(Operation, "-", 1) == 0)
    {
        printf("Вычитаемое: ");
        scanf("%f",&SecondNumeral);
        return(Numeral - SecondNumeral);
    }
    else if(strncmp(Operation, "*", 1) == 0)
    {
        printf("Множитель: ");
        scanf("%f",&SecondNumeral);
    }
}
U:--- calculate.c Top L1 (C/*l Abbrev)
tool-bar open-file

```

```

#ifndef CALCULATE_H_
#define CALCULATE_H_

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

#endif /*CALCULATE_H_*/

```

```

Обзор Emacs (GUI) Пт, 24 сентября 03:54 en
emacs@al-VirtualBox
File Edit Options Buffers Tools C Help
Save Undo
#include <stdio.h>
#include "calculate.h"
int
main (void)
{
    float Numeral;
    char Operation[4];
    float Result;
    printf("Число: ");
    scanf("%f",&Numeral);
    printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
    scanf("%s",&Operation);
    Result = Calculate(Numeral, Operation);
    printf("%6.2f\n",Result);
    return 0;
}
U:--- main.c All L1 (C/*l Abbrev)
tool-bar open-file


```

3. Выполните компиляцию программы посредством gcc: gcc -c calculate.c gcc -c main.c gcc calculate.o main.o -o calcul -lm

4. При необходимости исправьте синтаксические ошибки.(Добавил -g для отладки)

```
al@al-VirtualBox:~/work/os/lab_prog$ gcc -c calculate.c
al@al-VirtualBox:~/work/os/lab_prog$ gcc -c main.c
al@al-VirtualBox:~/work/os/lab_prog$ gcc calculate.o main.o -o calcul -lm
al@al-VirtualBox:~/work/os/lab_prog$
```

5. Создайте Makefile со следующим содержанием



```
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 *~
```

6. С помощью gdb выполните отладку программы calcul

```
(gdb) run
Starting program: /home/al/work/os/lab_prog/calcul
Число: 1
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 3
    4.00
[Inferior 1 (process 8007) exited normally]
(gdb)
```

```
(gdb) list
1      #include <stdio.h>
2      #include "calculate.h"
3      int
4      main (void)
5      {
6          float Numeral;
7          char Operation[4];
8          float Result;
9          printf("Число: ");
10         scanf("%f",&Numeral);
(gdb) list 12,15
12         scanf("%s", Operation);
13         Result = Calculate(Numeral, Operation);
14         printf("%6.2f\n",Result);
15         return 0;
(gdb)
```

```
(gdb) list calculate.c:20,29
20     }
21     else if(strncmp(Operation, "*", 1) == 0)
22     {
23         printf("Множитель: ");
24         scanf("%f",&SecondNumeral);
25         return(Numeral * SecondNumeral);
26     }
27     else if(strncmp(Operation, "/", 1) == 0)
28     {
29         printf("Делитель: ");
(gdb) break 21
Breakpoint 1 at 0x1319: file calculate.c, line 21.
(gdb)
```

```
(gdb) info breakpoints
Num   Type      Disp Enb Address                What
1     breakpoint keep y  0x00000000000001319 in Calculate
                                           at calculate.c:21
(gdb)
```

```
(gdb) run
Starting program: /home/al/work/os/lab_prog/calcul
Число: 1
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +

Breakpoint 3, Calculate (Numeral=4.59163468e-41,
    Operation=0xc2 <error: Cannot access memory at address 0xc2>)
    at calculate.c:7
7   {
(gdb) display Numeral
1: Numeral = 4.59163468e-41
(gdb) print Numeral
$1 = 4.59163468e-41
(gdb) delete 1
(gdb) delete 2
(gdb) delete 3
(gdb) info breakpoints
No breakpoints or watchpoints.
(gdb)
```

Вывод

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.