

Отчет по лабораторной работе №13

Алмазова Елизавета Андреевна

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №13

Цель работы и задание

Цель данной лабораторной 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.

Задание:

1. В домашнем каталоге создайте подкаталог `~/work/os/lab_prog`.
2. Создайте в нём файлы: `calculate.h`, `calculate.c`, `main.c`. Это будет примитивнейший

Ход работы

1. В домашнем каталоге создала подкаталог `~/work/os/lab_prog`.
2. Создала в нём файлы: `calculate.c` (рис.1), `calculate.h` (рис.2), `main.c` (рис.3). 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.

```

1 //////////////////////////////////////////////////
2 // calculate.c
3
4
5 #include <stdio.h>
6 #include <math.h>
7 #include <string.h>
8 #include "calculate.h"
9
10 float Calculate(float Numeral, char Operation[4])
11 {
12     float SecondNumeral;
13     if(strncmp(Operation, "+", 1) == 0) {
14         printf("Второе слагаемое: ");
15         scanf("%f", &SecondNumeral);
16         return(Numeral + SecondNumeral);
17     }
18     else if(strncmp(Operation, "-", 1) == 0) {
19         printf("Вычитаемое: ");
20         scanf("%f", &SecondNumeral);
21         return(Numeral - SecondNumeral);
22     }
23     else if(strncmp(Operation, "*", 1) == 0) {
24         printf("Множитель: ");
25         scanf("%f", &SecondNumeral);
26         return(Numeral * SecondNumeral);
27     }
28     else if(strncmp(Operation, "/", 1) == 0) {
29         printf("Делитель: ");
30         scanf("%f", &SecondNumeral);
31         if(SecondNumeral == 0) {
32             printf("Ошибка: деление на ноль! ");
33             return(HUGE_VAL);
34         }
35         else return(Numeral / SecondNumeral);
36     }
37     else if(strncmp(Operation, "pow", 3) == 0) {
38         printf("Степень: ");
39         scanf("%f", &SecondNumeral);
40         return(pow(Numeral, SecondNumeral));
41     }
42     else if(strncmp(Operation, "sqrt", 4) == 0) return(sqrt(Numeral));
43     else if(strncmp(Operation, "sin", 3) == 0) return(sin(Numeral));
44     else if(strncmp(Operation, "cos", 3) == 0) return(cos(Numeral));
45     else if(strncmp(Operation, "tan", 3) == 0) return(tan(Numeral));
46     else {
47         printf("Неправильно введено действие. ");
48         return(HUGE_VAL);
49     }
50 }

```

Рисунок 1 - calculate.c.

```
////////////////////////////////////  
// calculate.h  
  
#ifndef CALCULATE_H_  
#define CALCULATE_H_  
  
float Calculate(float Numeral, char Operation[4]);  
  
#endif /*CALCULATE_H_*/
```

Рисунок 2 - calculate.h.

```
1 ///////////////////////////////////////////////////
2 // main.c
3
4 #include <stdio.h>
5 #include "calculate.h"
6
7 int main(){
8     float Numeral;
9     char Operation[4];
10    float Result;
11    printf("Число: ");
12    scanf("%f",&Numeral);
13    printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
14    scanf("%s",&Operation);
15    Result = Calculate(Numeral, Operation);
16    printf("%.2f\n",Result);
17    return 0;
18 }
```

Рисунок 3 - main.c.

4.Выполнила компиляцию программы посредством gcc (gcc -c calculate.c, gcc -c main.c, gcc calculate.o main.o -o calcul -lm). Добавила пропущенные точки с запятой после предупреждения компилятора.

5.Создала Makefile. Переменная CC содержит название утилиты gcc, CFLAGS содержит опции утилиты, LIBS - опции для объединения объектных файлов в исполняемый. Имеются 4 цели: clean для удаления созданных исполняемых и объектных файлов, calcul для компиляции исполняемого файла, calculate.o и


```
#  
# Makefile  
#  
CC = gcc  
CFLAGS = -g  
LIBS = -lm  
  
calcul: calculate.o main.o  
    $(CC) calculate.o main.o -o calcul $(LIBS)  
calculate.o: calculate.c calculate.h  
    $(CC) -c calculate.c $(CFLAGS)  
main.o: main.c calculate.h  
    $(CC) -c main.c $(CFLAGS)  
clean:  
    -rm calcul *.o *~  
  
# End Makefile
```

Рисунок 4 - Отредактированный Makefile.

 calcul	27.9 kB
 calculate.c	1.5 kB
 calculate.h	162 bytes
 calculate.o	7.1 kB
 main.c	352 bytes
 main.o	4.6 kB
 Makefile	290 bytes

Рисунок 5 - Результат работы Makefile по компилированию программы.

```
[eaalmazova@fedora lab_prog]$ ./calcul  
Число: 20  
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -  
Вычитаемое: 15  
5.00  
[eaalmazova@fedora lab_prog]$
```

Рисунок 6 - Работа скомпилированной программы.

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

Рисунок 7 - list calculate.c:20,29.

```

Breakpoint 1, Calculate (Numeral=5, Operation=0x7fffffffdf44 "-") at calculate.c
:21
21             return(Numeral-SecondNumeral);
(gdb) print Numeral
$1 = 5
(gdb) display Numeral
1: Numeral = 5
(gdb) info breakpoints
Num      Type           Disp Enb Address                  What
1        breakpoint      keep y   0x000000000000401282 in Calculate
                                                at calculate.c:21

        breakpoint already hit 1 time
(gdb) delete 1

```

Рисунок 8 - Работа с точкой останова.

7. С помощью утилиты splint проанализировала коды файлов `calculate.c` и `main.c`. Выяснились например такие факты, что в перечисленных файлах есть функция `scanf`, которая возвращает целое значение, но эти числа нигде не используются и не сохраняются. Также есть потеря данных при записи в переменную типа `float` и сравнение вещественного числа с нулем.

Выводы

В ходе выполнения данной лабораторной работы 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.

Спасибо за внимание!