Российский университет дружбы народов

Факультет физико-математический и естественных наук

Кафедра прикладной информатики и теории вероятностей

Лабораторная работа №13 “Средства, применяемые при разработке программного обеспечения в ОС типа UNIX/Linux”

Дисциплина

Операционные системы

Студент

Олейник Анастасия Игоревна

Группа

НБИбд-03-21

Отчёт

1. В домашнем каталоге оздайте подкаталог /work/os/lab_prog.

2. оздайте 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. Реализация функций калькулятора в файле calculate.h: fdv

calculate.c

```c
////////////////////////////////////
// calculate.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);
    return(Numeral * SecondNumeral);
  } else if (strncmp(Operation, "/", 1) == 0) {
    printf("Делитель: ");
    scanf("%f",&SecondNumeral);
    if (SecondNumeral == 0) {
      printf("Ошибка: деление на ноль!");
      return(HUGE_VAL);
    }
    return(Numeral / SecondNumeral);
  } else if (strncmp(Operation, "pow", 3) == 0) {
    printf("Степень: ");
    scanf("%f",&SecondNumeral);
    return(pow(Numeral, SecondNumeral));
  } else if(strncmp(Operation, "sqrt", 4) == 0) {
    return(sqrt(Numeral));
  } else if(strncmp(Operation, "sin", 3) == 0) {
    return(sin(Numeral));
  } else if(strncmp(Operation, "cos", 3) == 0) {
    return(cos(Numeral));
  } else if(strncmp(Operation, "tan", 3) == 0) {
    return(tan(Numeral));
  } else {
    printf("Неправильно введено действие ");
    return(HUGE_VAL);
  }
}

```

calculate.h

```c
//////////////////////////////////////
// calculate.h

#ifndef CALCULATE_H_
#define CALCULATE_H_

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

#endif /*CALCULATE_H_*/

```

main.c

```c
//////////////////////////////////////
// main.c

#include <stdio.h>
#include "calculate.h"

int main() {
    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;
}

```

```
[anastasia@Air-Anastasia lab13 % gcc -c calculate.c
[anastasia@Air-Anastasia lab13 % gcc -c main.c
main.c:16:13: warning: format specifies type 'char *' but the argument has type
'char (*)[4]' [-Wformat]
        scanf("%s",&Operation);
              ~~  ^~~~~~~~~~
1 warning generated.
[anastasia@Air-Anastasia lab13 % gcc calculate.o main.o -o calcul -lm
[anastasia@Air-Anastasia lab13 % ls
Makefile        calculate.c     calculate.o     main.o
calcul          calculate.h     main.c

[anastasia@Air-Anastasia lab13 % gcc -c calculate.c
[anastasia@Air-Anastasia lab13 % gcc -c main.c
main.c:16:13: warning: format specifies type 'char *' but the argument has type
'char (*)[4]' [-Wformat]
        scanf("%s",&Operation);
              ~~  ^~~~~~~~~~
1 warning generated.
[anastasia@Air-Anastasia lab13 % gcc calculate.o main.o -o calcul -lm
[anastasia@Air-Anastasia lab13 % ls
Makefile        calculate.c     calculate.o     main.o
calcul          calculate.h     main.c
```

Makefile

```
#
# 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

```

```
anastasia@Air-Anastasia lab13 % make clean
rm calcul *.o *~
rm: *~: No such file or directory
make: [clean] Error 1 (ignored)
anastasia@Air-Anastasia lab13 % make
gcc -c calculate.c -g
gcc -c main.c -g
main.c:16:13: warning: format specifies type 'char *' but the argument has type
'char (*)[4]' [-Wformat]
        scanf("%s",&Operation);
               ~~  ^~~~~~~~~~
1 warning generated.
gcc calculate.o main.o -o calcul -lm
anastasia@Air-Anastasia lab13 % lldb calcul
(lldb) target create "calcul"
Current executable set to '/Users/anastasia/Developer/lab13/calcul' (arm64).
(lldb) run
Process 6362 launched: '/Users/anastasia/Developer/lab13/calcul' (arm64)
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): *
Множитель: 10
 50.00
Process 6362 exited with status = 0 (0x00000000)
(lldb)
(lldb) run
Process 6373 launched: '/Users/anastasia/Developer/lab13/calcul' (arm64)
Число: list
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): Неправильно введено действие    inf
Process 6373 exited with status = 0 (0x00000000)
(lldb) list
   17
   18           Result = Calculate(Numeral, Operation);
   19           printf("%6.2f\n",Result);
   20
   21           return 0;
   22   }
(lldb)
(lldb) list calculate.c:20
   20       printf("Множитель: ");
   21       scanf("%f",&SecondNumeral);
   22       return(Numeral * SecondNumeral);
   23     } else if (strncmp(Operation, "/", 1) == 0) {
   24       printf("Делитель: ");
   25       scanf("%f",&SecondNumeral);
   26       if (SecondNumeral == 0) {
   27         printf("Ошибка: деление на ноль!");
   28         return(HUGE_VAL);
   29       }
   30       return(Numeral / SecondNumeral);
(lldb) b 21
Breakpoint 1: where = calcul`Calculate + 224 at calculate.c:21:5, address = 0x00
00000100003b40
(lldb)
```

```
(lldb) b
Current breakpoints:
1: file = '/Users/anastasia/Developer/lab13/calculate.c', line = 21, exact_match
 = 0, locations = 1
  1.1: where = calcul`Calculate + 224 at calculate.c:21:5, address = 0x000000010
0003b40, unresolved, hit count = 0

(lldb) run
Process 6463 launched: '/Users/anastasia/Developer/lab13/calcul' (arm64)
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): *
Process 6463 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x0000000100003b40 calcul`Calculate(Numeral=5, Operation="*") at c
alculate.c:21:5
   18       return(Numeral - SecondNumeral);
   19     } else if (strncmp(Operation, "*", 1) == 0) {
   20       printf("Множитель: ");
-> 21       scanf("%f",&SecondNumeral);
   22       return(Numeral * SecondNumeral);
   23     } else if (strncmp(Operation, "/", 1) == 0) {
   24       printf("Делитель: ");
Target 0: (calcul) stopped.
```

```
anastasia@Air-Anastasia lab13 % splint calculate.c
Splint 3.1.2 --- 24 Dec 2020

calculate.h:7:37: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
  A formal parameter is declared as an array with size.  The size of the array
  is ignored in this context, since the array formal parameter is treated as a
  pointer. (Use -fixedformalarray to inhibit warning)
calculate.c:9:37: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:13:5: Return value (type int) ignored: scanf("%f", &Sec...
  Result returned by function call is not used. If this is intended, can cast
  result to (void) to eliminate message. (Use -retvalint to inhibit warning)
calculate.c:17:5: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:21:5: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:25:5: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:26:9: Dangerous equality comparison involving float types:
                    SecondNumeral == 0
  Two real (float, double, or long double) values are compared directly using
  == or != primitive. This may produce unexpected results since floating point
  representations are inexact. Instead, compare the difference to FLT_EPSILON
  or DBL_EPSILON. (Use -realcompare to inhibit warning)
```

```
(lldb) p Numeral
(float) $0 = 5
```

```
[anastasia@Air-Anastasia lab13 % splint main.c                                  ]
Splint 3.1.2 --- 24 Dec 2020

calculate.h:7:37: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
  A formal parameter is declared as an array with size.  The size of the array
  is ignored in this context, since the array formal parameter is treated as a
  pointer. (Use -fixedformalarray to inhibit warning)
main.c: (in function main)
main.c:13:2: Return value (type int) ignored: scanf("%f", &Num...
  Result returned by function call is not used. If this is intended, can cast
  result to (void) to eliminate message. (Use -retvalint to inhibit warning)
main.c:16:13: Format argument 1 to scanf (%s) expects char * gets char [4] *:
                &Operation
  Type of parameter is not consistent with corresponding code in format string.
  (Use -formattype to inhibit warning)
   main.c:16:10: Corresponding format code
main.c:16:2: Return value (type int) ignored: scanf("%s", &Ope...

Finished checking --- 4 code warnings
anastasia@Air-Anastasia lab13 %
```

culcui: собираем в исполняемый файл .o : компилирует в объектный файл

на моей платворме, а именно MacBook на ARM процессоре не поддерживается утилита GDB поэтому я использую аналог - LLDB