

Лабораторная работа №14

Филиппова Анна Дмитриевна inst{1}

1 июня, 2021, Москва, Россия

¹RUDN University, Moscow, Russian Federation

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

1. В домашнем каталоге создаем подкаталог ~/work/os/lab_prog. (рис. -fig. 1)

```
[adfilippova@adfilippova ~]$ mkdir ~/work/os/lab_prog  
[adfilippova@adfilippova ~]$ █
```

Рис. 1: Создаем подкаталог

2. Создаем в каталоге файлы: `calculate.h`, `calculate.c`, `main.c`. Они необходимы для написания программ. (рис. -fig. 2)

```
[adfilippova@adfilippova lab_prog]$ touch calculate.h calculate.c main.c
[adfilippova@adfilippova lab_prog]$ ls
calculate.c calculate.h main.c
[adfilippova@adfilippova lab_prog]$ █
```

Рис. 2: Создаем файлы

3. Реализация функций калькулятора в файле calculate.c.

Интерфейсный файл calculate.h, описывающий формат вызова функции калькулятора. Основной файл main.c, реализующий интерфейс пользователя к калькулятору. (рис. -fig. 3)

```
////////////////////////////////////  
// main.c  
  
#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("%.2f\n",Result);  
    return 0;  
}
```

Рис. 3: Пишем командные файлы

4. Выполняем компиляцию программы посредством gcc. (рис. -fig. 4)

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

Рис. 4: Компиляция программы

5. Далее с помощью gdb выполняем отладку программы calcul. (рис. -fig. 5)

```
[adfilippova@adfilippova lab_prog]$ gdb ./calcul
GNU gdb (GDB) Red Hat Enterprise Linux 7.6.1-120.el7
Copyright (C) 2013 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".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from /home/adfilippova/work/os/lab_prog/calcul...done.
(gdb) █
```

Рис. 5: Запуск отладчика

6. С помощью утилиты splint анализируем коды файлов calculate.c и main.c. (рис. -fig. 6)

```
[adfilippova@adfilippova lab_prog]$ splint calculate.c
Splint 3.1.2 --- 11 Oct 2015

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:9:31: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:15:7: 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:21:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:27:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:33:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:34:10: 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)
calculate.c:37:10: Return value type double does not match declared type float:
    (HUGE_VAL)
```

Рис. 6: Splint

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