

# Презентация по лабораторной работе №14

---

Denis A. Seregin

2021, 05 June

<sup>1</sup>RUDN University, Moscow, Russian Federation

# Содержание

---

- Цель работы
- Задачи
- Результаты
- Вывод

## 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.

# **Основные понятия**

---

- Написать программу реализующую функции простейшего калькулятора
- С помощью утилиты splint попробуйте проанализировать коды файлов `calculate.c` и `main.c`

## Результаты

---



## Первая задача

В домашнем каталоге создал подкаталог `~/work/os/lab_prog`.  
Создал в нём файлы: `calculate.h`, `calculate.c`, `main.c`.

Выполнил компиляцию программы посредством `gcc`:

Исправил синтаксические ошибки

Создал `Makefile` с содержанием из указаний к лабораторной работе

Запустил отладчик `GDB`, загрузив в него программу для отладки

# Создание файлов и директории

```
[daseregin@localhost ~]$ mkdir ~/work/os/lab_prog  
[daseregin@localhost ~]$ cd work  
[daseregin@localhost work]$ cd os  
[daseregin@localhost os]$ cd lab_prog  
[daseregin@localhost lab_prog]$ touch calculate.h  
[daseregin@localhost lab_prog]$ vi calculate.h  
[daseregin@localhost lab_prog]$ touch calculate.h  
[daseregin@localhost lab_prog]$ touch main.c  
[daseregin@localhost lab_prog]$
```

**Figure 1:** Создание файлов и директории

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

**Figure 2:** Компиляция программы с помощью gcc

```
[daseregin@localhost 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/daseregin/work/os/lab_prog/calcul...(no debugging symbols found)...done.
(gdb)
```

**Figure 3:** Запуск калькулятора в отладчике GDB

```
(gdb) run
Starting program: /home/daseregin/work/os/lab_prog/./calcul
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 9
14.00
[Inferior 1 (process 7258) exited normally]
Missing separate debuginfos, use: debuginfo-install glibc-2.17-317.el7.x86_64
(gdb)
```

**Figure 4:** Работа команды run

## Вторая задача

С помощью утилиты splint попробовал проанализировать коды файлов `calculate.c` и `main.c`.

# Результат работы команды

```
[root@localhost lab_prog]# splint calculate.c
Splint 3.1.2 --- 11 Oct 2015

calculate.h:4:37: 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:7:31: Function parameter Operation declared as manifest array (size
        constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:13:1: 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:19:1: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:25:1: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:31:1: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:32:4: 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:35:7: Return value type double does not match declared type float:
        (HUGE_VAL)
    To allow all numeric types to match, use +relaxtypes.
calculate.c:43:1: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:44:7: Return value type double does not match declared type float:
        (pow(Numeral, SecondNumeral))
calculate.c:47:7: Return value type double does not match declared type float:
        (sqrt(Numeral))
calculate.c:49:7: Return value type double does not match declared type float:
        (sin(Numeral))
calculate.c:51:7: Return value type double does not match declared type float:
        (cos(Numeral))
calculate.c:53:7: Return value type double does not match declared type float:
        (tan(Numeral))
calculate.c:57:7: Return value type double does not match declared type float:
        (HUGE_VAL)

Finished checking --- 15 code warnings
```

# Результат работы команды

```
[root@localhost lab_prog]# splint main.c
Splint 3.1.2 --- 11 Oct 2015

calculate.h:4:37: 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:11:1: 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:13:1: Return value (type int) ignored: scanf("%s", Oper...

Finished checking --- 3 code warnings
```

**Figure 6:** Работа команды splint



В ходе лабораторной работы мне удалось 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.

**Спасибо за внимание!**