

# Презентация к лабораторной работе №13

Ермолаев А.М.

## Презентация к лабораторной работе №13

### 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.



## Выполнение работы

### Создание директории

```
f1.txt      monthly.00    sc2.sh~      Загрузки
hello.c     mv_files.sh    sc4.sh~      Изображения
[amermolaev@amermolaev ~]$ mkdir ~/work/os/lab_prog
[amermolaev@amermolaev ~]$ cd ~/work/os/lab_prog
[amermolaev@amermolaev lab_prog]$
```

*создание директории*

### Компиляция посредством gcc

```
amermolaev@amermolaev:~/work/os/lab_prog
[amermolaev@amermolaev lab_prog]$ gcc -c calculate.c
In file included from calculate.c:7:
calculate.h:4: ошибка: незавершённая #ifndef
    4 | #ifndef CALCULATE_H_
      |
calculate.h:9:6: ошибка: expected «;» before «float»
    9 | endif /*CALCULATE_H_*/
      |      ^
      |      ;
[amermolaev@amermolaev lab_prog]$ emacs calculate.c
[amermolaev@amermolaev lab_prog]$ emacs calculate.h
[amermolaev@amermolaev lab_prog]$ gcc -c calculate.c
[amermolaev@amermolaev lab_prog]$ gcc -c main.c
[amermolaev@amermolaev lab_prog]$ gcc calculate.o main.o -o calcul -lm
[amermolaev@amermolaev lab_prog]$ ./calcul
Число: 12
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): sin
-0.54
[amermolaev@amermolaev lab_prog]$
```

*компиляция*

### Пересборка проекта при помощи Makefile

```
amermolaev@amermolaev:~/work/os/lab_prog
[amermolaev@amermolaev lab_prog]$ make calculate.o
gcc -c calculate.c -g
[amermolaev@amermolaev lab_prog]$ make main.o
gcc -c main.c -g
[amermolaev@amermolaev lab_prog]$ make calcul
gcc calculate.o main.o -o calcul -lm
[amermolaev@amermolaev lab_prog]$
```

*сборка при помощи файла make*

## Отладка

```
amercolaev@amercolaev:~/work/os/lab_prog — /home/amer...
[amercolaev@amercolaev lab_prog]$ gdb ./calcul
GNU gdb (GDB) Fedora 12.1-1.fc36
Copyright (C) 2022 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".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
  <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./calcul...

This GDB supports auto-downloading debuginfo from the following URLs:
https://debuginfod.fedoraproject.org/
Enable debuginfod for this session? (y or [n]) y
Debuginfod has been enabled.
To make this setting permanent, add 'set debuginfod enabled on' to .gdbinit.
(No debugging symbols found in ./calcul)
```

запуск команды `gdb ./calcul`

```
(gdb) run
Starting program: /home/amercolaev/work/os/lab_prog/calcul

This GDB supports auto-downloading debuginfo from the following URLs:
https://debuginfod.fedoraproject.org/
Enable debuginfod for this session? (y or [n]) y
Debuginfod has been enabled.
To make this setting permanent, add 'set debuginfod enabled on' to .gdbinit.
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 12
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): 1
Неправильно введено действие   inf
[Inferior 1 (process 5002) exited normally]
(gdb) run
Starting program: /home/amercolaev/work/os/lab_prog/calcul
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 12
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 54
66.00
[Inferior 1 (process 5005) exited normally]
```

запуск команды `run`

```

(gdb) list
1  //////////////////////////////////////////
2  // main.c
3
4  #include <stdio.h>
5  #include "calculate.h"
6
7  int main (void){
8      float Numeral;
9      char Operation[4];
10     float Result;
(gdb) list 12,15
12     scanf("%f",&Numeral);
13     printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
14     scanf("%s",&Operation);
15     Result = Calculate(Numeral, Operation);
(gdb) list calculate.c:20,29
20     {
21         printf("Вычитаемое: ");
22         scanf("%f",&SecondNumeral);
23         return(Numeral - SecondNumeral);
24     }
25     else if(strncmp(Operation, "+", 1) == 0)
26     {
27         printf("Множитель: ");
28         scanf("%f",&SecondNumeral);
29         return(Numeral * SecondNumeral);
(gdb)

```

*использование команды list*

```

(gdb) break 21
Breakpoint 1 at 0x40120f: file calculate.c, line 21.
(gdb) info breakpoints
Num      Type             Disp Enb Address                  What
1        breakpoint     keep y   0x000000000040120f in Calculate
                                                at calculate.c:21

```

*установка точки останова*

```

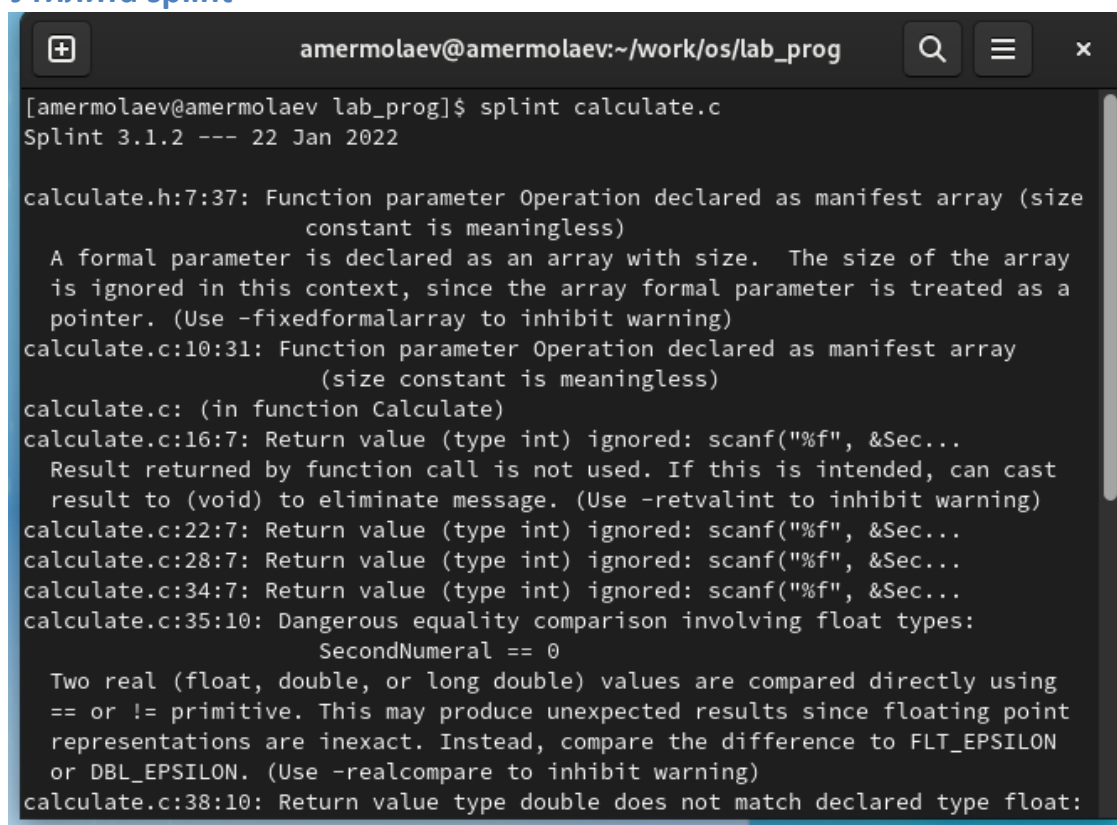
(gdb) run
Starting program: /home/amercolaev/work/os/lab_prog/calcul
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -

Breakpoint 1, Calculate (Numeral=5, Operation=0x7fffffffdf24 "-") at calculate.c:21
21      printf("Вычитаемое: ");
(gdb) backtrace
#0 Calculate (Numeral=5, Operation=0x7fffffffdf24 "-") at calculate.c:21
#1 0x00000000004014eb in main () at main.c:15
(gdb) print Numeral
$1 = 5
(gdb) display Numeral
1: Numeral = 5
(gdb) info breakpoints
Num   Type             Disp Enb Address                  What
1     breakpoint       keep y   0x000000000040120f in Calculate
                                     at calculate.c:21
                                     breakpoint already hit 1 time
(gdb) delete 1
(gdb) info breakpoints
No breakpoints or watchpoints.
(gdb)

```

*проверка корректности работы и удаление точки останова*

## Утилита splint



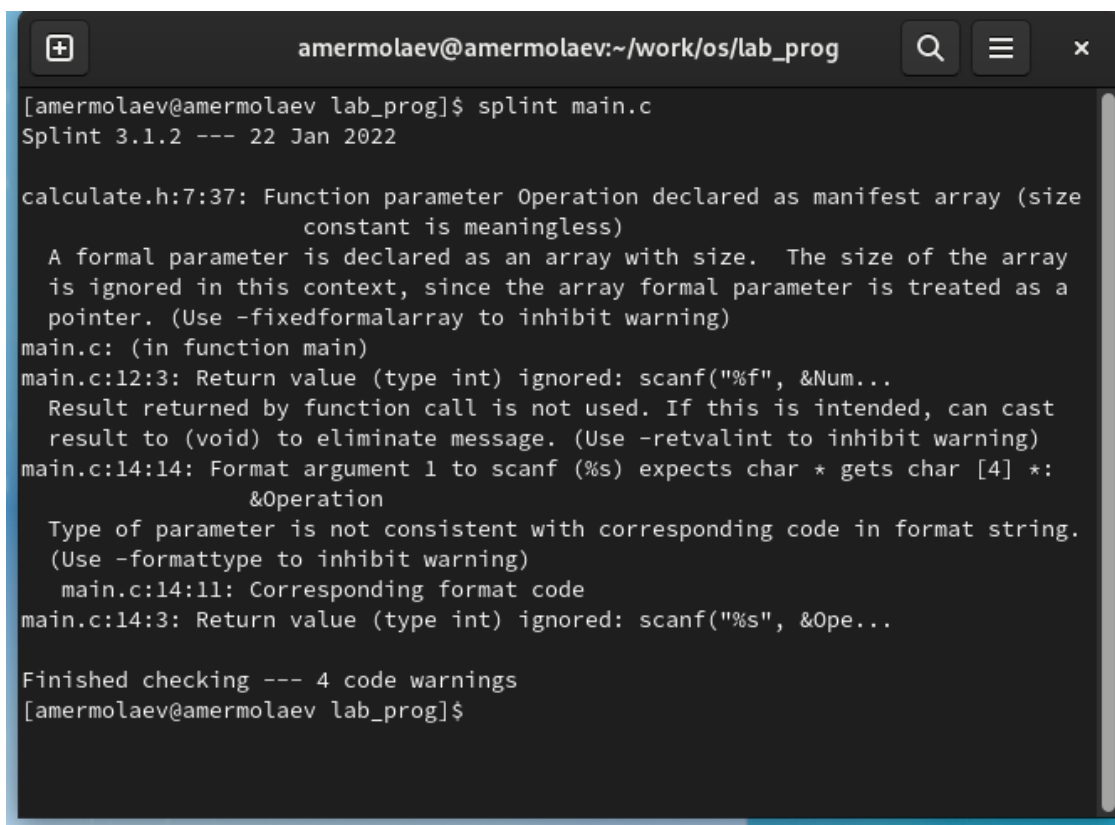
```

[amercolaev@amercolaev lab_prog]$ splint calculate.c
Splint 3.1.2 --- 22 Jan 2022

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:10:31: Function parameter Operation declared as manifest array
                    (size constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:16:7: 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:22:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:28:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:34:7: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:35:10: 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:38:10: Return value type double does not match declared type float:

```

*утилита splint*

A terminal window with a dark background and light text. The window title bar shows the user 'amermolaev' and the directory '~/work/os/lab\_prog'. The terminal content shows the execution of 'splint main.c', which produces several warnings about array sizes, ignored return values, and format string mismatches. It concludes with 'Finished checking --- 4 code warnings'.

```
amermolaev@amermolaev:~/work/os/lab_prog
[amermolaev@amermolaev lab_prog]$ splint main.c
Splint 3.1.2 --- 22 Jan 2022

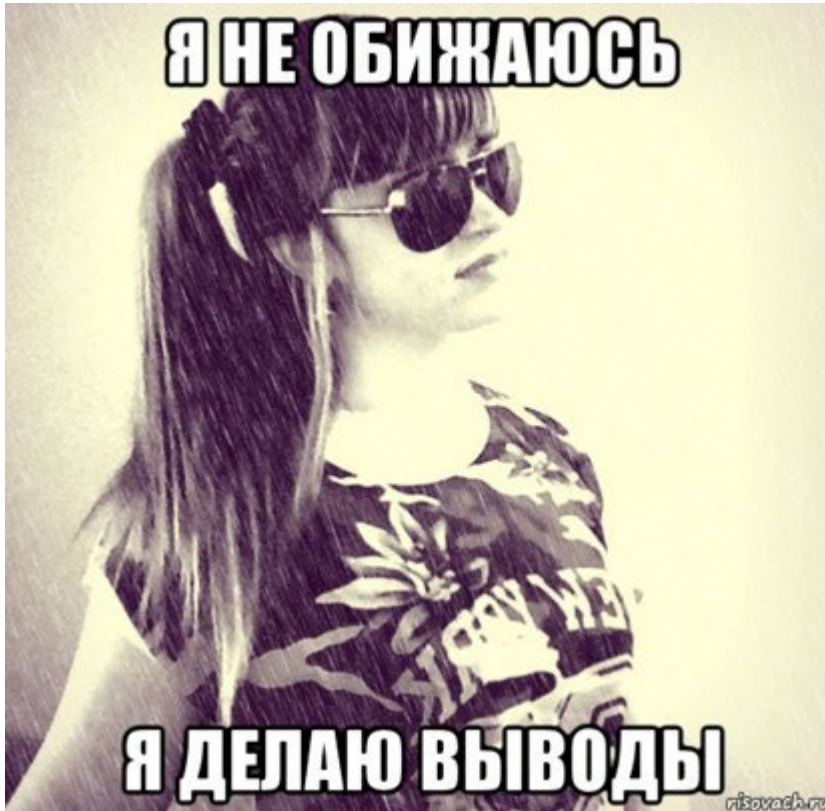
calculate.h:7:37: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
    A formal parameter is declared as an array with size.  The size of the array
    is ignored in this context, since the array formal parameter is treated as a
    pointer. (Use -fixedformalarray to inhibit warning)
main.c: (in function main)
main.c:12:3: 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:14:14: 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:14:11: Corresponding format code
main.c:14:3: Return value (type int) ignored: scanf("%s", &Ope...

Finished checking --- 4 code warnings
[amermolaev@amermolaev lab_prog]$
```

утилита *splint*

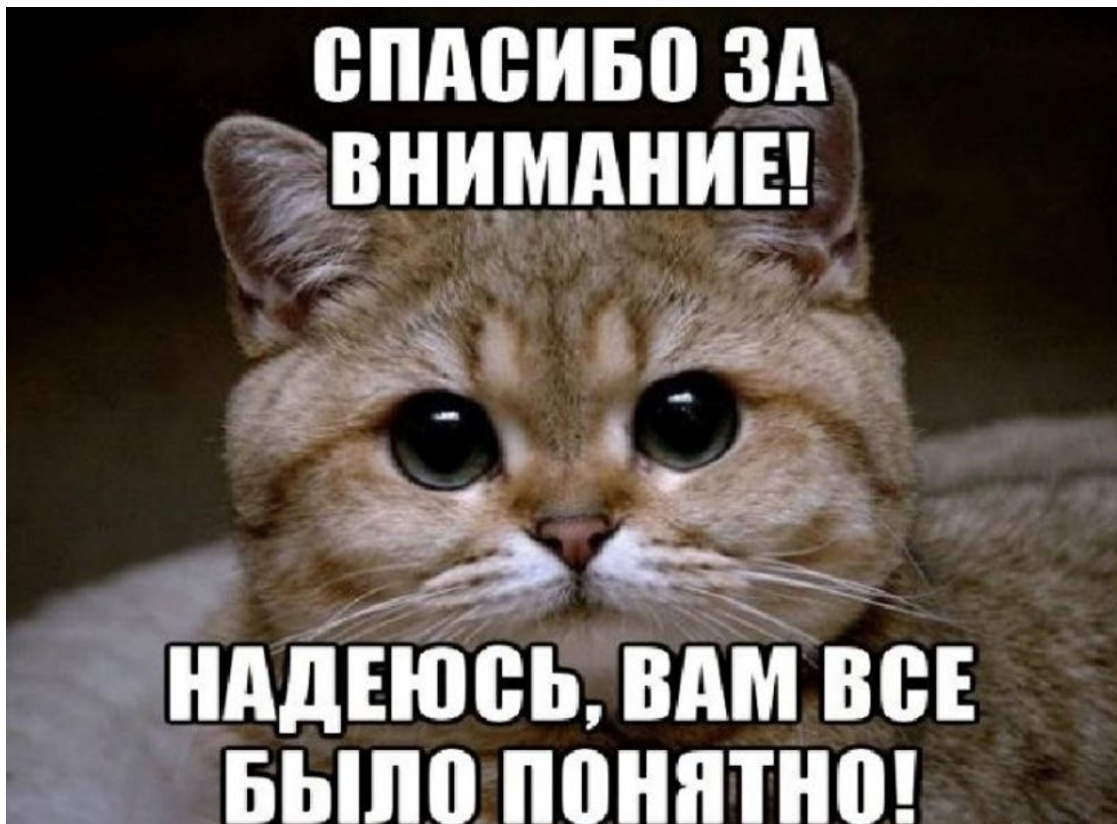
## Вывод

В рамках выполнения работы я приобрел простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования С калькулятора с простейшими функциями.





Финал



S