Презентация к лабораторной работе №13

Ермолаев А.М.

Презентация к лабораторной работе №13

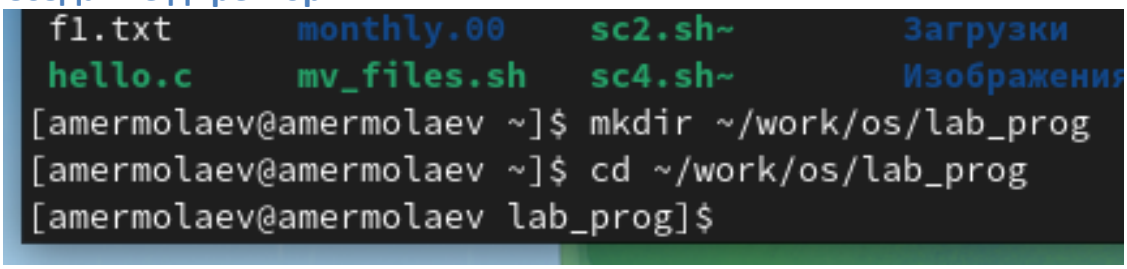
Цель работы

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.

s

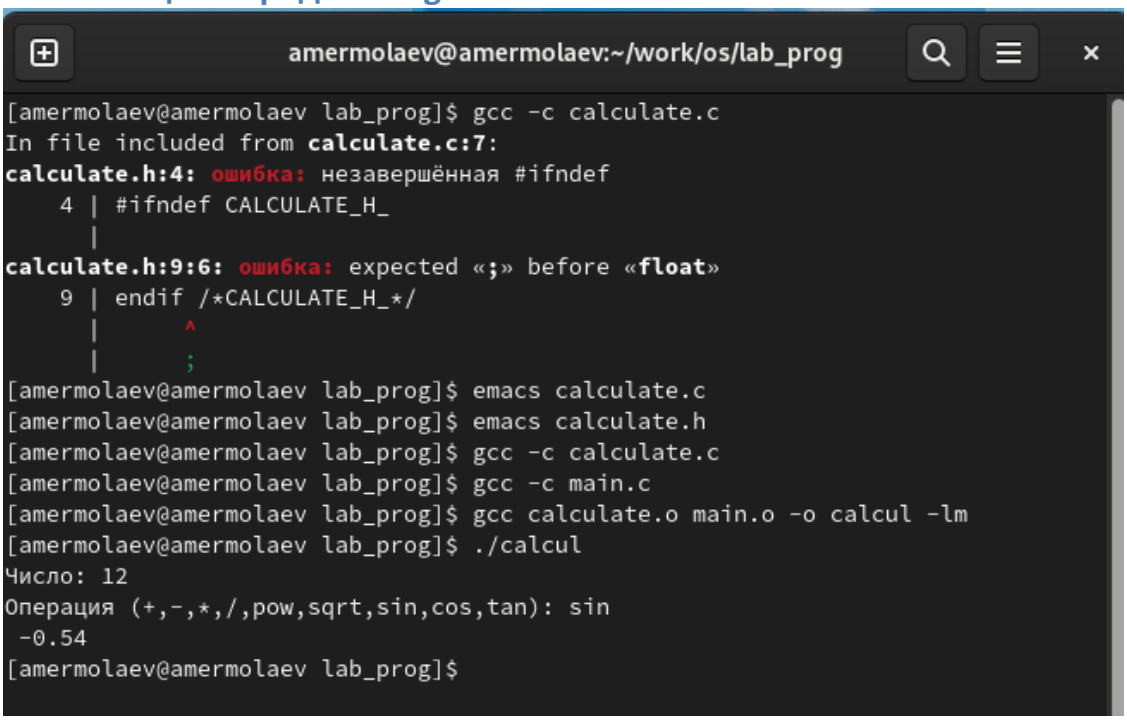
Выполнение работы

Создание директории



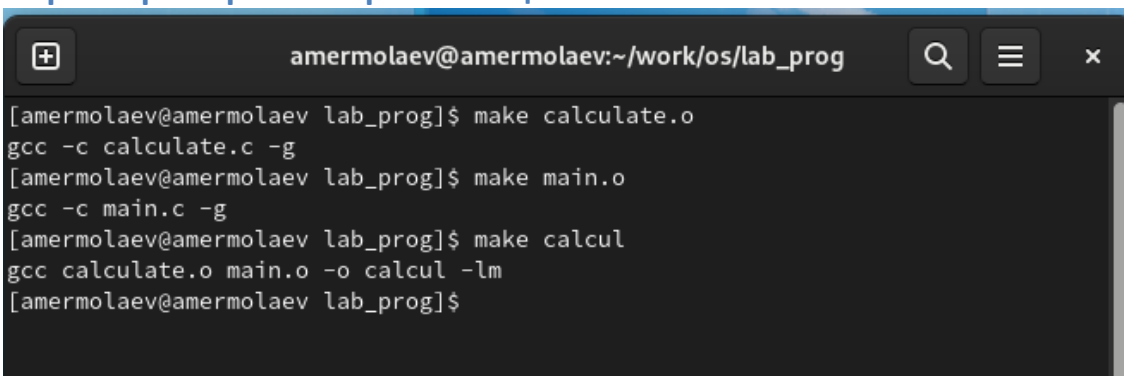
создание директории

Компиляция поредством gcc



компиляция

Пересборка проекта при помощи Makefile



сборка при помощи файла make

Отладка

```
amermolaev@amermolaev:~/work/os/lab_prog — /home/amer...
[amermolaev@amermolaev 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)
```

закпуск команды gdb ./calcul

```
(gdb) run
Starting program: /home/amermolaev/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/amermolaev/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

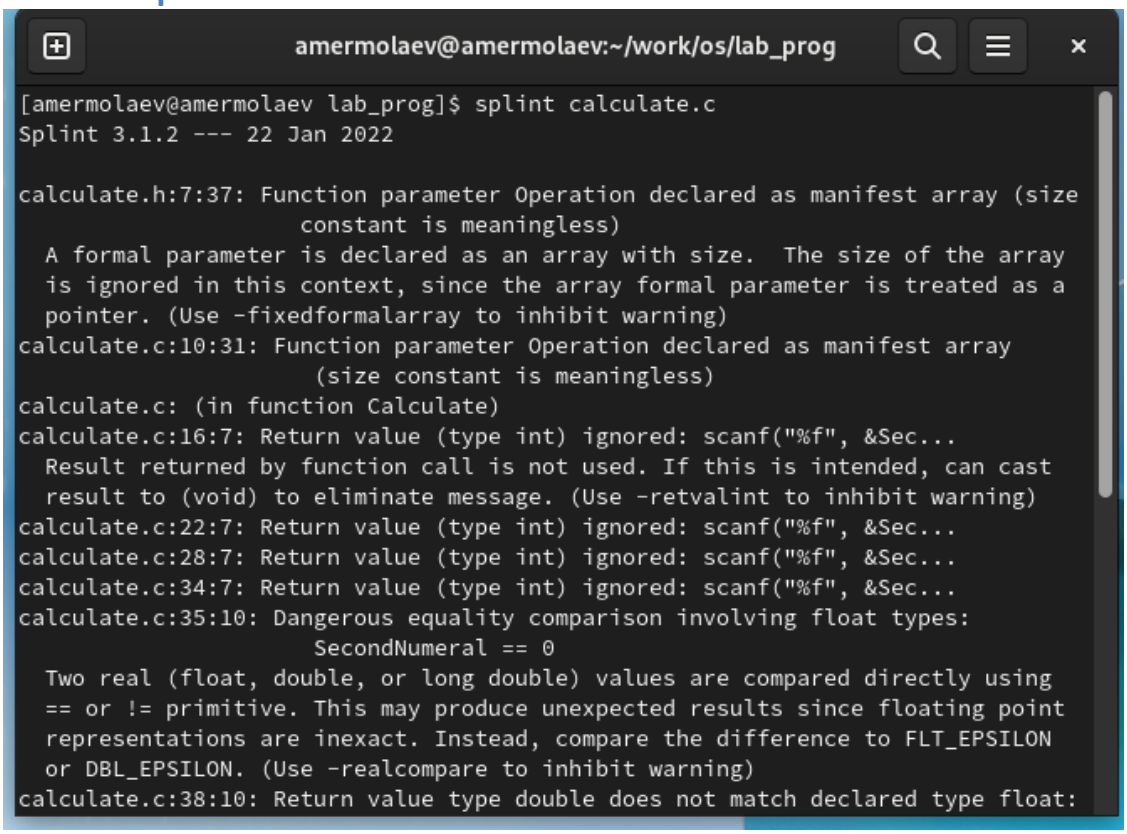
```
(gdb) break 21
Breakpoint 1 at 0x40120f: file calculate.c, line 21.
(gdb) info breakpoints
Num     Type           Disp Enb Address            What
1       breakpoint     keep y   0x000000000040120f in Calculate
                                                   at calculate.c:21
```

установка точки останова

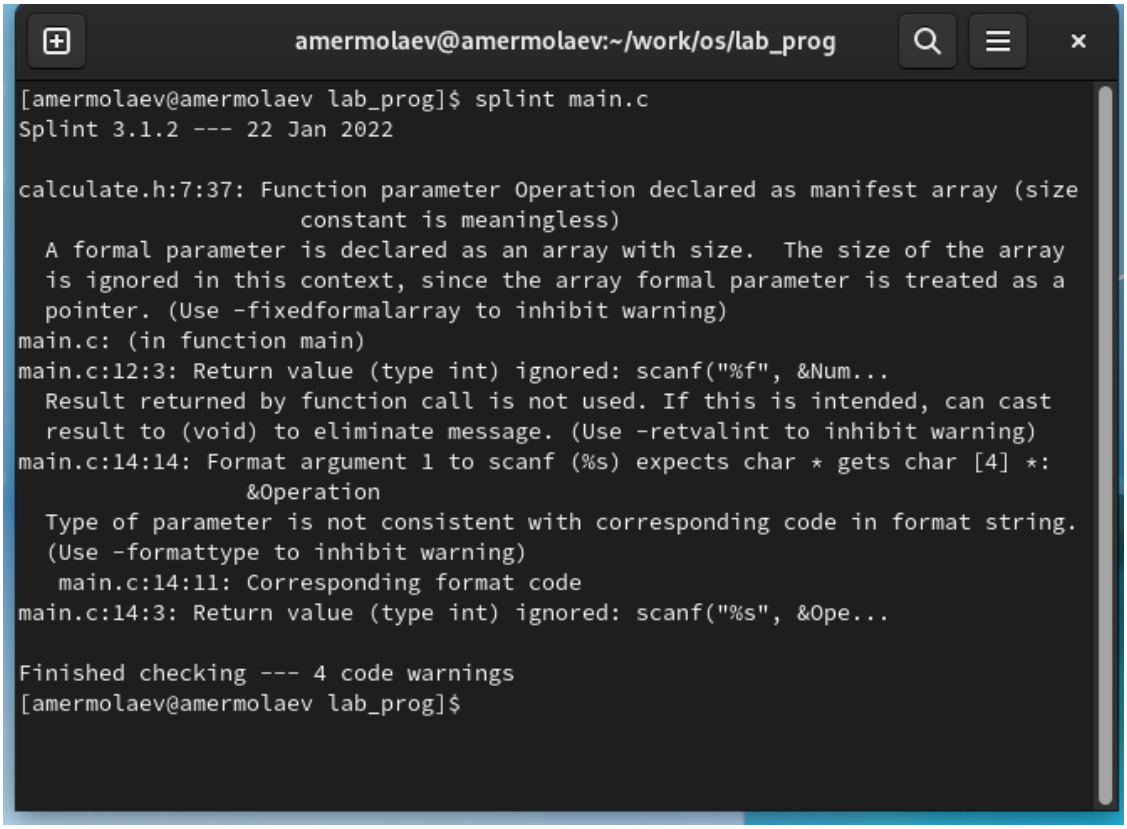
```
(gdb) run
Starting program: /home/amermolaev/work/os/lab_prog/calcul
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -

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 точчки останова

Утилита splint



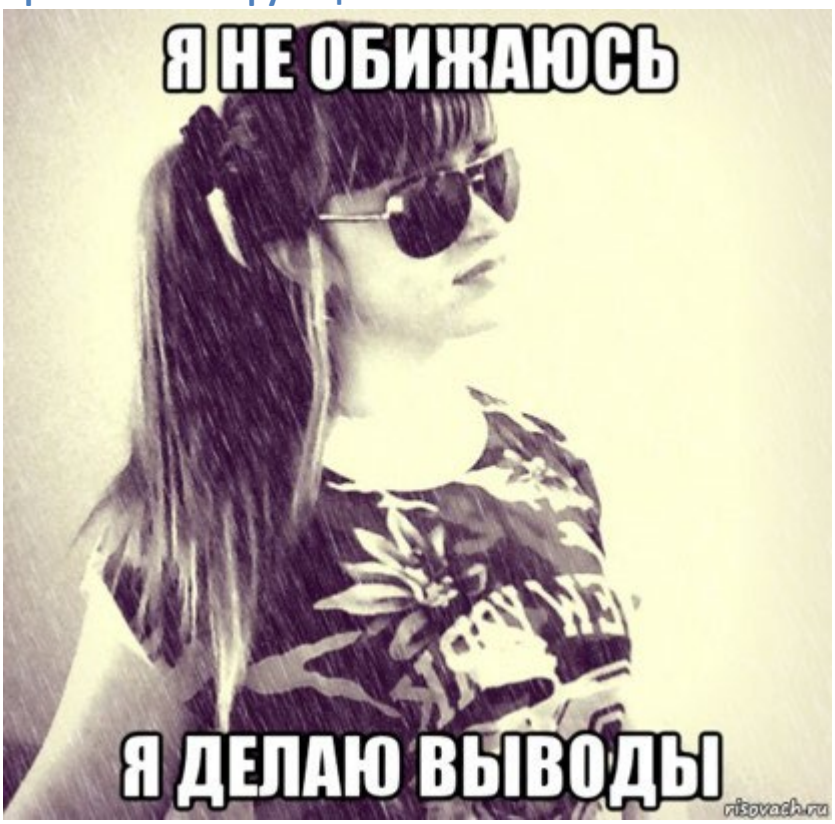
утилита splint



утилита splint

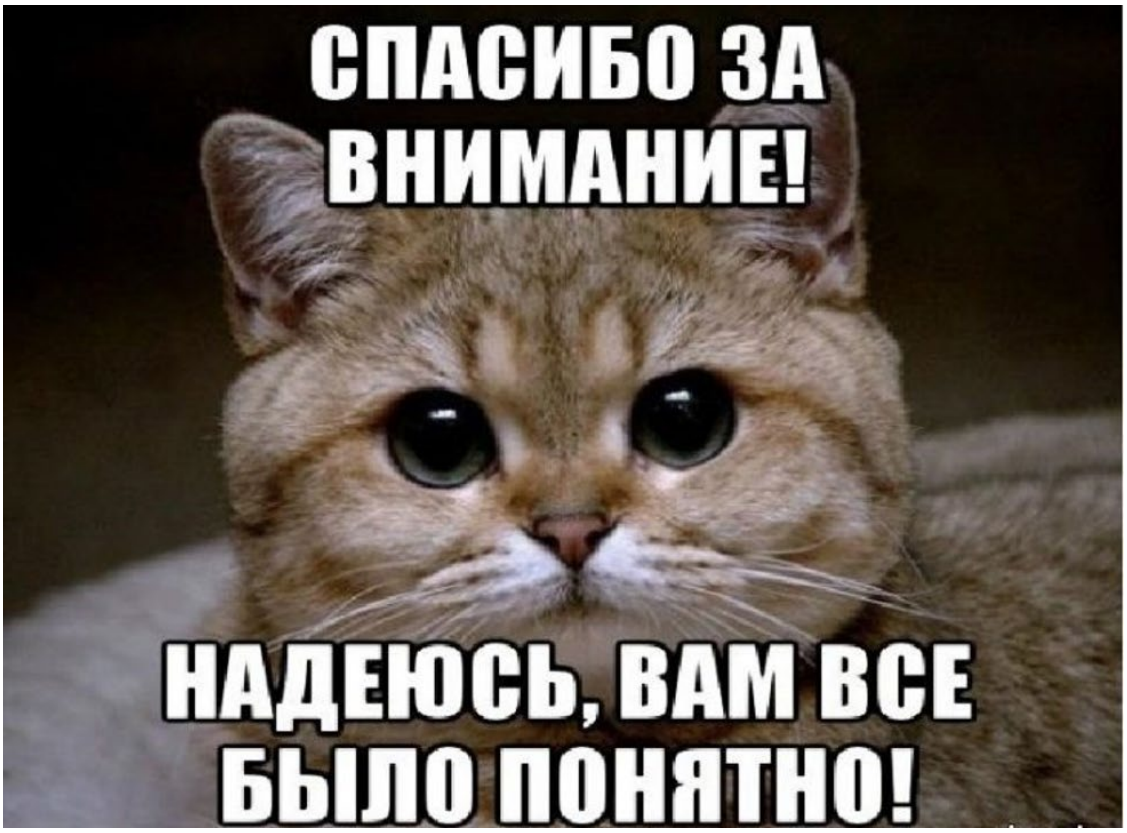
Вывод

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.



s

Финал



s