

Отчёт по лабораторной работе

**Средства, применяемые при разработке программного обеспечения в
ОС типа UNIX/Linux.**

Дедова Виктория Сергеевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	15
	Список литературы	16

Список иллюстраций

figno11	6
figno22	7
figno33	8
figno44	9
figno55	10
figno66	11
figno77	11
figno88	12
figno99	13
figno100	13

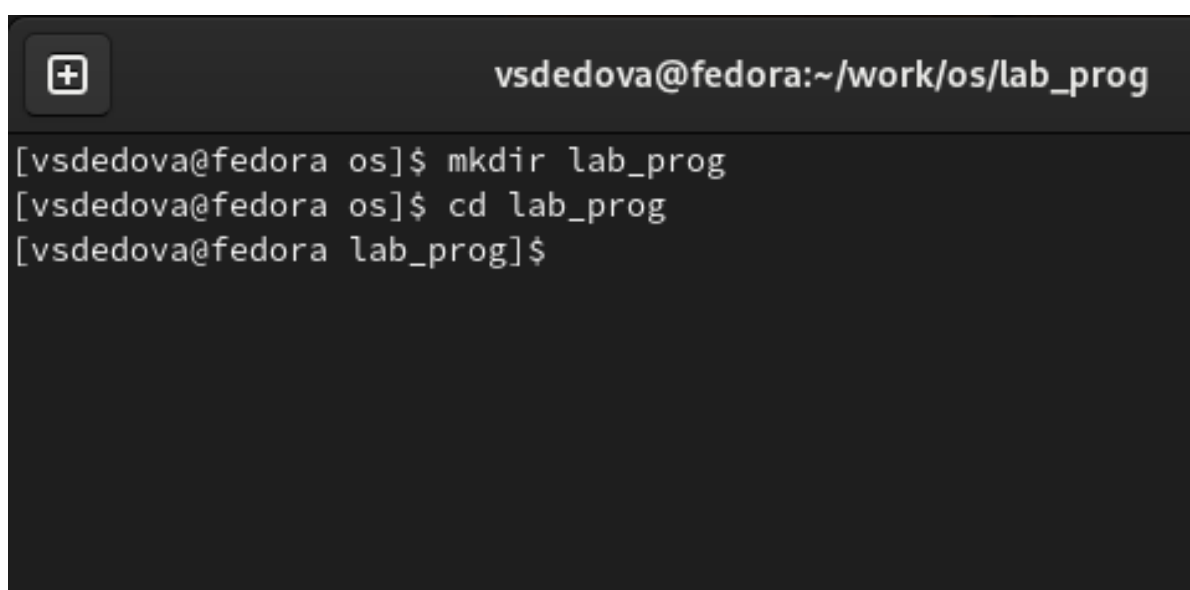
Список таблиц

1 Цель работы

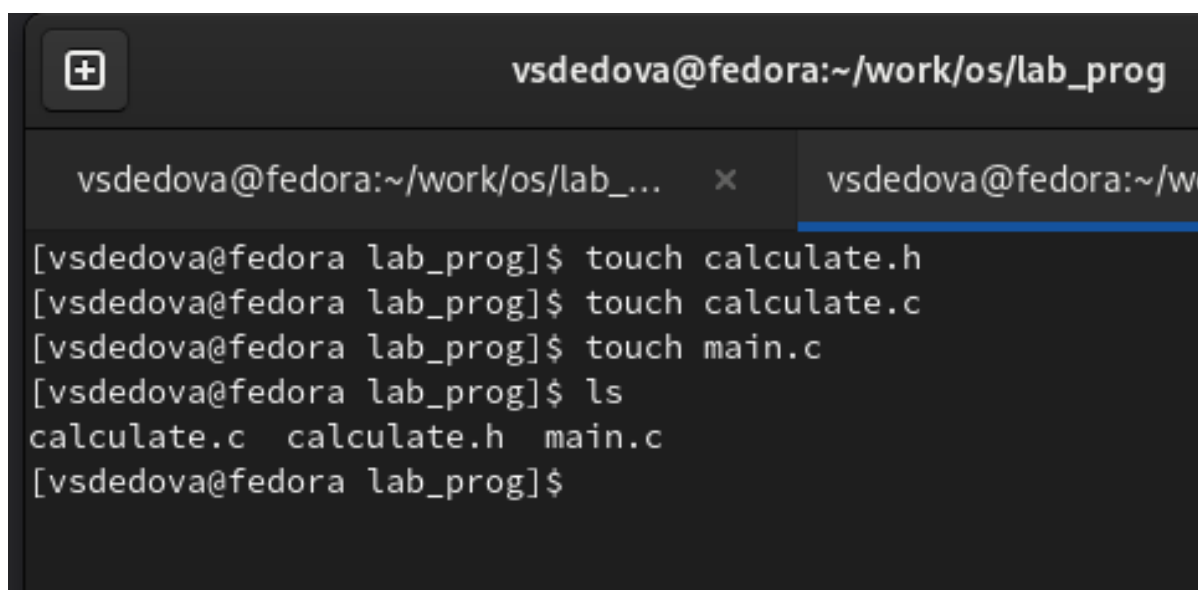
Приобрести простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования С калькулятора с простейшими функциями.

2 Выполнение лабораторной работы

В домашнем каталоге создайте подкаталог lab_prog.Создайте в нём файлы:calculate.h,calculate.c,main.c.



```
vsdedova@fedora:~/work/os/lab_prog  
[vsdedova@fedora os]$ mkdir lab_prog  
[vsdedova@fedora os]$ cd lab_prog  
[vsdedova@fedora lab_prog]$
```



```
vsdedova@fedora:~/work/os/lab_prog  
vsdedova@fedora:~/work/os/lab_prog x vsdedova@fedora:~/w  
[vsdedova@fedora lab_prog]$ touch calculate.h  
[vsdedova@fedora lab_prog]$ touch calculate.c  
[vsdedova@fedora lab_prog]$ touch main.c  
[vsdedova@fedora lab_prog]$ ls  
calculate.c calculate.h main.c  
[vsdedova@fedora lab_prog]$
```

2

Это будет примитивнейший калькулятор, способный выполнять математические операции. Реализация функций калькулятора в файле calculate.h:


7

```
calculate.c - GNU Emacs at fedora
File Edit Options Buffers Tools C Help
Save Undo
#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)
{
U:--- calculate.c Top L1 (C/*l Abbrev)
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

Emacs Tutorial Learn basic keystroke commands (Учебник Emacs)
Emacs Guided Tour Overview of Emacs features at gnu.org
View Emacs Manual View the Emacs manual using Info
```

Интерфейсный файл calculate.h, описывающий формат вызова функции-калькулятора:

 **vsdedova@fedora:~/work/os/lab_prog**
[vsdedova@fedora os]\$ mkdir lab_prog
[vsdedova@fedora os]\$ cd lab_prog
[vsdedova@fedora lab_prog]\$

4

Основной файл main.c, реализующий интерфейс пользователя к калькулятору:

9

```
main.c - GNU Emacs at fedora
File Edit Options Buffers Tools C Help
[Icons: New, Open, Save, Close, Save All, Undo, Redo, Copy, Paste]

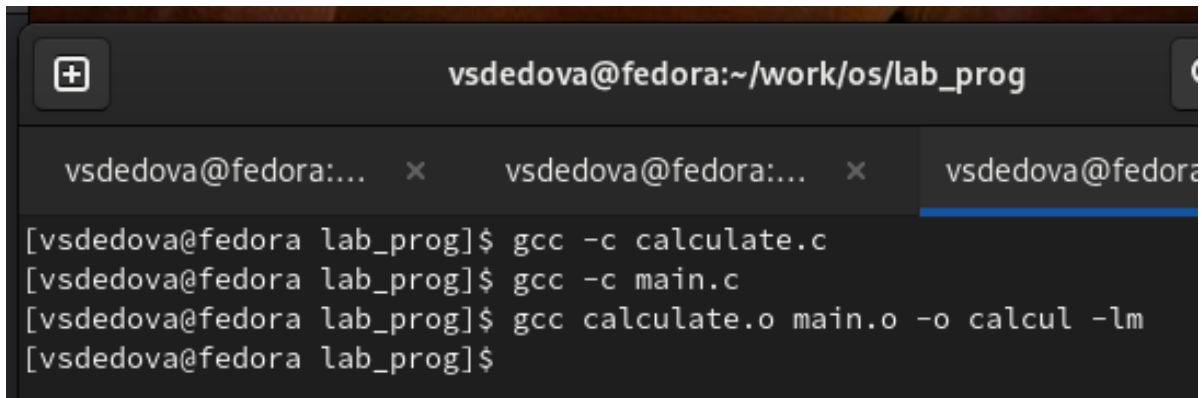
#include <stdio.h>
#include "calculate.h"
int
main (void)
{
    float Numeral;
    char Operation[4];
    float Result;
    printf("Число: ");
    scanf("%f",&Numeral);
    printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
    scanf("%s",&Operation);
    Result = Calculate(Numeral, Operation);
    printf("%6.2f\n",Result);
    return 0;
}

U:--- main.c All L1 (C/*l Abbrev)
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

Emacs Tutorial Learn basic keystroke commands (Учебник Emacs)
Emacs Guided Tour Overview of Emacs features at gnu.org
View Emacs Manual View the Emacs manual using Info
```

5

Выполните компиляцию программы посредством gcc:

A terminal window with a dark background. The title bar shows the user 'vsdedova' on a 'fedora' machine in the directory '~/work/os/lab_prog'. There are three tabs, all with the same text. The terminal shows the following commands and output:

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

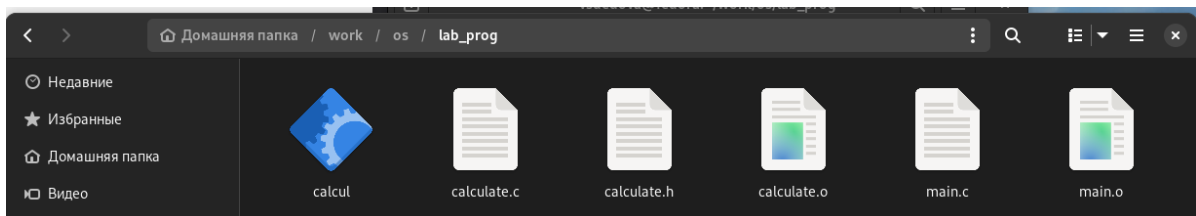
vsdedova@fedora:~/work/os/lab_prog

vsdedova@fedora:... x vsdedova@fedora:... x vsdedova@fedora

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

6

Создайте Makefile со следующим содержанием:



7

С помощью gdb выполните отладку программы calcul (перед использованием gdb исправьте Makefile):

```
Makefile - GNU Emacs
File Edit Options Buffers Tools Makefile Help
[Icons] Save Undo
CC = gcc
CFLAGS =
LIBS = -lm

calcul: calculate.o main.o
gcc calculate.o main.o -o calcul $(LIBS)

calculate.o: calculate.c calculate.h
gcc -c calculate.c $(CFLAGS)

main.o: main.c calculate.h
gcc -c main.c $(CFLAGS)

clean:
-rm calcul *.o *~

# End Makefile
-:--- Makefile All L1 (GNUmakefile)
Welcome to GNU Emacs, one component of the GNU/Linux system.

Emacs Tutorial Learn basic keystroke commands (Y4c)
Emacs Guided Tour Overview of Emacs features at gnu.org
View Emacs Manual View the Emacs manual using Info
Absence of Warranty GNU Emacs comes with ABSOLUTELY NO WARRANTY
Copying Conditions Conditions for redistributing and changing
Ordering Manuals Purchasing printed copies of manuals
To quit a partially entered command, type Control-g.
```

Для запуска программы внутри отладчика введите команду run:

```
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 3
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -
Вычитаемое: 2
  1.00
[Inferior 1 (process 3707) exited normally]
(gdb)
```

9

Для постраничного (по 9 строк) просмотра исходного код используйте команду list

```
[Inferior 1 (process 3707) exited normally]
(gdb) list
Downloading 0.00 MB source file /usr/src/debug/glibc-2.36-9.fc37.x86_64/elf/sofini.c
1      /* Terminate the frame unwind info section with a 4byte 0 as a sentinel;
2         this would be the 'length' field in a real FDE.  */
3
4      typedef unsigned int ui32 __attribute__((mode(SI)));
5      static const ui32 __FRAME_END__[1]
6          __attribute__((used, section(".eh_frame")))
7          = { 0 };

```

10

Уберите точки останова:

```
(gdb) delete 1
```

С помощью утилиты splint попробуйте проанализировать коды файлов

13

```

[vsdedova@fedora lab_prog]$ splint calculate.c
Splint 3.1.2 --- 23 Jul 2022

calculate.h:4:38: 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:7:32: Function parameter Operation declared as manifest array (size
                    constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:13:2: 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:19:2: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:25:2: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:31:2: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:32:5: 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 и main.c.

3 Выводы

На данной лабораторной работе мы приобрели простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования C калькулятора с простейшими функциями.

Список литературы