

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

Кеан Путхеаро НПИбд-01-20¹

3 июня, 2021, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

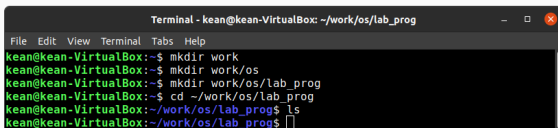
Цель лабораторной работы

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

Задачи лабораторной работы

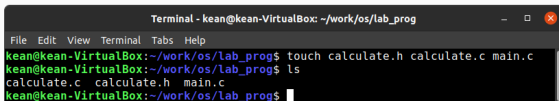
- 1 Написать код приложения
- 2 Выполнить компиляцию
- 3 Подготовить Makefile
- 4 Выполнить отладку в GDB
- 5 Проанализировать код при помощи splint

Процесс выполнения лабораторной работы

A terminal window titled "Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal shows the following commands and output:

```
kean@kean-VirtualBox:~$ mkdir work
kean@kean-VirtualBox:~$ mkdir work/os
kean@kean-VirtualBox:~$ mkdir work/os/lab_prog
kean@kean-VirtualBox:~$ cd ~/work/os/lab_prog
kean@kean-VirtualBox:~/work/os/lab_prog$ ls
kean@kean-VirtualBox:~/work/os/lab_prog$
```

Рис. 1: каталог

A terminal window titled "Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal shows the following commands and output:

```
kean@kean-VirtualBox:~/work/os/lab_prog$ touch calculate.h calculate.c main.c
kean@kean-VirtualBox:~/work/os/lab_prog$ ls
calculate.c  calculate.h  main.c
kean@kean-VirtualBox:~/work/os/lab_prog$
```

Рис. 2: файлы

calculate.h файл

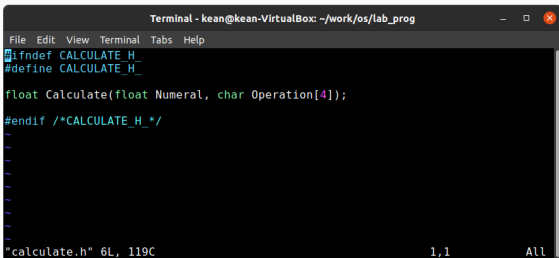
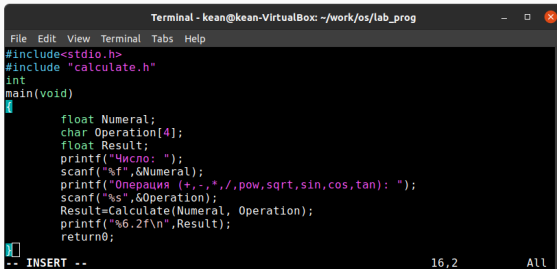


Рис. 3: calculate.h файл

calculate.c файл

```
Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog
File Edit View Terminal Tabs Help
#include<stdio.h>
#include<math.h>
#include<string.h>
#include"calculate.h"

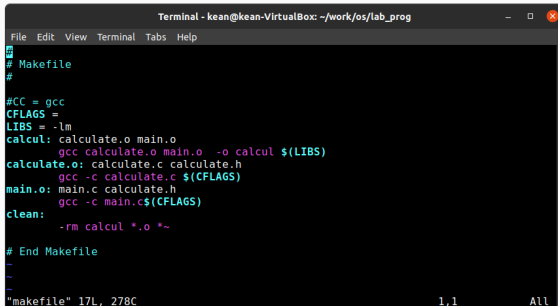
float Calculate(float Numeral,char Operation[4])
{
    float SecondNumeral;
    if(strncmp(Operation,"+",1) == 0)
    {
        printf("Второе слагаемое: ");
        scanf("%f",&SecondNumeral);
        return(Numeral+SecondNumeral);
    }
    else if(strncmp(Operation,"-",1) == 0)
    {
        printf("Вычитаемое: ");
        scanf("%f",&SecondNumeral);
        return(Numeral-SecondNumeral);
    }
    else if(strncmp(Operation,"*",1) == 0)
    {
        printf("Множитель: ");
        scanf("%f",&SecondNumeral);
        return(Numeral*SecondNumeral);
    }
    else if(strncmp(Operation,"/",1) == 0)
    {
        printf("Делитель: ");
        scanf("%f",&SecondNumeral);
        if(SecondNumeral==0)
        {
            printf("Ошибка: деление на ноль! ");
            return(HUGE_VAL);
        }
        else
            return(Numeral/SecondNumeral);
    }
    else if(strncmp(Operation,"pow",3) == 0)
    {
        printf("Степень: ");
        scanf("%f",&SecondNumeral);
        return(pow(Numeral, SecondNumeral));
    }
    else if(strncmp(Operation,"sqrt",4) == 0)
        return(sqrt(Numeral));
    else if(strncmp(Operation,"sin",3) == 0)
        return(sin(Numeral));
    else if(strncmp(Operation,"cos",3) == 0)
```

A screenshot of a terminal window titled "Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog". The terminal displays the C code for main.c. The code includes stdio.h and calculate.h, declares an integer for the main function, and defines a main function that takes no arguments. Inside main, it declares a float Numeral, a char array Operation of size 4, and a float Result. It then prints "Число: ", reads a float into Numeral, prints "Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ", reads a string into Operation, calls Calculate(Numeral, Operation), prints the result with 6.2f precision, and returns 0. The terminal status bar at the bottom shows "-- INSERT --", "16,2", and "All".

```
Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog
File Edit View Terminal Tabs Help
#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);
    return0;
}
```

Рис. 5: main.c файл

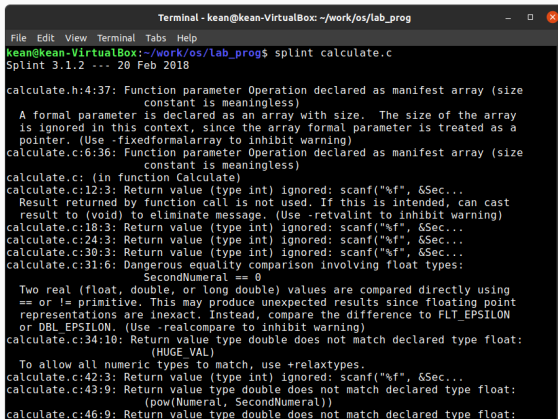
Makefile

A screenshot of a terminal window titled "Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog". The terminal displays the content of a Makefile. The Makefile defines variables for the compiler (gcc), flags, and libraries, and includes rules for building the 'calcul' target, the 'calculate.o' object file, the 'main.o' object file, and a 'clean' target to remove the output files. The status bar at the bottom of the terminal shows "makefile" 17L, 278C, 1,1, and All.

```
Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog
File Edit View Terminal Tabs Help
#
# Makefile
#
#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" 17L, 278C 1,1 All
```

Рис. 6: Makefile



```
Terminal - kean@kean-VirtualBox: ~/work/os/lab_prog
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~/work/os/lab_prog$ splint calculate.c
Splint 3.1.2 --- 20 Feb 2018

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:6:36: Function parameter Operation declared as manifest array (size
        constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:12:3: 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:18:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:24:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:30:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:31:6: 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:34:10: Return value type double does not match declared type float:
        (HUGE_VAL)
    To allow all numeric types to match, use +relaxtypes.
calculate.c:42:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:43:9: Return value type double does not match declared type float:
        (pow(Numeral, SecondNumeral))
calculate.c:46:9: Return value type double does not match declared type float:
```

Рис. 7: splint calculate.c

Выводы по проделанной работе

В результате работы , я приобрёл простейшие навыки разработки, анализа, тестирования и отладки приложений в Линукс

Библиография

1. (Лабораторная работа №14)
https://esystem.rudn.ru/pluginfile.php/1142386/mod_resource/content/1/lab_prog.pdf
2. (stackoverflow)
<https://vi.stackexchange.com/questions/10209/execute-current-buffer-as-bash-script-from-vim>
3. (BASH: функция getopt — используем опции в скриптах)
https://esystem.rudn.ru/pluginfile.php/1142380/mod_resource/content/1/lab_shell_prog_2.pdf
4. (stackoverflow)
<https://stackoverflow.com/questions/16483119/an-example-of-how-to-use-getopts-in-bash>