

Лабораторная работа №13

Коняева Марина НФИбд-01-21

01.06.2022

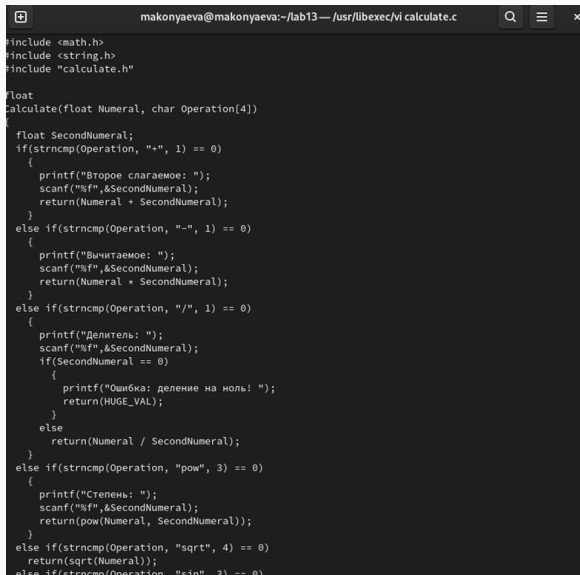
Средства, применяемые при разработке программного обеспечения в ОС типа UNIX/Linux

- 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.

Процесс разработки программного обеспечения обычно разделяется на следующие этапы: – планирование, включающее сбор и анализ требований к функционалу и другим характеристикам разрабатываемого приложения; – проектирование, включающее в себя разработку базовых алгоритмов и спецификаций, определение языка программирования; – непосредственная разработка приложения: – кодирование — по сути создание исходного текста программы (возможно в нескольких вариантах); – анализ разработанного кода; – сборка, компиляция и разработка исполняемого модуля; – тестирование и отладка, сохранение произведённых изменений; – документирование. Для создания исходного текста программы разработчик может воспользоваться любым удобным для него редактором текста: vi, vim, mceditor, emacs, geany и др. После завершения

Выполнение лабораторной работы

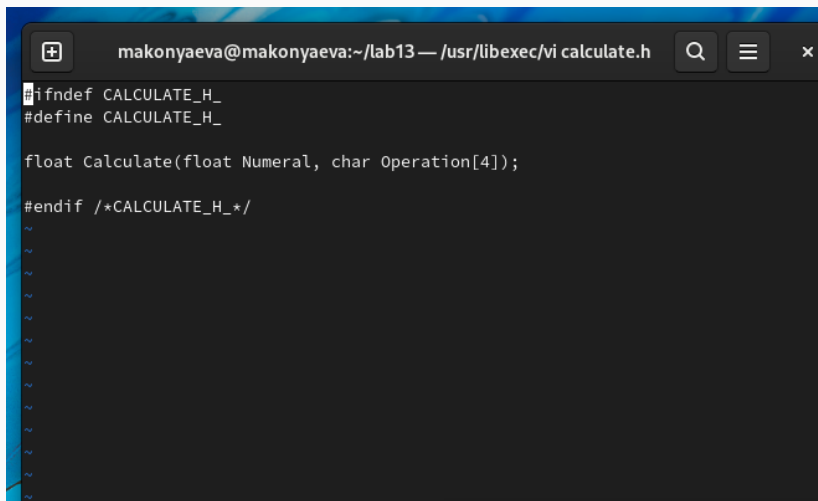
1. Скрипт 1 (изображение 1.1)



```
makonyaeva@makonyaeva:~/lab13 — /usr/libexec/vi calculate.c
#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("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));
    }
}
```

2. Скрипт 2 (изображение 2.1)



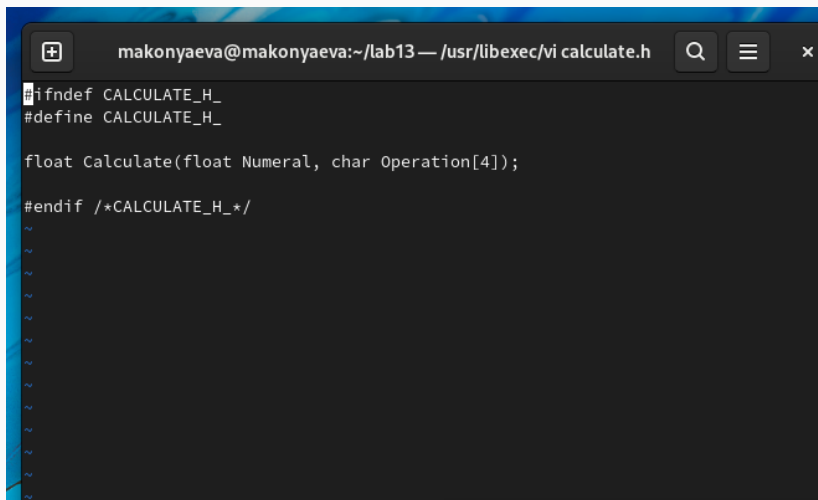
```
makonyaeva@makonyaeva:~/lab13 — /usr/libexec/vi calculate.h
#ifndef CALCULATE_H_
#define CALCULATE_H_

float Calculate(float Numeral, char Operation[4]);

#endif /*CALCULATE_H_*/
~
~
~
~
~
~
~
~
~
~
~
```

Изображение 2.1 Скрипт 2

3. Скрипт 3 (изображение 3.1)



```
#ifndef CALCULATE_H_
#define CALCULATE_H_

float Calculate(float Numeral, char Operation[4]);

#endif /*CALCULATE_H_*/
~
~
~
~
~
~
~
~
~
~
~
```

Изображение 3.1 Скрипт 3

В ходе данной лабораторной работы научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов, а также ответили на контрольные вопросы.