

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ  
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное  
бюджетное образовательное учреждение высшего образования  
**«Московский технический университет связи и информатики»**

---

Кафедра «Информатика»

**Лабораторная работа №4**  
**« Изучение структуры программного кода**  
**консольных проектов Visual Studio и правил**  
**определения, объявления и вызова функций**  
**VC++»**

**по дисциплине**  
**«Введение в информационные технологии»**

Выполнил: студент гр. БКС2501 Грачев Б.С..

Проверил:                    доц.                    Халабия                    Р.Ф.

Москва, 2025 г

# **тчет по выполнению лабораторной работы**

## **1.1 Общее задание**

Изучить структуру программного кода консольных проектов Visual Studio и правила определения, объявления и вызова функций VC++. Создать консольное решение, содержащее пять проектов, каждый из которых реализует вычисление заданного арифметического выражения различными способами организации функций.

## **1.2 Индивидуальное задание**

Вычислить значение выражения:

$$d = \frac{xy^2 - \sqrt{|x^2 - 2,5 \cdot 10^{-3}y|}}{2 \sin xy} + 0,5$$

при значениях:

$x = 1.3802$

$y = -1.9$

## **1.3 Формализация и уточнение задания:**

Для вычисления заданного выражения необходимо разработать три варианта схем алгоритмов и соответствующие процедуры:

- 1. Функция с входными параметрами и возвращаемым значением**
- 2. Функция с входными и выходными параметрами без возвращаемого значения**
- 3. Функция без параметров и без возвращаемого значения (с глобальными переменными)**

Эти варианты должны быть реализованы в пяти различных проектах:

**Проект 1:** Функция с параметрами и возвращаемым значением (определение перед main)

**Проект 2:** Функция с параметрами и возвращаемым значением (определение после main)

**Проект 3:** Функция с параметрами и без возвращаемого значения

**Проект 4:** Функция без параметров и без возвращаемого значения

## Проект 5: Функция с параметрами и возвращаемым значением (в разных файлах)

### 2 Программные проекты:

#### Проект 1:

Функция с параметрами и возвращаемым значением (определение перед main) на рисунке 2.1

```
#include <iostream>
#include <cmath>
#include <iomanip>

using namespace std;

double calculateExpression(double x, double y) {
    double numerator = x * y * y - sqrt(fabs(x * x - 2.5e-3 * y));
    double denominator = 2 * sin(x * y);

    if (fabs(denominator) < 1e-10) {
        cout << "Ошибка: деление на ноль!" << endl;
        return 0;
    }

    return numerator / denominator + 0.5;
}

int main() {
    setlocale(LC_ALL, "rus");
    double x = 1.3802;
    double y = -1.9;

    cout << "Проект 1: Функция с параметрами и возвращаемым значением" << endl;
    cout << "x = " << x << endl;
    cout << "y = " << y << endl;

    double result = calculateExpression(x, y);

    cout << fixed << setprecision(6);
    cout << "Результат: d = " << result << endl;

    return 0;
}
```

Рисунок 2.1 - Функция с параметрами и возвращаемым значением.

Результаты выполнения данной работы на рисунке 2.2

```
Проект 1: Функция с параметрами и возвращаемым значением
x = 1.3802
y = -1.9
Результат: d = -3.128201
```

Рисунок 2.2 - Результаты выполнения кода.

#### Проект 2:

Функция с параметрами и возвращаемым значением на рисунке 2.3

```

    int main() {
        setlocale(LC_ALL, "rus");
        // Установка исходных данных
        double x = 1.3802;
        double y = -1.9;

        cout << "Проект 2: Функция с параметрами и возвращаемым значением (определение после main)" << endl;
        cout << "x = " << x << endl;
        cout << "y = " << y << endl;

        // Вызов функции и получение результата
        double result = calculateExpression(x, y);

        // Вывод результата с заданной точностью
        cout << fixed << setprecision(6);
        cout << "Результат: d = " << result << endl;

        return 0;
    }

    // Определение функции (реализация)
    double calculateExpression(double x, double y) {
        // Вычисление подкоренного выражения с использованием модуля
        double underRoot = fabs(x * x - 2.5e-3 * y);

        // Вычисление числителя:  $xy^2 - \sqrt{|x^2 - 0.0025y|}$ 
        double numerator = x * y * y - sqrt(underRoot);

        // Вычисление знаменателя:  $2\sin(xy)$ 
        double denominator = 2 * sin(x * y);

        // Проверка деления на ноль
        if (fabs(denominator) < 1e-10) {
            cout << "Ошибка: деление на ноль!" << endl;
            return 0;
        }

        // Вычисление конечного результата
        return numerator / denominator + 0.5;
    }
}

```

Рисунок 2.3 - Функция с параметрами и возвращаемым значением.

Результаты выполнения кода на рисунке 2.4

```

Проект 2: Функция с параметрами и возвращаемым значением (определение после main)
x = 1.3802
y = -1.9
Результат: d = -3.128201

```

Рисунок 2.4 - Результаты выполнения кода.

### Проект 3:

Функция с параметрами и без возвращаемого значения на рисунке 2.5

```
#include <iostream>
#include <cmath>
#include <iomanip>

using namespace std;

// Прототип функции с выходным параметром (передача по ссылке)
void calculateExpression(double x, double y, double& result);

int main() {
    setlocale(LC_ALL, "rus");
    // Установка исходных данных
    double x = 1.3882;
    double y = -1.9;
    double result; // Переменная для хранения результата

    cout << "Проект 3: Функция с параметрами и без возвращаемого значения" << endl;
    cout << "x = " << x << endl;
    cout << "y = " << y << endl;

    // Вызов функции с передачей результата через параметр
    calculateExpression(x, y, result);

    // Вывод результата
    cout << fixed << setprecision(6);
    cout << "Результат: d = " << result << endl;

    return 0;
}

// Определение функции с выходным параметром
void calculateExpression(double x, double y, double& result) {
    // Вычисление подкоренного выражения с использованием модуля
    double underRoot = fabs(x * x - 2.5e-3 * y);

    // Вычисление числителя: xy3 - ∫|x2 - 0.0025y|
    double numerator = x * y * y - sqrt(underRoot);

    // Вычисление знаменателя: 2sin(xy)
    double denominator = 2 * sin(x * y);

    // Проверка деления на ноль
    if (fabs(denominator) < 1e-10) {
        cout << "Ошибка: деление на ноль!" << endl;
        result = 0;
        return; // Выход из функции без возврата значения
    }

    // Запись результата в выходной параметр
    result = numerator / denominator + 0.5;
}
```

Рисунок 2.5 - Функция с параметрами и без возвращаемого значения.

Результаты выполнения программы на рисунке 2.6

```
Проект 3: Функция с параметрами и без возвращаемого значения
x = 1.3882
y = -1.9
Результат: d = -3.128201
```

Рисунок 2.6 - Результаты выполнения программы.

## Проект 4:

Функция без параметров и без возвращаемого значения на рисунке 2.7

```
#include <iostream>
#include <cmath>
#include <iomanip>

using namespace std;

// Глобальные переменные – доступны во всех функциях
double x, y, result;

// Прототип функции без параметров
void calculateExpression();

int main() {
    setlocale(LC_ALL, "rus");
    // Инициализация глобальных переменных
    x = 1.3802;
    y = -1.9;

    cout << "Проект 4: Функция без параметров и без возвращаемого значения (с глобальными переменными)" << endl;
    cout << "x = " << x << endl;
    cout << "y = " << y << endl;

    // Вызов функции без параметров
    calculateExpression();

    // Вывод результата из глобальной переменной
    cout << fixed << setprecision(6);
    cout << "Результат: d = " << result << endl;

    return 0;
}

// Определение функции без параметров
void calculateExpression() {
    // Использование глобальных переменных x, y, result
    // Вычисление подкоренного выражения с использованием модуля
    double underRoot = fabs(x * x - 2.5e-3 * y);

    // Вычисление числителя:  $xy^3 - \sqrt{|x^2 - 0.0025y|}$ 
    double numerator = x * y * y - sqrt(underRoot);

    // Вычисление знаменателя:  $2\sin(xy)$ 
    double denominator = 2 * sin(x * y);

    // Проверка деления на ноль
    if (fabs(denominator) < 1e-10) {
        cout << "Ошибка: деление на ноль!" << endl;
        result = 0;
        return;
    }

    // Запись результата в глобальную переменную
    result = numerator / denominator + 0.5;
}
```

Рисунок 2.7 - Функция без параметров и без возвращаемого значения

Результаты выполнения программы на рисунке 2.8

```
Проект 4: Функция без параметров и без возвращаемого значения (с глобальными переменными)
x = 1.3802
y = -1.9
Результат: d = -3.128201
```

Рисунок 2.8 - Результаты выполнения программы.

Проект

5:  
Функция с параметрами и возвращаемым значением (в разных файлах) на

рисунке 2.9, 2.10, 2.11

```
#ifndef FUNCTIONS_H
#define FUNCTIONS_H

double calculateExpression(double x, double y);

#endif
```

Рисунок 2.9 - Функция с параметрами и возвращаемым значением (файл 1)

```
#include "functions.h"
#include <cmath>
#include <iostream>

using namespace std;

double calculateExpression(double x, double y) {
    double numerator = x * y * y - sqrt(fabs(x * x - 2.5e-3 * y));
    double denominator = 2 * sin(x * y);

    if (fabs(denominator) < 1e-10) {
        cout << "Ошибка: деление на ноль!" << endl;
        return 0;
    }

    return numerator / denominator + 0.5;
}
```

Рисунок 2.10 - Функция с параметрами и возвращаемым значением (файл 2)

```
#include <iostream>
#include <iomanip>
#include "functions.h" // Убедитесь, что файл в той же папке

using namespace std;

int main() {
    setlocale(LC_ALL, "rus");
    double x = 1.3802;
    double y = -1.9;

    cout << "Проект 5: Функция с параметрами и возвращаемым значением (в разных файлах)" << endl;
    cout << "x = " << x << endl;
    cout << "y = " << y << endl;

    double result = calculateExpression(x, y);

    cout << fixed << setprecision(6);
    cout << "Результат: d = " << result << endl;

    return 0;
}
```

Рисунок 2.11 - Функция с параметрами и возвращаемым значением (файл 3)

Результаты выполнения кода на рисунке 2.12

```
Проект 5: Функция с параметрами и возвращаемым значением (в разных файлах)
x = 1.3802
y = -1.9
Результат: d = -3.128201
```

Рисунок 2.12 - Результат выполнения кода.

3

Эксперименты:

Эксперимент 1: Проверка деления на ноль  
Проверка деления на ноль на рисунке 3.1

```
Проект 1: Функция с параметрами и возвращаемым значением
x = 0
y = -1.9
Ошибка: деление на ноль!
Результат: d = 0.000000
```

Рисунок 3.1 - проверка деления на ноль.

Эксперимент 2: Изменение порядка параметров  
Изменение порядка параметров на рисунке 3.2

```
Проект 1: Функция с параметрами и возвращаемым значением
x = 1.3802
y = -1.9
Результат: d = 6.060799
```

Рисунок 3.2 - изменение порядка параметров.

Эксперимент 3: Использование выражений в параметрах

Использование выражений в параметрах на рисунке 3.3

```
Проект 3: Функция с параметрами и без возвращаемого значения
x = 1.3802
y = -1.9
Результат: d = -7.820895
```

Рисунок 3.3 - использование выражений в параметрах.