

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»

Лабораторные работы
по дисциплине «Алгоритмы и алгоритмические языки»
Вариант №21

Выполнил: студент группы БФИ №2202

Сидорук Д. В.

Принял: старший преподаватель Загвоздкина А. В.

Москва, 2023 г.

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»

ОТЧЕТ

по дисциплине «Алгоритмы и алгоритмические языки»

Лабораторная работа № 1

Программирование арифметических выражений на языке Visual C# с
использованием методов.

Выполнил: студент группы БФИ №2202

Сидорук Д. В.

Принял: старший преподаватель Загвоздкина А. В.

Москва, 2023 г.

РАЗРАБОТКА ПРОГРАММЫ ДЛЯ ВЫЧИСЛЕНИЯ АРИФМЕТИЧЕСКОГО ВЫРАЖЕНИЯ

Разработать программу для расчета арифметического выражения

$$z = \left(1 + \frac{1}{x^2}\right)^x - 12x^2y$$

Для того, чтобы выполнить задание, необходимо разработать следующие методы, которые должны быть размещены в внешней dll-библиотеке:

1. Метод `public static double GetDouble(TextBox t)`, предназначенный для получения текста из текстового поля ввода и преобразования его в значение с плавающей точкой.
2. Метод `public static double OutputDouble(TextBox t, double value)`, предназначенный для конвертации переданного значения в текст и установки его в качестве текста переданного текстового поля.
3. Метод `public static double Evaluate(double x, double y)`, предназначенный для вычисления значения арифметического вычисления с возвратом результата по значению.
4. Метод `public static double Evaluate(double x, double y, out double z)`, предназначенный для вычисления значения арифметического вычисления с возвратом результата через аргумент с `out` модификатором.
5. Метод `public static double ReferenceEvaluate(double x, double y, ref double z)`, предназначенный для вычисления значения арифметического вычисления с возвратом результата через аргумент с `ref` модификатором.

Перечень блок-схем

На рисунке ниже приведена блок-схема алгоритма метода `double GetDouble(TextBox t)`, предназначенного для получения текста из текстового поля ввода и преобразования его в значение с плавающей точкой.

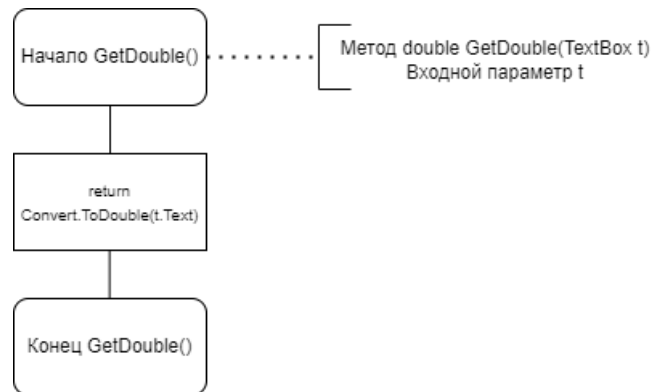


Рисунок 1 — блок-схема алгоритма метода `double GetDouble(TextBox t)`, предназначенного для получения текста из текстового поля ввода и преобразования его в значение с плавающей точкой.

На рисунке ниже приведена блок-схема метода `void OutputDouble(TextBox t, double value)`, предназначенного для конвертации переданного значения в текст и установки его в качестве текста переданного текстового поля.

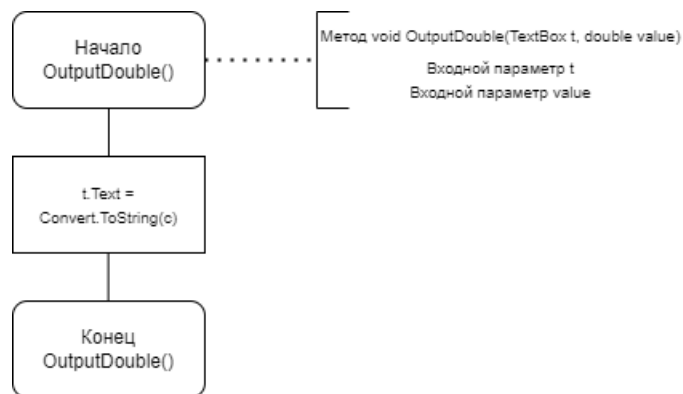


Рисунок 2 — блок-схема алгоритма метода `double GetDouble(TextBox t)`, предназначенного для конвертации переданного значения в текст и установки его в качестве текста переданного текстового поля.

На рисунках ниже представлены:

1. Рисунок 3 — блок-схема алгоритма метода, вычисляющего арифметическое выражение и возвращающего значение с помощью оператора `return`
2. Рисунок 4 — блок-схема алгоритма метода, вычисляющего арифметическое выражение и возвращающего значение с помощью модификатора `out`
3. Рисунок 5 — блок-схема алгоритма метода, возвращающего значение с помощью модификатора `ref`

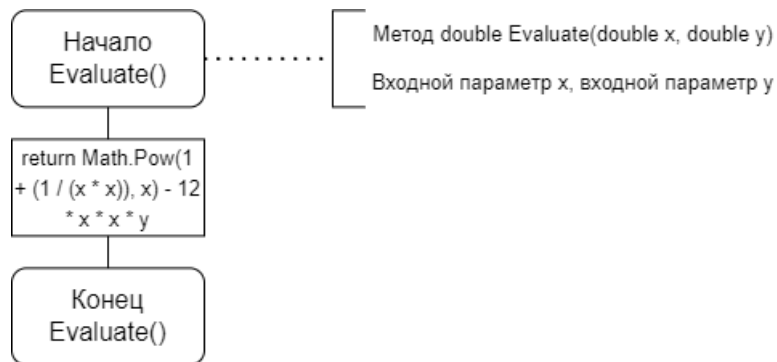


Рисунок 3 — блок-схема алгоритма метода `double Evaluate(double x, double y)`, предназначенного для вычисления значения арифметического вычисления с возвратом результата по значению.

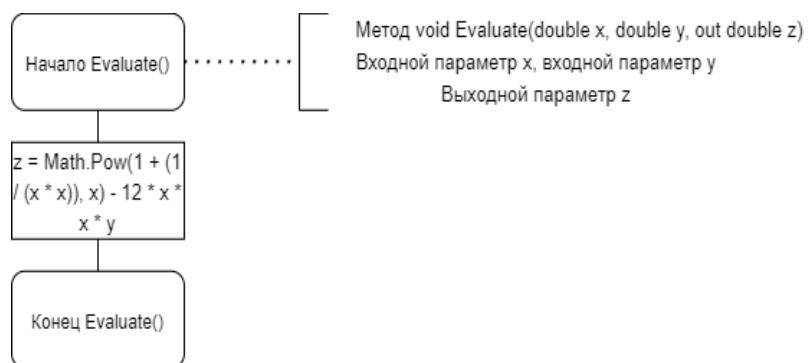


Рисунок 4 — блок-схема алгоритма метода `void Evaluate(double x, double y, out double z)`, предназначенного для вычисления значения арифметического вычисления с возвратом результата через аргумент с `out` модификатором.

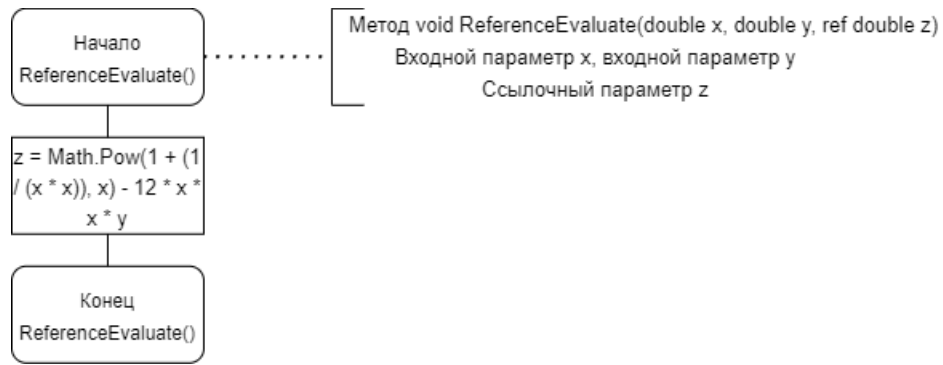


Рисунок 5 — блок-схема алгоритма метода `double ReferenceEvaluate(double x, double y, ref double z)`, предназначенного для вычисления значения арифметического вычисления с возвратом результата через аргумент с `ref` модификатором.

На рисунке ниже представлена блок-схема алгоритма событийной кнопки.

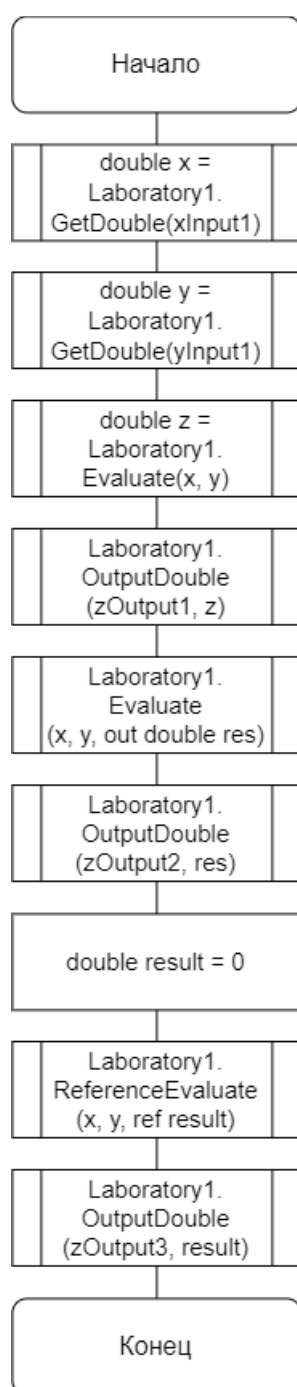


Рисунок 6 — блок-схема алгоритма событийной кнопки.

Содержание DLL-библиотеки

```
using System;
using System.Windows.Forms;

namespace Laboratory1Library
{
    public class Laboratory1
    {
        public static double GetDouble(TextBox t) {
            return Convert.ToDouble(t.Text);
        }
        public static void OutputDouble(TextBox t, double
value)
        {
            t.Text = Convert.ToString(value);
        }
        public static double Evaluate(double x, double y)
        {
            return Math.Pow(1 + (1 / (x * x)), x) - 12 *
x * x * y;
        }
        public static void Evaluate(double x, double y,
out double z)
        {
            z = Math.Pow(1 + (1 / (x * x)), x) - 12 * x *
x * y;
        }
        public static void ReferenceEvaluate(double x,
double y, ref double z)
        {
            z = Math.Pow(1 + (1 / (x * x)), x) - 12 * x *
x * y;
        }
    }
}
```

Содержание основной части программы

```
using System;
using System.Windows.Forms;

using Laboratory1Library;

namespace LaboratoryWorks
{
    public partial class Laboratory1Form : Form
    {
        public Laboratory1Form()
        {
            InitializeComponent();
        }

        private void evaluateButton1_Click(object sender,
EventArgs e)
        {
            double x = Laboratory1.GetDouble(xInput1);
            double y = Laboratory1.GetDouble(yInput1);

            double z = Laboratory1.Evaluate(x, y);
            Laboratory1.OutputDouble(zOutput1, z);

            Laboratory1.Evaluate(x, y, out double res);
            Laboratory1.OutputDouble(zOutput2, res);

            double result = 0;
            Laboratory1.ReferenceEvaluate(x, y, ref
result);
            Laboratory1.OutputDouble(zOutput3, result);
        }
    }
}
```

Результаты выполнения программы

На рисунке ниже приведен результат выполнения программы при входных данных $x = 1,005$; $y = 3,01$

Лабораторная работа №1 - условие

Лабораторная работа №1
Программирование арифметических выражений на языке Visual C# с использованием методов

$$z = \left(1 + \frac{1}{x^2}\right)^x - 12x^2y$$

x = y =

z (return) = z (out) = z (ref) =

Рисунок 7 — результат выполнения программы при входных данных $x = 1,005$; $y = 3,01$

Список используемых источников

1. Гуриков С. Р. Введение в программирование на языке Visual C#: учебное пособие / С. Р. Гуриков. — Москва : ФОРУМ : ИНФРА-М, 2020. — 447 с.