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 заданию №1**

**по дисциплине**

**«Учебная (ознакомительная) практика»**

Выполнил: студент гр. БЭИ2202

Кулешов А. С.

Вариант 15.

Проверил: доц. Гуриков С. Р.

Москва, 2023 г.

**1) Индивидуальное задание на разработку проекта**

Разработать программный код функции:

Осуществить вычисление тремя разными способами (при помощи return, out и ref).

**2) Техническое задание**

1. Выбрать вариант задания из таблицы согласно порядковому номеру в журнале группы.

2. Следует учесть, что исходные данные в таблице вариантов даны исключительно для тестирования программы. В программе предусмотреть ввод любых данных от пользователя с последующей защитой от ошибок.

3. Записать заданные или полученные в процессе формализации математические формулы на языке программирования.

4. Разработать алгоритмы решения заданной задачи (количество алгоритмов равно количеству используемых пользовательских методов алгоритм событийной).

5. Создать проект с DLL-библиотекой, решающий предписанные задачи, разработать метод ввода исходных данных и метод вывода результата. Обмен данными между подпрограммами должен осуществляться через параметры, без использования глобальных переменных. Событийная процедура должна содержать только операторы вызова пользовательских подпрограмм.

6. В ходе решения задачи своего варианта предусмотреть расчет арифметического выражения тремя способами: с модификаторами ref и out, оператором return.

7. Организовать использование проверки корректности ввода данных (параграф 2.6 основного источника литературы) относительно объектов Текстовое поле на всех формах проекта.

8. Предусмотреть возможность вывода сообщения об ошибке в виде всплывающей подсказки, используя элемент управления ErrorProvider (стр. 279 основного источника литературы) относительно объектов Текстовое поле на всех формах проекта.

9. Результаты работы программы записать в текстовый файл.

10. Подготовить тесты для контрольного решения задачи. Показать их использование, отобразив в отчете по заданию.

11. Выполнить проект и получить результаты, которые отобразить в виде рисунков при написании отчета по заданию.

12. Выполнить отчет в соответствии с ГОСТ на схемы алгоритмов и согласно ГОСТ оформления отчетов по НИР.

**3) Схемы алгоритмов**

Алгоритмы ввода/вывода представлены на рисунках 1, 2. Ввод реализуется благодаря конвертации строки в вещественное число (метод Convert.ToDouble()), также используется проверка корректности данных для преобразования и выдаётся ошибка в Error Provider. Вывод реализуется благодаря конвертации числа в строку (метод ToString()). Для округления вывода до пятого знака после запятой дополнительно была использована маска “0.00000”.

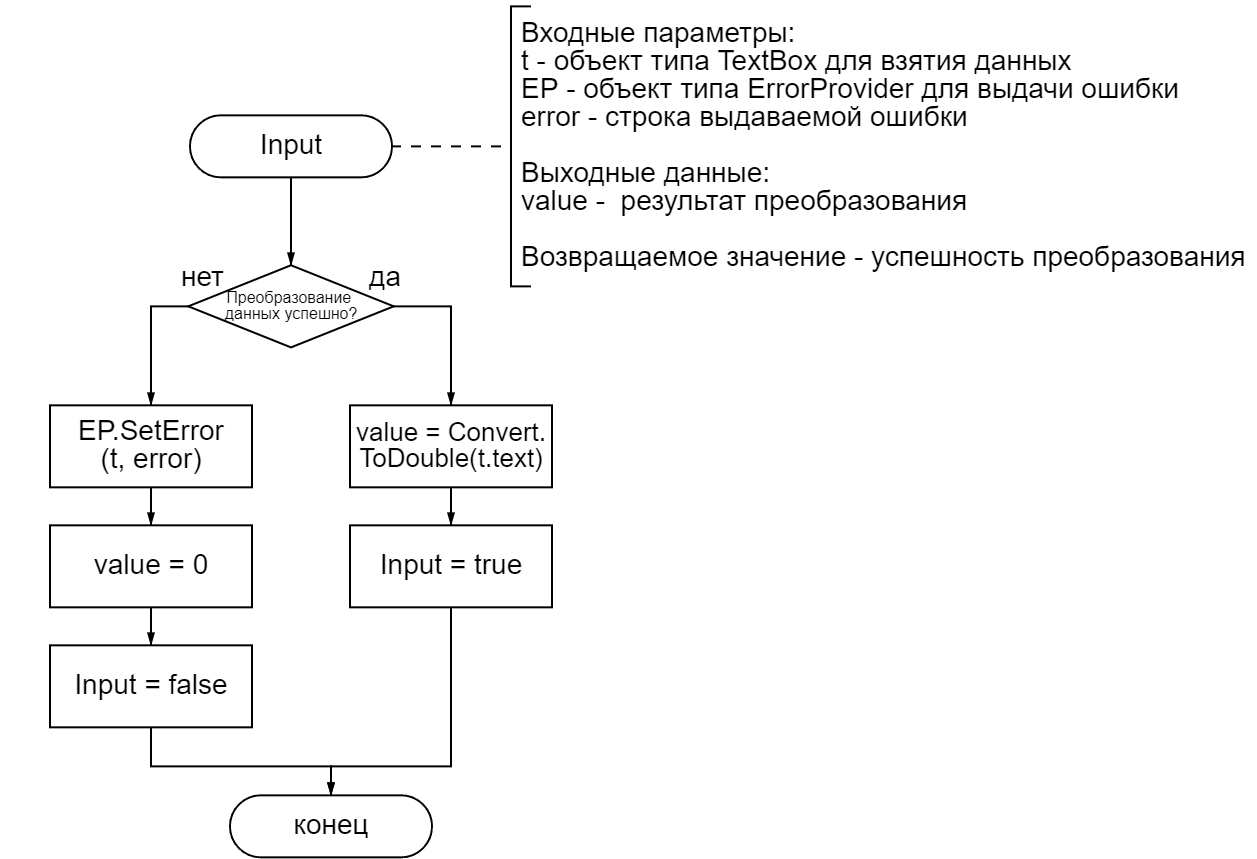


Рисунок 1 – Метод ввода Input

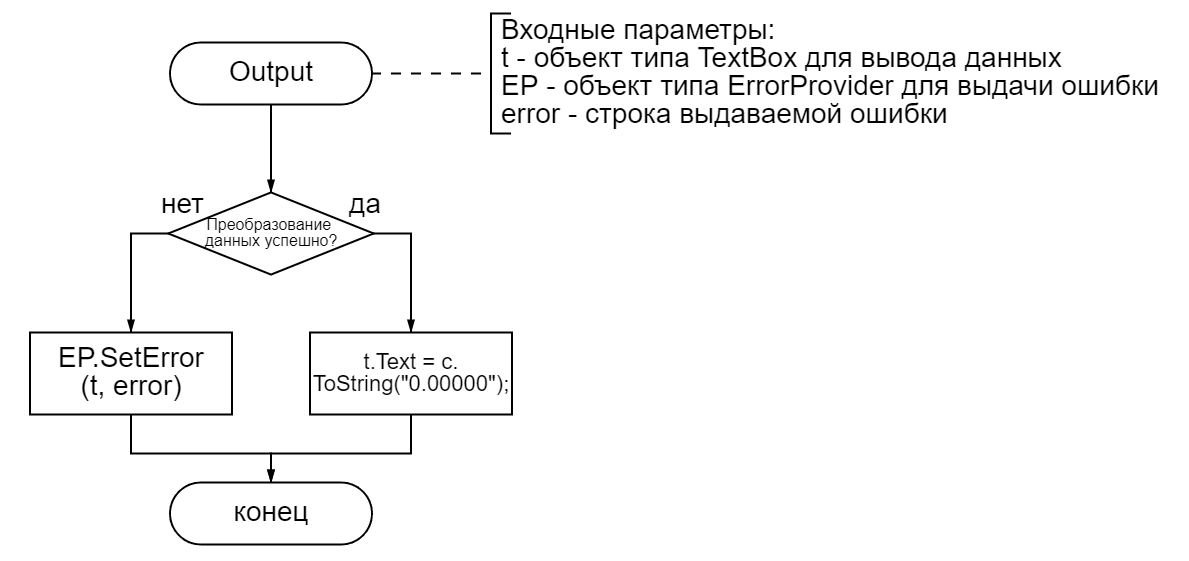


Рисунок 2 – Метод вывода Output

Алгоритм метода, решающего поставленную задачу, представлен на рисунке 3. Математические функции были взяты из математического пакета System.Math. Для файлового вывода использовалась функция System.IO.File.WriteAllText

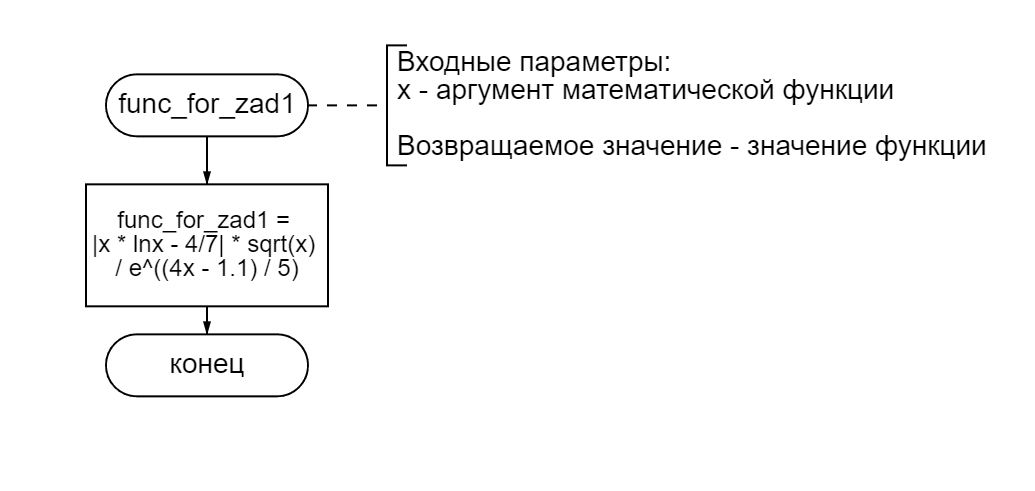


Рисунок 3 – Метод решения задачи func\_for\_zad1

**4) Программный код DLL-библиотеки**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace dll\_for\_prac

{

public class Class1

{

// Ввод данных с обработкой ошибки (входные данные могут быть не преобразованны при конвертации). Возвращает корректность процесса конвертации

public static bool Input(TextBox t, out double value, ErrorProvider EP, String error)

{

try

{

value = Convert.ToDouble(t.Text);

return true;

}

catch (Exception Ex)

{

EP.SetError(t, error);

value = 0;

return false;

}

}

// Вывод данных с обработкой ошибки (double переменная может иметь тип NaN)

public static void Output(TextBox t, double c, ErrorProvider EP, String error)

{

if (Double.IsNaN(c))

{

EP.SetError(t, error);

}

else

{

t.Text = c.ToString("0.00000");

}

}

// Функция для решения первого задания (через return)

public static double func\_for\_zad1(double x)

{

return Math.Abs(x \* Math.Log(x) - 4d / 7d) \* Math.Sqrt(x) / Math.Exp((4 \* x - 1.1d) / 5);

}

// Функция для решения первого задания (через out)

public static void func\_for\_zad1\_out(double x, out double res)

{

res = Math.Abs(x \* Math.Log(x) - 4d / 7d) \* Math.Sqrt(x) / Math.Exp((4 \* x - 1.1d) / 5);

}

// Функция для решения первого задания (через ref)

public static void func\_for\_zad1\_ref(double x, ref double res)

{

res = Math.Abs(x \* Math.Log(x) - 4d / 7d) \* Math.Sqrt(x) / Math.Exp((4 \* x - 1.1d) / 5);

}

}

}

**5) Программный код событийной процедуры**

private void Calculate\_Click(object sender, EventArgs e)

{

double x1, x2, x3;

double ans1, ans2, ans3 = 0;

String final\_content = "Задание №1\n\n";

if (Input(textBox1, out x1, errorProvider1, "В поля для ввода данных допускается лишь ввод числовых значений с плавающей точкой"))

{

try

{

ans1 = func\_for\_zad1(x1);

final\_content += "Входные данные(return): " + x1 + '\n';

final\_content += "Выходные данные(return): " + ans1 + "\n\n";

Output(textBox4, ans1, errorProvider1, "Результатом вычисления является не действительное число");

}

catch (Exception Ex)

{

errorProvider1.SetError(textBox4, "Во время вычислений произошла ошибка");

}

}

if (Input(textBox2, out x2, errorProvider1, "В поля для ввода данных допускается лишь ввод числовых значений с плавающей точкой"))

{

try

{

func\_for\_zad1\_out(x2, out ans2);

final\_content += "Входные данные(out): " + x2 + '\n';

final\_content += "Выходные данные(out): " + ans2 + "\n\n";

Output(textBox5, ans2, errorProvider1, "Результатом вычисления является не действительное число");

}

catch (Exception Ex)

{

errorProvider1.SetError(textBox5, "Во время вычислений произошла ошибка");

}

}

if (Input(textBox3, out x3, errorProvider1, "В поля для ввода данных допускается лишь ввод числовых значений с плавающей точкой"))

{

try

{

func\_for\_zad1\_ref(x3, ref ans3);

final\_content += "Входные данные(ref): " + x3 + '\n';

final\_content += "Выходные данные(ref): " + ans3 + "\n\n";

Output(textBox6, ans3, errorProvider1, "Результатом вычисления является не действительное число");

}

catch (Exception Ex)

{

errorProvider1.SetError(textBox6, "Во время вычислений произошла ошибка");

}

}

System.IO.File.WriteAllText("task1.txt", final\_content);

}

**6) Результаты работы программы**

На рисунке 5 представлен результат работы программы при данном в задании для проверки значении входного параметра и значении функции. Как видно, программа работает корректно.

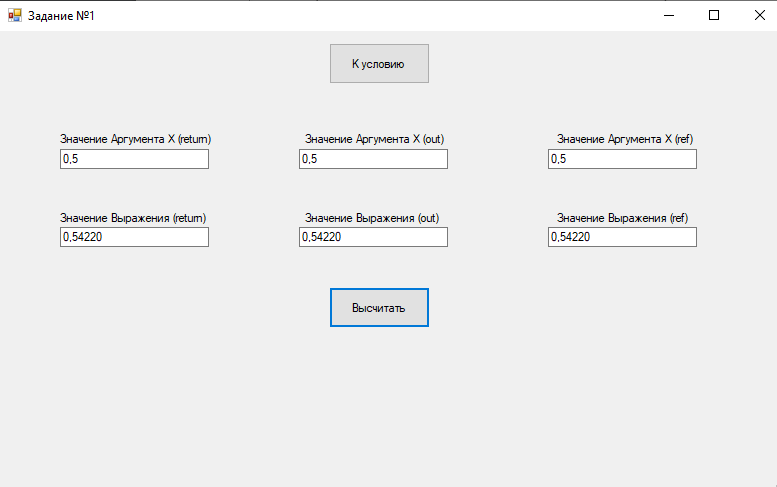


Рисунок 5 – Результат работы программы при заданном параметре

Попробуем ввести недопустимые данные, например слово или число не из области допустимых значений. Рисунки 6,7.

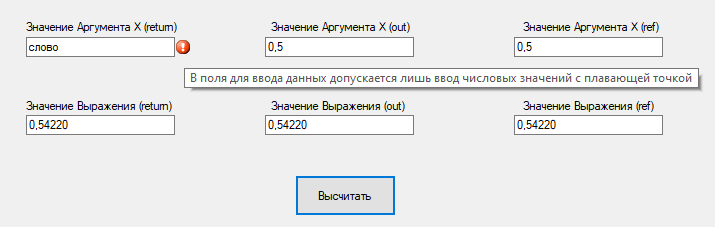


Рисунок 6 – Результат при вводе слова

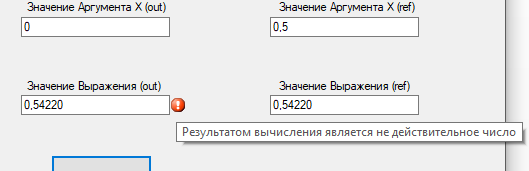


Рисунок 7 – Результат при вводе некорректного числа

**7) Список основных источников литературы**

1. Гуриков С. Р. Введение в программирование на языке Visual C# : учебное пособие / С.Р. Гуриков. — Москва : ФОРУМ : ИНФРА-М, 2020. — 447 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-00091-458-8. - Текст : электронный. - URL: https://znanium.com/catalog/product/1092167 (дата обращения: 23.06.2023). – Режим доступа: по подписке + библиотека МТУСИ