**4 ОБРАБОТКА ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ**

Задание 1. Вычислить значение выражений. В каждой разработанной программе должна быть осуществлена обработка исключительных ситуаций. Отдельно обработаны исключения DivideByZeroException, FormatException.

Листинг программы:

using System;

namespace task1

{

class Program

{

static void Main(string[] args)

{

try

{

Console.Write("Введите x: ");

int x = Convert.ToInt32(Console.ReadLine());

double y = Math.Atan(x) / (x - 3);

if (Double.IsInfinity(y))

{

Console.WriteLine("Бесконечность");

}

else

{

Console.WriteLine(y);

}

double y1 = Math.Log10(x) + (5 \* x - 3) / (x - 1);

if (Double.IsInfinity(y1))

{

Console.WriteLine("Бесконечность");

}

else

{

Console.WriteLine(y1);

}

}

catch (FormatException)

{

Console.WriteLine("Нужно ввести число");

}

catch (DivideByZeroException)

{

Console.WriteLine("Деление на ноль недопустимо!!!");

}

}

}

}

Таблица 4.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 5 | 0,6867, 5,699 |

Анализ результатов:



Рисунок 4.1 – Результат работы программы

Источник – собственная разработка

Задание 2. Для данного вещественного x найти значение следующей функции f, принимающей вещественные значения. В каждой разработанной программе должна быть осуществлена обработка исключительных ситуаций. Отдельно обработаны исключения DivideByZeroException, FormatException. Сгенерируйте пользовательское исключение с помощью оператора throw при проверке условий выхода за диапазон.

Листинг программы:

using System;

namespace task2

{

class Program

{

static void Main(string[] args)

{

try

{

Console.Write("Введите x: ");

double x = Convert.ToDouble(Console.ReadLine());

double f;

if ((0 < x) && (x < 1))

{

f = x \* Math.Cos(x);

}

else if (1 <= x)

{

f = 12 / (3 \* x - 3);

}

else

{

throw new Exception("выход из диапазона допустимых значений x");

}

if (Double.IsInfinity(f))

{

Console.WriteLine("Бесконечность");

}

else

{

Console.WriteLine(f);

}

}

catch (FormatException)

{

Console.WriteLine("Нужно ввести число");

}

catch (DivideByZeroException)

{

Console.WriteLine("Делить на 0 нельзя!!!");

}

catch (Exception e)

{

Console.WriteLine($"Ошибка: {e.Message}");

}

}

}

}

Таблица 4.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 6 | 0,8 |

Анализ результатов:



Рисунок 4.2 – Результат работы программы

Источник – собственная разработка