**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:

while (true)

{

try

{

Console.Write("Enter x: ");

double x = double.Parse(Console.ReadLine());

Console.WriteLine($"Formula 1 = {Formula(x)}");

Console.WriteLine($"Formula 2 = {Formula2(x)}");

break;

}

catch (FormatException e)

{

Console.WriteLine($"Sorry, it's format exception (Error: {e})");

}

catch (DivideByZeroException e)

{

Console.WriteLine($"Divide by zero (x / 0) (Error: {e})");

}

catch (Exception e)

{

Console.WriteLine($"Some error (Error: {e})");

}

}

static double Formula(double x)

{

try

{

return x + ((2 \* x) / (4 \* Math.Sin(x)));

}

catch (Exception)

{

throw new DivideByZeroException();

}

}

static double Formula2(double x)

{

try

{

return x - 3 +(1/ Math.Tan(x - 1));

}

catch (Exception)

{

throw new DivideByZeroException();

}

}

Таблица 4.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 78 | 153.8786, 74.9690 |

Источник: собственная разработка

Анализ результатов:

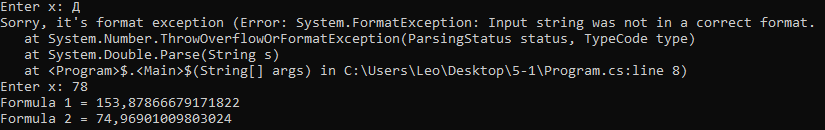


Рисунок 4.1 – Результат работы программы

Источник: собственная разработка