**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:

namespace task1

{

using System;

class Program

{

static double CalculateExpressionA(double x)

{

try

{

double numerator = Math.Pow(x, 2) - 3;

double denominator = Math.Pow(x, 3) - 4;

if (denominator == 0)

{

throw new DivideByZeroException("Деление на ноль!.");

}

return numerator / denominator;

}

catch (DivideByZeroException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

catch (FormatException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

}

static double CalculateExpressionB(double x)

{

try

{

if (x <= 0)

{

throw new FormatException("Натуральный логорифм не может быть меньше или равен нулю.");

}

return Math.Log(x) / (x - 1);

}

catch (DivideByZeroException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

catch (FormatException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

}

static void Main()

{

Console.WriteLine("Введите X: ");

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

double resultA = CalculateExpressionA(x);

Console.WriteLine($"Результат выражения А для Х = {resultA}");

double resultB = CalculateExpressionB(x);

Console.WriteLine($"Результат выражения B для Х = {resultB}");

}

}

}

Таблица 4.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| X: 2 | A: 0,25;  B: 0,6931471805599453; |

Анализ результатов:

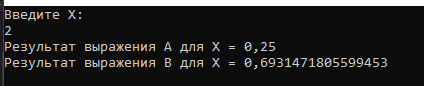


Рисунок 4.1 – Результат работы программы

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:

namespace task2

{

using System;

class Program

{

static void SortInc3(ref double A, ref double B, ref double C)

{

if (A > B)

{

double temp = A;

A = B;

B = temp;

}

if (B > C)

{

double temp = B;

B = C;

C = temp;

if (A > B)

{

temp = A;

A = B;

B = temp;

}

}

}

static void Main()

{

Console.WriteLine("Введите значение А1: ");

double A1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите значение B1: ");

double B1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите значение C1: ");

double C1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите значение A2: ");

double A2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите значение B2: ");

double B2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите значение C2: ");

double C2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Начальные значения:");

Console.WriteLine($"Набор 1: A={A1}, B={B1}, C={C1}");

Console.WriteLine($"Набор 2: A={A2}, B={B2}, C={C2}");

SortInc3(ref A1, ref B1, ref C1);

SortInc3(ref A2, ref B2, ref C2);

Console.WriteLine("\nСортированные значения:");

Console.WriteLine($"Набор 1: A={A1}, B={B1}, C={C1}");

Console.WriteLine($"Набор 2: A={A2}, B={B2}, C={C2}");

}

}

}

Таблица 4.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| A1 = 1.0;  B1 = 2.0;  C1 = 5.0;  A1 = 1.0;  B1 = 2.0;  C1 = 5.0; | Сортированные значения:  A1 = -2; B1 = 1; C1 = 2;  A2 = -3; B2 = 1; C2 = 2; |

Анализ результатов:



Рисунок 4.2 – Результат работы программы

Задание 3. В каждой разработанной программе должна быть осуществлена

обработка исключительных ситуаций. Отдельно обработаны исключения

DivideByZeroException, FormatException. По возможности сгенерируйте

пользовательское исключение с помощью оператора. Описать метод SortInc3(A, B, C), меняющий содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по возрастанию (A, B, C — вещественные параметры,являющиеся одновременно входными и выходными). С помощью этогометода упорядочить по возрастанию два данных набора из трех чисел: (A1,B1, C1) и (A2, B2, C2).

Листинг программы:

namespace task3

{

using System;

class Program

{

static double CalculateFunction(double x)

{

try

{

if (x > -1 && x <= 5)

{

if (x == -1)

{

throw new DivideByZeroException("Деление на ноль!.");

}

return x / (5 \* x + 5);

}

else if (x > -5 && x < -1)

{

return Math.Pow(x, 3);

}

else

{

throw new ArgumentOutOfRangeException("Значение Х вышло за пределы.");

}

}

catch (DivideByZeroException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

catch (FormatException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

catch (ArgumentOutOfRangeException ex)

{

Console.WriteLine(ex.Message);

return double.NaN;

}

}

static void Main()

{

Console.WriteLine("Введите значение Х: ");

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

double result = CalculateFunction(x);

Console.WriteLine($"Значения функции для x = {x}: {result}");

}

}

}

Таблица 4.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| X = -3 | Значение функции = -27 |

Анализ результатов:



Рисунок 4.3 – Результат работы программы