**4 ОБРАБОТКА ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ**

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

Листининг 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.

Листининг 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

Задание 3. Описать метод SortDec3(A, B, C), меняющий содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этого метода упорядочить по убыванию два данных набора из трех чисел: (A1, B1, C1) и (A2, B2, C2).

Листининг программы:

using System;

namespace task3

{

class Program

{

static void SortInc3(ref int a,ref int b,ref int c)

{

int[] numbers = { a, b, c };

Array.Sort(numbers);

a = numbers[0];

b = numbers[1];

c = numbers[2];

}

static void Main(string[] args)

{

Console.Write("Введите а1: ");

int a1 = Convert.ToInt32(Console.ReadLine());

Console.Write("Введите b1: ");

int b1 = Convert.ToInt32(Console.ReadLine());

Console.Write("Введите c1: ");

int c1 = Convert.ToInt32(Console.ReadLine());

SortInc3(ref a1, ref b1, ref c1);

Console.WriteLine($"a1: {a1}\nb1: {b1}\nc1: {c1}");

Console.Write("Введите а2: ");

int a2 = Convert.ToInt32(Console.ReadLine());

Console.Write("Введите b2: ");

int b2 = Convert.ToInt32(Console.ReadLine());

Console.Write("Введите c2: ");

int c2 = Convert.ToInt32(Console.ReadLine());

SortInc3(ref a2, ref b2, ref c2);

Console.WriteLine($"a2: {a2}\nb2: {b2}\nc2: {c2}");

}

}

}

Таблица 4.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 14, 15, 7 | 7, 14, 15 |
| 15, 13,1 | 1, 13, 15 |

Анализ результатов:

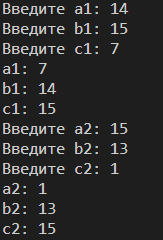


Рисунок 4.3 – Результат работы программы

Источник – собственная разработка