ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

1. Задание

Для заданного натурального *k* и действительного *x* подсчитать следующее выражение:



1. Планируемая реализация

В методе Main() внутри блока try вводятся значения k и x с помощью метода InputValue() и проводятся проверки x=0 и k<0. Если возникает исключение, то с помощью блока catch отлавливается супертип Exepction и выводится стандартное сообщение об ошибке.

1. Итоговая реализация

В классе Program, в блоке try вещественной переменной x и целочисленной переменной k присваиваются значения, вызывая методы класса Input InputX() и InputK(), о которых будет сказано далее. Затем производятся проверки, что x=0 и k<0. Если x равен 0, то вызывается исключение DivideByZeroException() и происходит переход в блок catch, если k меньше 0, то вызывается пользовательское исключение KIsNotNaturalException(), о котором будет сказано ниже, и происходит переход в блок catch. В конце работы программы вызывается метод SumResult(), выводящий на экран результат расчёта суммы.

Фрагмент программного кода

static void Main()

{

try{

double x = Input.InputX();

int k = Input.InputK();

if (x == 0){

throw new DivideByZeroException("На ноль делить нельзя.");

}

if (k < 0){

throw new KIsNotNaturalException("k < 0. k должно быть натуральным.");

}

Console.WriteLine(SumResult(x, k));

}

catch (Exception e){

Console.WriteLine(e.Message);

}

Console.WriteLine("Нажмите любую клавишу для завершения программы");

Console.ReadKey();

}

private static string SumResult(double x, int k)

{

double Sum = 0;

try{

for (int i = 1; i <= k; i++){

Sum += (Math.Pow(3.0, i) + Math.Pow(4.0, i)) / Math.Pow(x, 3.0);

}

}

catch (Exception e){

Console.WriteLine(e.Message);

}

return Sum.ToString();

}

При разработке программы был создан класс Input, внутри класса есть два метода InputX() и InputK(). В них считываются с консоли переменные, в первом методе вещественная, во втором целочисленная, для каждой переменной производится проверка формата ввода методом TryParse(). Если формат не соответствует необходимому, то вызывается исключение FormatException(). Методы практически дублируют себя, вместо того чтобы использовать один метод, потому что в методе InputK() производится проверка на переполнение с помощью оператора checked(), применимый только для переменных типа int. В тексте будет приведён только метод InputK(), другой метод отличается от него, отсутствием проверки с помощью checked() и вещественным типом возвращаемого значения.

Фрагмент программного кода

public static int InputK()

{

Console.Write("Введите k:");

string strValue = Console.ReadLine();

int Value;

if (!int.TryParse(strValue, out Value)){

Value = checked(int.Parse(strValue));

throw new FormatException("Нельзя использовать букву вместо цифры.");

}

else{

return int.Parse(strValue);

}

}

Для обработки ситуации k<0 было создано пользовательское исключение KIsNotNaturalException, фактически реализуется с помощью класса, наследуемого от базового класса Exeption.

Фрагмент программного кода

public class KIsNotNaturalException : Exception

{

public KIsNotNaturalException(string message) : base(message) { }

}