ФГБОУ ВПО

ДВГУПС

Кафедра: ИТИС

Курсовая работа

Выполнил: Точилкин М.А.

Группа 220

Проверил: Володажский А.А.

Хабаровск, 2015

Цель: Использовать полученные знания для реализации программ на языке C#.

Задание 1. Пользователь вводит количество точек и создается массив со случайными координатами. Диапазон x и y заранее известен. На сгенерированном множестве точек найти три треугольника: 1) Содержащие максимальное количество точек внутри. 2) Имеющие максимальную площадь. Координаты вывести в файл.

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Text;

namespace lulz

{

class Program

{

static void Main(string[] args)

{

Console.Write("Введите количество точек = ");

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

//Формула центрированного треугольного числа

int final\_count\_array = (3 \* (n \* n) + 3 \* n + 2) / 2;

if (n >= 3)

{

int[,] point = new int[n, 2];

int point\_h = point.GetLength(0);

int point\_w = point.GetLength(1);

//Центрированное треугольное число равное количеству возможных комбинаций

int[,] ploshad = new int[final\_count\_array, 7];

//Собственный лист созданный в классе

List<FinalPointer> all\_pointers = new List<FinalPointer>();

Random rnd = new Random();

//Создаем массив точек где i,z соответсвенно равны x,y

for (int i = 0; i < point\_h; i++)

{

for (int z = 0; z < point\_w; z++)

{

point[i, z] = rnd.Next(-5, 25);

Console.Write(point[i, z] + " ");

}

Console.WriteLine();

}

int kokoko = 0;

int zab = 0;

//Перебор вариантов площади и вариантов вхождения

for (int i = 0; i < point\_h; i++)

{

for (int g = 0; g < point\_h; g++)

{

for (int z = 0; z < point\_h; z++)

{

int x\_a = point[i, 0];

int x\_b = point[g, 0];

int x\_c = point[z, 0];

int y\_a = point[i, 1];

int y\_b = point[g, 1];

int y\_c = point[z, 1];

//Проверка на вхождение

for (int r = 0; r < point\_h; r++)

{

int x = point[r, 0];

int y = point[r, 1];

//Проверка принадлежности точки треугольнику формулами Герона

int summa\_s = 1 / 2 \* Math.Abs((x\_b - x\_a) \* (y\_c - y\_a) - (x\_c - x\_a) \* (y\_b - y\_a));

int s1 = 1 / 2 \* Math.Abs((x\_b - x\_a) \* (y - y\_a) - (x - x\_a) \* (y\_b - y\_a));

int s2 = 1 / 2 \* Math.Abs((x - x\_a) \* (y\_c - y\_a) - (x\_c - x\_a) \* (y - y\_a));

int s3 = 1 / 2 \* Math.Abs((x\_b - x) \* (y\_c - y) - (x\_c - x) \* (y\_b - y));

if (summa\_s == s1 + s2 + s3)

{

zab += 1;

};

}

//Площадь треугольника по координатам

int s = (Math.Abs((x\_b - x\_a) \* (y\_c - y\_a) - (x\_c - x\_a) \* (y\_b - y\_a))) / 2;

//Удаляет исключения т.к они равны 0

if (s != 0)

{

all\_pointers.Add(new FinalPointer()

{

Summa = s,

Position\_x\_a = x\_a,

Position\_x\_b = x\_b,

Position\_x\_c = x\_c,

Position\_y\_a = y\_a,

Position\_y\_b = y\_b,

Position\_y\_c = y\_c,

Counter\_point = zab

});

};

kokoko++;

zab = 0;

}

}

}

//Запись в файл

string fileName = "final.txt";

StreamWriter writer = new StreamWriter(fileName);

//Сортируем массив по площади

List<FinalPointer> SortedList\_Counter\_point = all\_pointers.OrderBy(FinalPointer => FinalPointer.Counter\_point).ToList();

List<FinalPointer> SortedList\_Pointers = all\_pointers.OrderBy(FinalPointer => FinalPointer.Summa).ToList();

int SortedList\_Pointers\_L = SortedList\_Pointers.Count;

//Запись в файл максимальную площадь и координаты

for (int i = SortedList\_Pointers\_L - 3; i < SortedList\_Pointers\_L; i++)

{

writer.WriteLine(Convert.ToString(SortedList\_Pointers[i].Summa + " - [" + SortedList\_Pointers[i].Position\_x\_a + "," + SortedList\_Pointers[i].Position\_y\_a + "] [" + SortedList\_Pointers[i].Position\_x\_b + "," + SortedList\_Pointers[i].Position\_y\_b + "] [" + SortedList\_Pointers[i].Position\_x\_c + "," + SortedList\_Pointers[i].Position\_y\_c + "]"));

}

writer.WriteLine("--------------------------------------------------");

for (int i = SortedList\_Pointers\_L - 3; i < SortedList\_Pointers\_L; i++)

{

writer.WriteLine(Convert.ToString(SortedList\_Counter\_point[i].Summa + " - [" + SortedList\_Counter\_point[i].Position\_x\_a + "," + SortedList\_Counter\_point[i].Position\_y\_a + "] [" + SortedList\_Counter\_point[i].Position\_x\_b + "," + SortedList\_Counter\_point[i].Position\_y\_b + "] [" + SortedList\_Counter\_point[i].Position\_x\_c + "," + SortedList\_Counter\_point[i].Position\_y\_c + "] " + SortedList\_Counter\_point[i].Counter\_point));

}

writer.Close();

Console.WriteLine("Готово!");

}

else

{

Console.WriteLine("Количество точек не может быть меньше 3");

};

Console.ReadKey();

}

}

//Кастомный лист с 8-ми столбцами

public class FinalPointer

{

//Сумма чисел

public int Summa { get; set; }

//Точка x1

public int Position\_x\_a { get; set; }

//Точка x2

public int Position\_x\_b { get; set; }

//Точка x3

public int Position\_x\_c { get; set; }

//Точка y1

public int Position\_y\_a { get; set; }

//Точка y2

public int Position\_y\_b { get; set; }

//Точка y3

public int Position\_y\_c { get; set; }

//Количество точек входящих в треугольник

public int Counter\_point { get; set; }

}

}

Задание 2. Реализовать арифметические операции с бесконечно большими числами. Числа выходящие за long и больше 25-ти символов.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Numerics;

using System.Threading.Tasks;

namespace ConsoleApplication3

{

class Program

{

static void Main(string[] args)

{

Console.Write("Введите первое число = ");

BigInteger num1 = BigInteger.Parse(Console.ReadLine());

Console.Write("Введите второе число = ");

BigInteger num2 = BigInteger.Parse(Console.ReadLine());

BigInteger mumnoj = num1 \* num2;

BigInteger sloj = num1 + num2;

BigInteger vihit = num1 - num2;

BigInteger delenie = num1 / num2;

Console.WriteLine("Результат умножения = " + mumnoj);

Console.WriteLine("Результат сложения = " + sloj);

Console.WriteLine("Результат вычитания = " + vihit);

Console.WriteLine("Результат деления = " + delenie);

Console.ReadKey();

}

}

}