1

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication1

{

/\* Заданы три вектора X, Y, Z.

\* Написать программу вычисления S(от 1 до n) = C \* D

\* соответственно равны С = X + У; D = X - Z

\*/

class Garic

{

public double X;

public double Y;

public double Z;

public int n;

public Garic(double X, double Y, double Z, int n)

{

this.X = X;

this.Y = Y;

this.Z = Z;

this.n = n;

}

public double C()

{

return X + Y;

}

public double D()

{

return X - Z;

}

public double arifmetic()

{

double S = 0;

for (int i = 1; i < n; i++)

S += C() \* D();

return S;

}

}

class Program

{

static void Main(string[] args)

{

Console.Write("Введите X: ");

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

Console.Write("Введите Y: ");

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

Console.Write("Введите Z: ");

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

Console.Write("Введите n: ");

int n = int.Parse(Console.ReadLine());

Garic g = new Garic(X, Y, Z, n);

Console.WriteLine(g.arifmetic());

Console.ReadKey();

}

}

}

2

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp9\_

{

/\*

\* Разработать класс, обеспечивающий представление матрицы произвольного размера

\* с возможностью изменения числа строк и столбцов,

\* вывода на экран подматрицы любого размера и всей матрицы.

\* Значения элементов матриц – целочисленные. Матрицы заполняются с помощью генератора случайных чисел.

\* Предусмотреть возможность выполнения следующих операций, для чего разработать соответствующие методы:

• поэлементного сложения объектов класса (матриц одинаковой размерности);

• поэлементного вычитания объектов класса (матриц одинаковой размерности);

• определения среднего арифметического значения матрицы;

• замены отрицательных элементов матрицы на ноль.

\*/

class Matrix

{

public int n;

public int m;

public int[,] arr;

public Matrix(int n, int m, int[,] arr)

{

this.n = n;

this.m = m;

this.arr = arr;

}

public int Ini()

{

Random rand = new Random();

for (int i = 0; i < n; i++)

for (int j = 0; j < m; j++)

arr[i, j] = rand.Next(-9, 10);

return (0);

}

public void Vyvod(int bn, int bj, int ei, int ej)

{

Console.WriteLine("Матрица: ");

for (int i = bn; i < ei; i++)

{

for (int j = bj; j < ej; j++)

Console.Write(arr[i, j] + " ");

Console.WriteLine();

}

}

public static Matrix operator +(Matrix v1, Matrix v2)

{

int[,] arr = new int[5, 5];

Matrix temp = new Matrix(5, 5, arr);

for (int i = 0; i < 5; i++)

for (int j = 0; j < 5; j++)

temp.arr[i, j] = v1.arr[i, j] + v2.arr[i, j];

return temp;

}

public static Matrix operator -(Matrix v1, Matrix v2)

{

int[,] arr = new int[5, 5];

Matrix temp = new Matrix(5, 5, arr);

for (int i = 0; i < 5; i++)

for (int j = 0; j < 5; j++)

temp.arr[i, j] = v2.arr[i, j] - v1.arr[i, j];

return temp + v1;

}

public void Sr()

{

double srednee;

int sum = 0, count = 0;

for (int i = 0; i < 5; i++)

for (int j = 0; j < 5; j++)

{

sum += arr[i, j];

count++;

}

srednee = Convert.ToDouble(Convert.ToDouble(sum) / Convert.ToDouble(count));

Console.WriteLine("Среднее: ");

Console.WriteLine(srednee);

}

public int Preobr()

{

for (int i = 0; i < n; i++)

for (int j = 0; j < m; j++)

if (arr[i, j] < 0)

arr[i, j] = 0;

Vyvod(0, 0, 5, 5);

return (0);

}

}

class Program

{

static void Main(string[] args)

{

int n = 5, m = 5;

int[,] arr = new int[n, m];

Matrix M1 = new Matrix(n, m, arr);

Matrix M2 = new Matrix(n, m, arr);

Matrix M3 = new Matrix(n, m, arr);

M1.Ini();

M1.Vyvod(0, 0, n, m);

M1.Vyvod(1, 1, 3, 3);

M2.Ini();

M2.Vyvod(0, 0, n, m);

M1.Sr();

M3 = M1 + M2;

M3.Vyvod(0, 0, n, m);

M3 = M1 - M2;

M3.Vyvod(0, 0, n, m);

M1.Preobr();

}

}

}