Лекция 2.

Введение в классы, перегрузка, индексатор, наследование

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Cons1009

{

class cla

{

protected int[]mas;

public cla(int n)

{

mas = new int[n];

}

public void inpt()

{

for(int i=0;i<mas.Length;i++)

{

Console.Write(i + " ");

mas[i] = Int32.Parse(Console.ReadLine());

}

}

public void fun1(out int kol)

{

kol = 0;

foreach (int x in mas)

if (x < 0) kol++;

}

~cla()

{

Console.WriteLine("I am a destructor");

// Console.ReadLine();

}

}

class array

{

protected double[] mas;

public array(int n)

{

mas = new double[n];

}

public double this[int k]

{

get

{

return mas[k];

}

set

{

if(k>=0&&k<mas.Length)

mas[k] = value;

}

}

public void inpt()

{

for(int i=0;i<mas.Length;i++)

{

Console.Write(i + " ");

mas[i] = Convert.ToDouble(Console.ReadLine());

}

}

public void otpt()

{

for (int i = 0; i < mas.Length; i++)

Console.WriteLine(i + " " + mas[i].ToString("F2"));

}

public static array operator+(array op1, array op2)

{

array temp = new array(op1.mas.Length);

for (int i = 0; i < op1.mas.Length; i++)

temp.mas[i] = op1.mas[i] + op2.mas[i];

return temp;

}

public static bool operator <(array op1, array op2)

{

bool b1 = true;

for(int i=0; i<op1.mas.Length; i++)

if(op1.mas[i]>op2.mas[i])

{

b1=false;

break;

}

return b1;

}

public static bool operator >(array op1, array op2)

{

bool b1 = true;

for (int i = 0; i < op1.mas.Length; i++)

if (op1.mas[i] < op2.mas[i])

{

b1 = false;

break;

}

return b1;

}

}

//----------------------------------------------

class cld:array

{

int KolPol;

public cld(int k) : base(k) { }

public void cnt(double c)

{

KolPol = 0;

foreach (double x in mas)

if (x > c) KolPol++;

}

public int ValKol

{

get

{ return KolPol; }

}

}

class Program

{

static void Main(string[] args)

{

cld my2 = new cld(6);

my2.inpt();

my2.cnt(7);

Console.WriteLine("Res=" + my2.ValKol);

/\* array ar1, ar2, ar3;

ar1 = new array(5);

ar2 = new array(5);

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

ar1[i] = 2.5 \* i;

Console.WriteLine();

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

Console.WriteLine(ar1[i].ToString("F1"));

; ar1.inpt();

Console.WriteLine();

ar2.inpt();

ar3 = ar1 + ar2;

ar3.otpt();

bool b2 = ar1 < ar2;

Console.WriteLine("Bool " + b2);

cla my;

my = new cla(4);

my.inpt();

int res;

my.fun1(out res);

Console.WriteLine("Res=" + res);\*/

Console.ReadLine();

}

}

}

Свойства

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Cons1009A

{

class clb

{

double a, b, c;

public void assgn(double a, double b, double c)

{ this.a=a; this.b=b; this.c=c; }

double fun1()

{

return 2.5 \* a + 1.5 \* b + 0.8 \* c;

}

public double prop1

{

get

{

return fun1();

}

}

public double aa

{

get

{

return a;

}

set

{

if (value > 0 && value < b + c)

a = value;

else

Console.WriteLine("Failed value");

}

}

public double per

{

get

{ return a + b + c; }

}

public void otpt()

{

Console.WriteLine(a + " " + b + " " + c);

}

}

class Program

{

static void Main(string[] args)

{

clb my = new clb();

my.assgn(4, 5, 6);

Console.WriteLine((my.prop1).ToString("F1"));

// my.aa = 80;

// my.otpt();

Console.ReadLine();

}

}

}