**11. Полиморфизм**

Задание 1. Составить программу с одним родительским классом и двумя потомками. Потомки должны содержать виртуальные функции. Создать виртуальную функцию выдачи результатов расчета методов на экран монитора с указанием названий и полей, и их значений соответствующего объекта. Составить тестирующую программу с выдачей протокола на экран монитора. При этом создать объекты базового и производных типов, используя полиморфный контейнер - массив ссылок базового класса на объекты базового и производных классов (количество объектов >=5).

namespace task1

{

using System;

class Program

{

static void Main(string[] args)

{

Vector[] vectors = new Vector[5];

// заполняем массив объектами базового и производных классов

vectors[0] = new Vector("Вектор 1");

vectors[1] = new TwoDimensionalVector("(первый)", 3.0, 4.0);

vectors[2] = new ThreeDimensionalVector("(первый)", 1.0, 2.0, 3.0);

vectors[3] = new TwoDimensionalVector("(второй)", -1.0, 5.0);

vectors[4] = new ThreeDimensionalVector("(Второй)", 4.0, -2.0, 0.0);

// вызываем виртуальную функцию для каждого объекта в массиве

foreach (Vector vector in vectors)

{

vector.PrintResults();

}

Console.ReadLine();

}

}

class Vector

{

protected string \_name;

public Vector(string name)

{

\_name = name;

}

public virtual void PrintResults()

{

Console.WriteLine($"Вектор {\_name}");

}

public virtual double Length()

{

return 0.0;

}

}

class TwoDimensionalVector : Vector

{

private double \_x;

private double \_y;

public TwoDimensionalVector(string name, double x, double y) : base(name)

{

\_x = x;

\_y = y;

}

public override void PrintResults()

{

Console.WriteLine($"Двухмерный вектор {\_name}:");

Console.WriteLine($"x: {\_x}");

Console.WriteLine($"y: {\_y}");

Console.WriteLine($"Длина: {Length()}");

}

public override double Length()

{

return Math.Sqrt(\_x \* \_x + \_y \* \_y);

}

}

class ThreeDimensionalVector : Vector

{

private double \_x;

private double \_y;

private double \_z;

public ThreeDimensionalVector(string name, double x, double y, double z) : base(name)

{

\_x = x;

\_y = y;

\_z = z;

}

public override void PrintResults()

{

Console.WriteLine($"Трехмерный вектор {\_name}:");

Console.WriteLine($"x: {\_x}");

Console.WriteLine($"y: {\_y}");

Console.WriteLine($"z: {\_z}");

Console.WriteLine($"Длина: {Length()}");

}

public override double Length()

{

return Math.Sqrt(\_x \* \_x + \_y \* \_y + \_z \* \_z);

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  |  |

Анализ результатов:

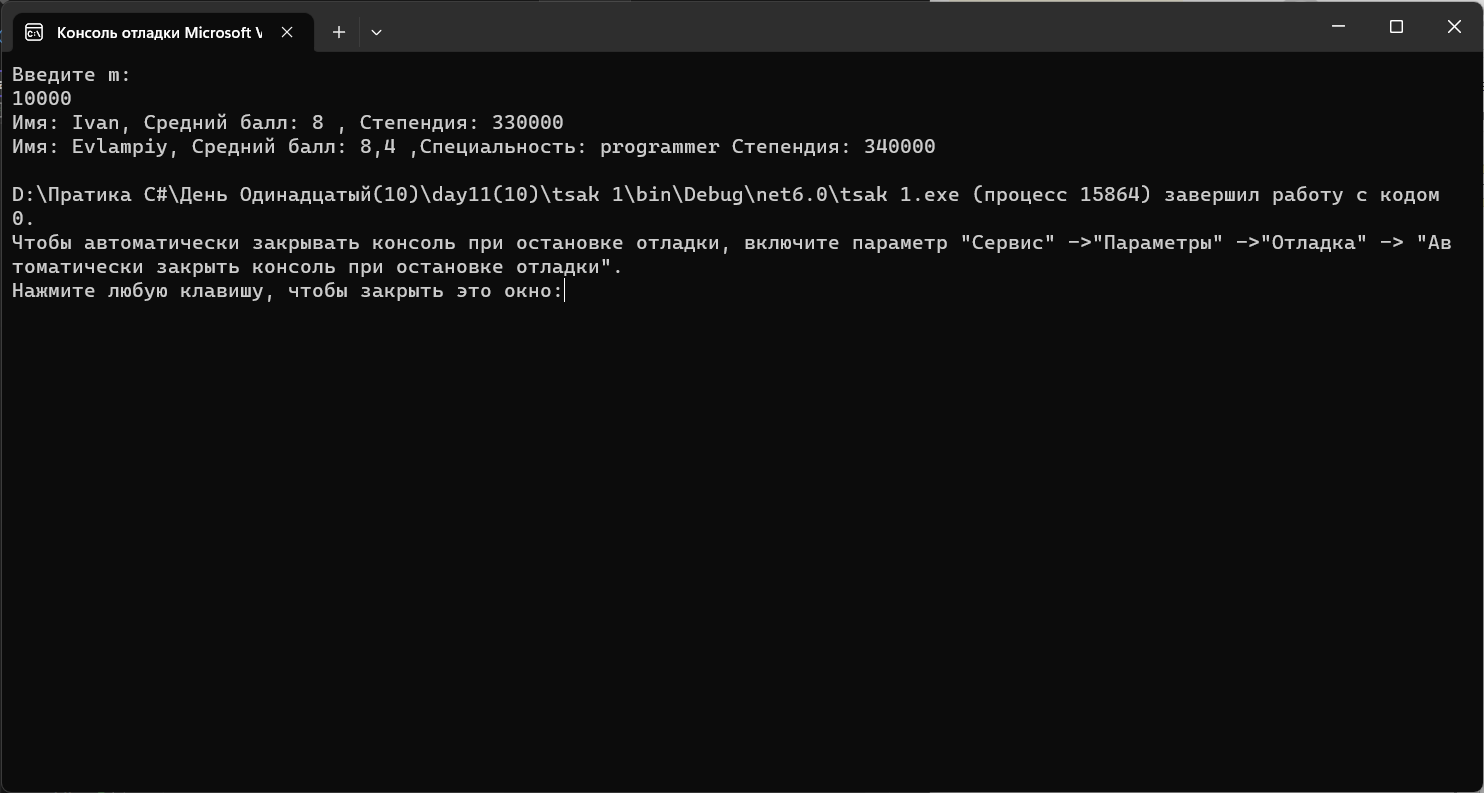


Рисунок 1.1 – Результат работы программы

Задание 2. Составить программу с абстрактным родительским классом и двумя объектами - потомками. Для этого модифицировать задание 2. Составить тестирующую программу с выдачей протокола на экран монитора. В ней нужно реализовать циклический вывод параметров объектов, используя полиморфный контейнер - массив объектов базового класса (количество объектов>=5).

internal class Program

{

static void Main(string[] args)

{

A a = new A();

B b = new B(4.3f);

Console.WriteLine(a.C);

Console.WriteLine(b.C);

Console.WriteLine(b.C2);

Console.ReadLine();

}

}

public class A

{

private float a = 3.2f;

private float b = 5.1f;

public float C

{

get { return a \*= b / a; }

}

}

class B : A

{

private float \_d;

public float C2

{

get

{

if (base.C > 5)

return base.C \* \_d;

else

return base.C / \_d;

}

}

public B(float d) : base()

{

\_d = d;

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | 5,1  5,1  21,93 |

Анализ результатов:

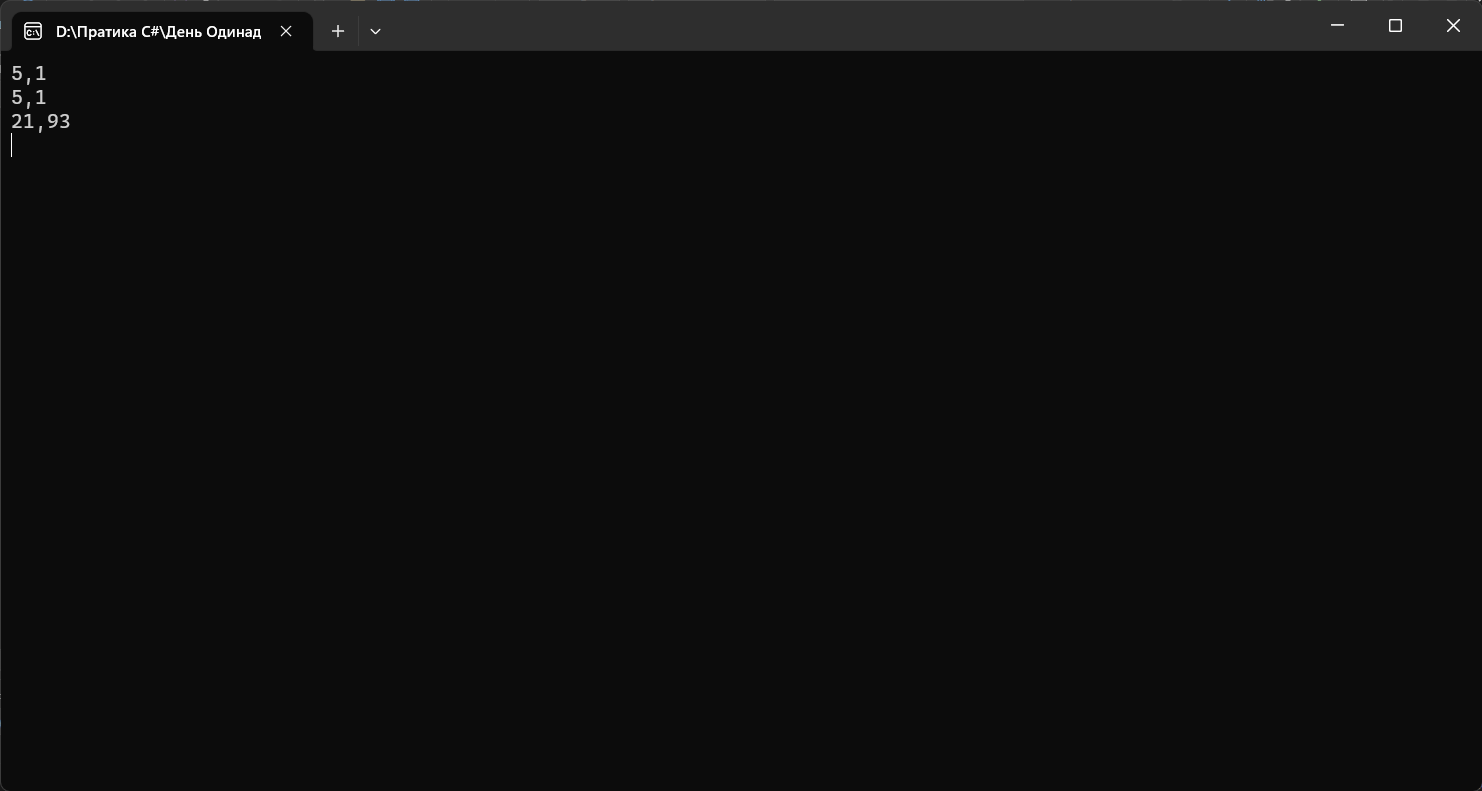
**

Рисунок 1.2 – Результат работы программы

работы программы