Отчет по практической работе №12

Наименование работы: полиморфизм.

Индивидуальные задания (вариант №9):

1. При выполнении данной работы необходимо определить базовый класс и производные от него классы.

В следующих заданиях требуется создать базовый класс и определить общие методы show(), get(), set() и другие, специфические для данного класса. Создать производные классы, в которые добавить свойства и методы. Все методы базового класса переопределять с помощью override. Один или несколько методов перегрузить.

Создать базовый класс «Учащийся» и производные классы «Школьник» и «Студент». Создать массив объектов базового класса и заполнить этот массив объектами. Показать отдельно студентов и школьников.

1. Дан класс, в котором есть одномерный массив, необходимо перегрузить оператор согласно варианту индивидуального задания.

|  |  |
| --- | --- |
| 9 Вариант | Класс — одномерный массив. Дополнительно перегрузить следующую операцию:  = = — проверка на равенство массивов |

Ход работы:

1. Разработал программу в соответствии с заданием №1. Были реализованы методы, классы, наследование, перегрузка, переопределение.

Листинг программы:

using System;

using System.Collections.Generic;

namespace Lab12

{

class Program

{

static void Main(string[] args)

{

Console.Write("Введите количество объектов: ");

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

Student st = new Student();

Shoolboy sb = new Shoolboy();

Base[] b = new Base[n];

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

{

b[i] = new Base();

Console.Write($"Введите имя объекта №{i + 1}: ");

b[i].Name = Console.ReadLine();

Console.Write($"Введите возраст объекта №{i + 1}: ");

b[i].Age = int.Parse(Console.ReadLine());

Console.Write($"Введите место обучения объекта №{i + 1}: ");

b[i].Geo = Console.ReadLine();

if (b[i].Geo.ToLower() == "школа" || b[i].Geo.ToLower() == "гимназия")

{

Console.Write($"Введите класс объекта №{i + 1}: ");

sb.Clas = Console.ReadLine();

sb.Name = b[i].Name;

sb.Age = b[i].Age;

sb.Geo = b[i].Geo;

}

else

{

Console.Write($"Введите группа объекта №{i + 1}: ");

st.Group = Console.ReadLine();

st.Name = b[i].Name;

st.Age = b[i].Age;

st.Geo = b[i].Geo;

}

}

st.GetInfo();

sb.GetInfo();

Console.WriteLine(st.ToString());

}

}

public class Base

{

public string Name { get; set; }

public int Age { get; set; }

public string Geo { get; set; }

public Base()

{ }

public Base(string name, int age, string geo)

{

Name = name;

Age = age;

Geo = geo;

}

public virtual void GetInfo()

{

Console.WriteLine($"Имя: {Name}, Возраст: {Age}, Место учебы: {Geo}");

}

}

public class Student : Base

{

public string Group { get; set; }

public Student()

{ }

public Student(string name, int age, string geo, string group):base(name, age, geo)

{

Group = group;

}

public override void GetInfo()

{

Console.WriteLine("Студенты: ");

Console.WriteLine($"Имя: {Name}, Возраст: {Age}, Место учебы: {Geo}, Группа: {Group}");

}

public override string ToString()

{

return $"Перегруженый метод: \n Имя: {Name}, Возраст: {Age}, Место учебы: {Geo}, Группа: {Group}";

}

}

public class Shoolboy : Base

{

public string Clas { get; set; }

public Shoolboy()

{ }

public Shoolboy(string name, int age, string geo, string clas):base(name, age, geo)

{

Clas = clas;

}

public override void GetInfo()

{

Console.WriteLine("Школьники: ");

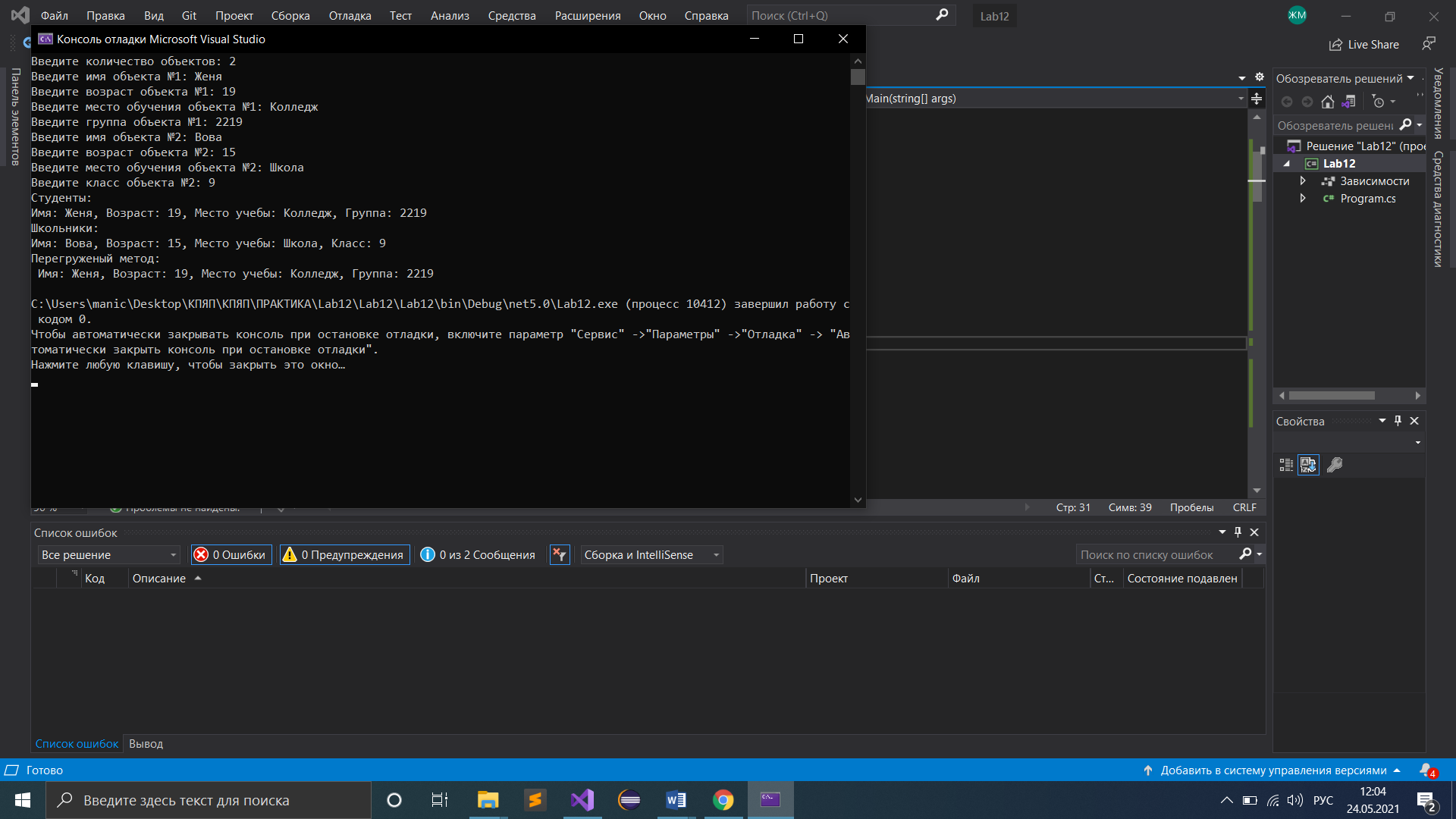
Console.WriteLine($"Имя: {Name}, Возраст: {Age}, Место учебы: {Geo}, Класс: {Clas}");

}

}

}

Результат выполнения программы Рис 1.1.



**Рисунок 1.1 – Пример работы программы c наследованием**

1. Разработал программу в соответствии с заданием №2. Были реализованы методы, классы, перегрузка.

Листинг программы:

using System;

namespace Lab12.\_2

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Hello World!");

Mass m1 = new Mass(new int[]{ 1,2,3,4});

Mass m2 = new Mass(new int[] { 1, 2, 3, 4 });

bool result = m1 == m2;

Console.WriteLine(result);

}

}

public class Mass

{

public int[] arr;

public Mass(int[] arr1)

{

arr = arr1;

}

public static bool operator ==(Mass arr1, Mass arr2)

{

{

if (arr1.arr.ToString()==arr2.arr.ToString())

{

return true;

}

return false;

}

}

public static bool operator !=(Mass arr1, Mass arr2)

{

if(arr1==arr2)

return false;

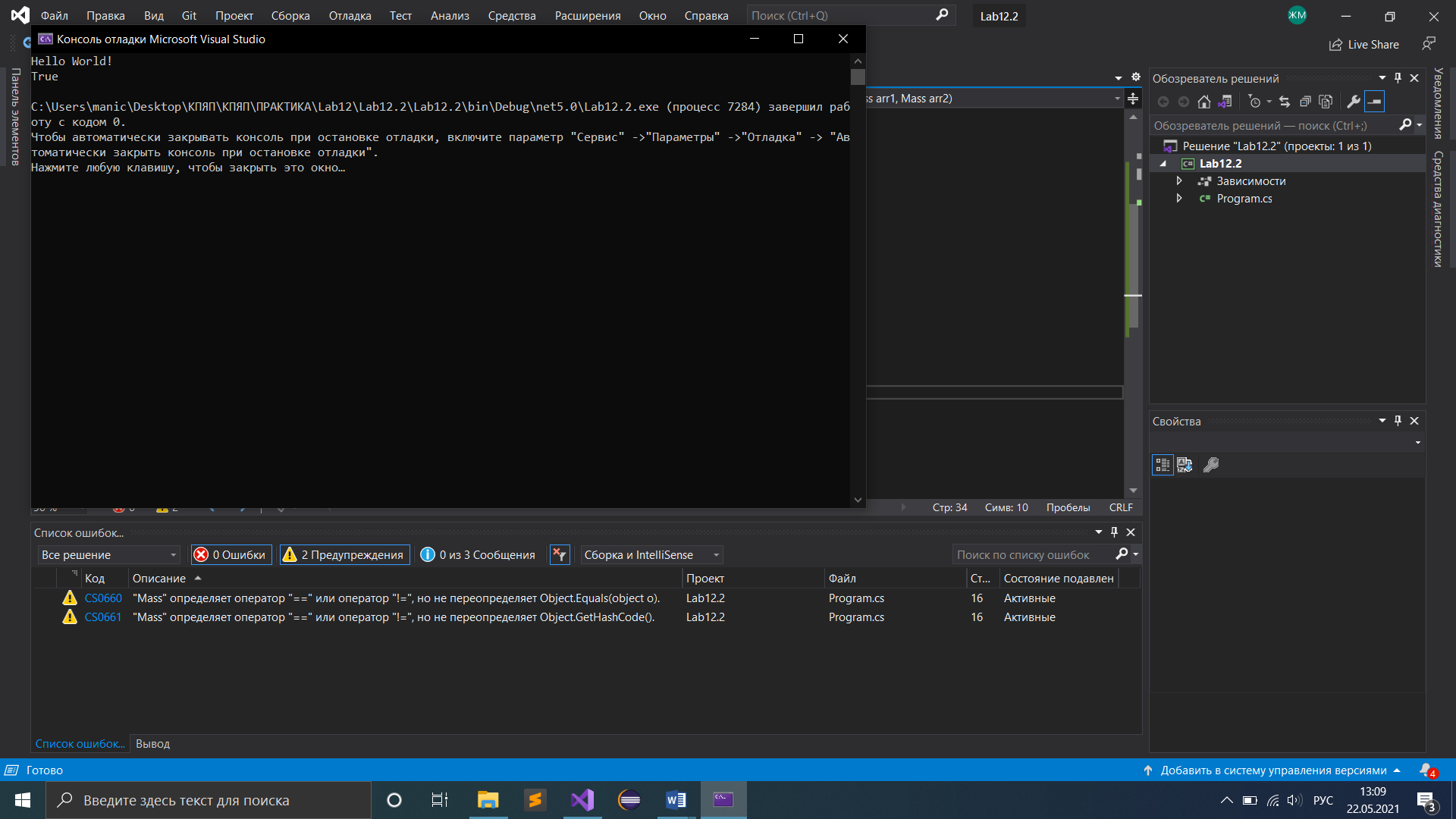
return true;

}

}

}

Результат выполнения программы Рис 2.1.



**Рисунок 2.1 – Пример работы программы c перегрузкой**

Отметка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Подпись преподавателя\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_