МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Курский государственный университет»

кафедра программного обеспечения и администрирования

информационных систем

Отчёт

по лабораторной работе №2

«Объекты и классы»

по дисциплине

###### «Объектно-ориентированные языки и системы»

|  |  |
| --- | --- |
| Выполнил(а): | студент группы 313  Козявин Максим  Сергеевич |
| Проверил: | к.т.н., доцент  кафедры ПОиАИС  Макаров К.С. |

Курск

2023

**Цель работы:** познакомиться с основой объектного подхода в языке

C#, созданием объектов и классов. Получить практические навыки разработки

классов и использования их в приложениях.

**Индивидуальное задание (вариант 4):**

**Задание 1:** Определить пользовательский класс в соответствии с вариантом

задания.

Вариант 4. Изделие

имя – string

шифр – string

количество – int

1. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.
2. Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор вызвал).
3. В программе предусмотреть использование массивов объектов.

**Задание 2:** Разработать структуру класса в соответствии с вариантом задания. Предусмотреть наличие соответствующих методов и внутренних полей класса.

Вариант 4. Составьте описание класса для представления комплексных чисел. Обеспечьте выполнение операций сложения, вычитания и умножения

комплексных чисел. Напишите программу, демонстрирующую работу с этим

классом. Программа должна содержать меню, позволяющее осуществить

проверку всех методов класса.

**Основные теоретические положения**

C# является объектно-ориентированным языком. Это значит, что программу на C# можно представить в виде взаимосвязанных взаимодействующих между собой объектов. Описанием объекта является класс, а объект представляет экземпляр этого класса.

**Экспериментальные результаты**

Код программы для решения задачи 1:

//Class.cs

namespace lab2 {

class Product {

private string name = "";

private string code = "";

private int count;

private static string text = "";

public Product() {

text = "Конструктор без параметров вызван для объекта <" + this.ToString() + ">\n";

}

public Product(string name, string code, int count) {

this.name = name;

this.code = code;

this.count = count;

text = "Конструктор с параметрами вызван для объекта <" + this.ToString() + ">\n";

}

public Product(Product product) {

this.name = product.name;

this.code = product.code;

this.count = product.count;

text = "Конструктор копироваия вызван для объекта <" + this.ToString() + ">\n";

}

public static string GetText() {

string res = text;

text = "";

return res;

}

public int Count {

get { return count; }

set { count = value; }

}

public string Name {

get { return name; }

}

public string Code {

get { return code; }

}

public override string ToString() {

string st = "name\"" + this.name + "\" code=\"" + this.code + "\" count=" + count.ToString();

return st;

}

}

}

//Program.cs

namespace lab2 {

internal class Program {

private static void Main(string[] args) {

Product[] productArray = new Product[3];

productArray[0] = new Product();

Console.WriteLine(Product.GetText());

productArray[1] = new Product("Стол", "ст1", 10);

Console.WriteLine(Product.GetText());

productArray[2] = new Product(new Product("Ноутбук", "нб12", 3));

Console.WriteLine(Product.GetText());

Console.ReadLine();

}

}

}

Код программы для решения задачи 2:

//Complex.cs

namespace lab22 {

class Complex {

private double real;

private double imaginary;

public Complex() {

real = 0;

imaginary = 0;

}

public Complex(double real, double imaginary) {

this.real = real;

this.imaginary = imaginary;

}

public override string ToString() {

if (imaginary < 0)

return $"{real}{imaginary}i";

return $"{real}+{imaginary}i";

}

public static Complex Add(Complex num1, Complex num2) {

return new Complex(num1.real + num2.real, num1.imaginary + num2.imaginary);

}

public static Complex Sub(Complex num1, Complex num2) {

return new Complex(num1.real - num2.real, num1.imaginary - num2.imaginary);

}

public static Complex Mul(Complex num1, Complex num2) {

return new Complex(num1.real\*num2.real - num1.imaginary\*num2.imaginary, num1.real\*num2.imaginary + num2.real\*num1.imaginary);

}

}

}

//Program.cs

namespace lab22 {

internal class Program {

private static void Main(string[] args) {

while (true) {

Console.WriteLine("Enter operation symbol +, -, \* or exit:");

var operation = Console.ReadLine();

if (operation == "exit")

break;

Console.WriteLine("Enter first number\nreal part:");

double r1 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("imaginary part:");

double i1 = Convert.ToDouble(Console.ReadLine());

Complex c1 = new Complex(r1, i1);

Console.WriteLine("\nEnter second number\nreal part:");

double r2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("imaginary part:");

double i2 = Convert.ToDouble(Console.ReadLine());

Complex c2 = new Complex(r2, i2);

switch (operation) {

case "+":

Console.WriteLine(Complex.Add(c1, c2));

break;

case "-":

Console.WriteLine(Complex.Sub(c1, c2));

break;

case "\*":

Console.WriteLine(Complex.Mul(c1, c2));

break;

default:

Console.WriteLine("Wrong input.");

break;

}

}

}

}

}

Тестирование:

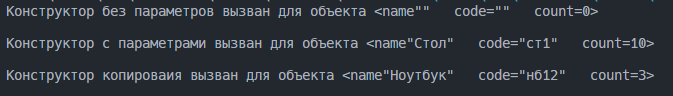


Рисунок 1 – Тест 1 задачи 1

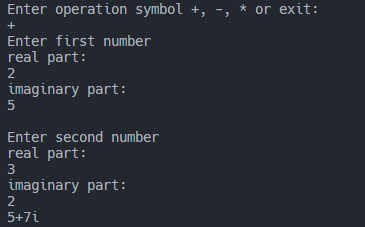


Рисунок 2 – Тест 1 задачи 2

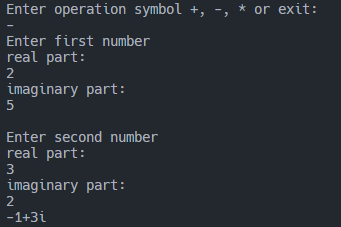


Рисунок 3 – Тест 2 задачи 2

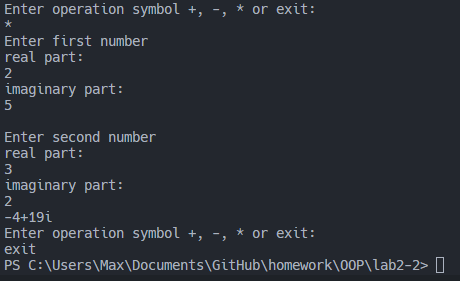


Рисунок 4 – Тест 3 задачи 2

**Вывод:** в данной лабораторной работе я познакомился с основой объектного подхода в языке C#, созданием объектов и классов и получил практические навыки разработки классов и использования их в приложениях.