МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования” Гомельский государственный университет им Ф. Скорины”

**ЛАБОРАТОРНАЯ РАБОТА КФ-Программирование-1-1-07 ПРОГРАММИРОВАНИЕ СОБСТВЕННЫХ КЛАССОВ НА ЯЗЫКЕ C#**

Выполнил:

Студент группы КФ-17

Гуревич П.А.

Проверил Дей Е.А.

Гомель 2022

**Цель работы:** изучение правил оформления и приемов создания собственных классов на языке C#

**ЗАДАЧА 1.** Создайте проект и повторите пример программирования класса. Выполните этот примеры. Результаты включите в отчет.

**ЗАДАЧА 2.** Дополните пример класса методами:

- добавление к каждому полю заданного значения

- вычисление произведения полей

- обмен значений полей

Реализуйте вызов этих методов для двух объектов с выбранными самостоятельно значениями.

Реализуйте вывод информации об объектах после выполнения каждого метода.

Результаты включите в отчет.

**ЗАДАЧА 3.** Создайте новый проект консольного приложения, в котором разработайте собственный класс в соответствии с вариантом задания.

▪ Состав класса обсудите с преподавателем.

▪ Для класса составьте: ‒ необходимые закрытые поля ‒ соответствующие свойства ‒ различные варианты конструкторов ‒ метод вывода параметров на экран ‒ методы вычисления характеристик объектов

▪ Элементы класса поясните комментариями

▪ Реализуйте вывод на экран описания класса и данных исполнителя при запуске программы 11

▪ Для использования класса в главной функции создайте 3 объекта класса и для каждого объекта реализуйте работу конструкторов, методов, свойств – на примерах, подобранных самостоятельно.

▪ Постановку задачи, расчетные формулы, программный проект и результаты вычислений включите в отчет.

**ЗАДАЧА 1.** Создайте проект и повторите пример программирования класса. Выполните этот примеры. Результаты включите в отчет.

using ConsoleApplication6;

using System;

namespace ConsoleApplication6

{

class IntClass

// обработка пары целых чисел

{

private int a, b; // поля

public int A

{

get

{

return a;

}

set

{

a = value;

}

}

public int B

{

get

{

return b;

}

set

{

b = value;

}

}

public IntClass()

{

a = 0;

b = 0;

}

public IntClass(int a, int b)

{

this.a = a;

this.b = b;

}

public int Summa()

{

return a + b;

}

public int Difference()

{

return a - b;

}

public void Info()

{

Console.WriteLine("\n Параметры объекта:");

Console.WriteLine("Поле a = {0} поле b = {1} \n", a, b);

}

public override string ToString()

{

string s;

s = a.ToString() + " " + b.ToString();

return s;

}

}

class Progran

{

static void Main(string[] args)

{

IntClass p1 = new IntClass();

p1.Info();

Console.WriteLine("Сумма = " + p1.Summa());

Console.ReadKey();

IntClass p2 = new IntClass(126, 154);

p2.Info();

Console.ReadKey();

int k = p2.Summa();

Console.WriteLine("Сумма = " + k);

Console.WriteLine("Разность = " + p2.Difference());

Console.WriteLine("Значения полей: " + p2.ToString());

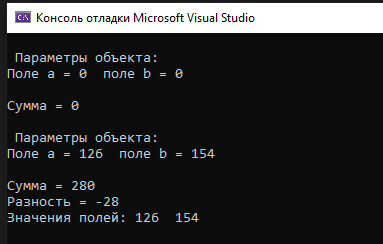
Console.ReadKey();

}

}

}

**Результат:**



**ЗАДАЧА 2.** Дополните пример класса методами:

- добавление к каждому полю заданного значения

- вычисление произведения полей

- обмен значений полей

Реализуйте вызов этих методов для двух объектов с выбранными самостоятельно значениями.

Реализуйте вывод информации об объектах после выполнения каждого метода.

Результаты включите в отчет.

using System;

namespace ConsoleApplication6

{

class IntClass

{

private int a, b;

public int A

{ get { return a; } set { a = value; } }

public int B

{ get { return b; } set { b = value; } }

public IntClass(int a, int b)

{

this.a = a;

this.b = b;

}

public int Intwork()

{

return a \* b;

}

public void Intexchange()

{

int c;

c = a;

this.a = this.b;

this.b = c;

}

public void Add(int k, int f)

{

this.a = this.a + k;

this.b = this.b + f;

}

public void Info()

{

Console.WriteLine("\n Параметры объекта");

Console.WriteLine("Поле a = {0} поле b = {1} \n ", a, b);

}

}

class Program

{

static void Main(string[] args)

{

IntClass p1 = new IntClass(12, 3);

p1.Info();

Console.WriteLine("Умножения = " + p1.Intwork());

p1.Intexchange();

p1.Info();

p1.Add(4, 5);

p1.Info();

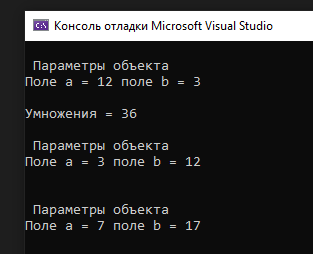
Console.ReadKey();

}

}

}

**Результат:**



**ЗАДАЧА 3.** Создайте новый проект консольного приложения, в котором разработайте собственный класс в соответствии с вариантом задания.

▪ Состав класса обсудите с преподавателем.

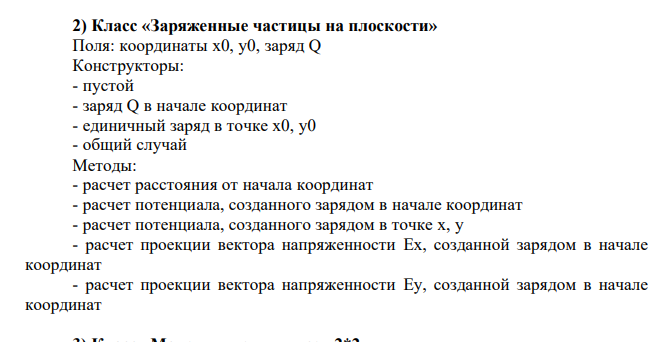
▪ Для класса составьте: ‒ необходимые закрытые поля ‒ соответствующие свойства ‒ различные варианты конструкторов ‒ метод вывода параметров на экран ‒ методы вычисления характеристик объектов

▪ Элементы класса поясните комментариями

▪ Реализуйте вывод на экран описания класса и данных исполнителя при запуске программы

▪ Для использования класса в главной функции создайте 3 объекта класса и для каждого объекта реализуйте работу конструкторов, методов, свойств – на примерах, подобранных самостоятельно.

▪ Постановку задачи, расчетные формулы, программный проект и результаты вычислений включите в отчет.



using System;

namespace ConsoleApplication6

{

class Vector

{

private int x2, y2, x1, y1; double Q; //--поля

public int A

{ set { x2 = value; } get { return x2; } } //--свойства

public int B

{ set { y2 = value; } get { return y2; } }//--свойства

public int Y

{ set { x1 = value; } get { return x1; } } //--свойства

public int N

{ set { y1 = value; } get { return y1; } }//--свойства

public double C

{ set { Q = value; } get { return Q; } }//--свойства

public Vector()//--пустой конструктор

{

Q = 0; x2 = 0; y2 = 0; x1 = 0; y1 = 0;

}

public Vector(double Q)//--Q в начале координат

{

this.Q = Q; this.x2 = 0; this.y2 = 0; this.x1 = 0; this.y1 = 0;

}

public Vector(int x2, int x1, int y1, int y2)//--единичный заряд в точке x1, y1, x2, y2

{

this.x2 = x2; this.y2 = y2; Q = 1.602176; this.x1 = x1; this.y1 = y1;

}

public Vector(double Q, int x2, int x1, int y1, int y2)//--общий случай

{

this.Q = Q; this.x2 = x2; this.x1 = x1; this.y1 = y1; this.y2 = y2;

}

public double Distanc()//--расстояние

{ return Math.Sqrt(Math.Pow((x2 - x1), 2) + Math.Pow((y2 - y1), 2)); }

public void Poten()//--потенциал в точке (0;0)

{ Console.WriteLine("Несуществует потенциала в точке "); }

public double Poten1() //потенциал

{ return (1 / (4 \* Math.PI) \* Q) / Math.Sqrt(Math.Pow((x2 - x1), 2) + Math.Pow((y2 - y1), 2)); }

public double Ox()//проекция на x

{ return Math.Sqrt(Math.Pow((x2 - x1), 2)); }

public double Oy()//проекция на y

{ return Math.Sqrt(Math.Pow((y2 - y1), 2)); }

}

class Program

{

static void Main(string[] args)

{

Vector p = new Vector(2, 6, 23, 2, 6);

Console.WriteLine("Растояние => " + p.Distanc());

Console.WriteLine("Потенциал в точке x,y => " + p.Poten1());

Console.WriteLine("Проекция на ось Ox => " + p.Ox());

Console.WriteLine("Проекция на ось Oy => " + p.Oy());

}

}

}

**Результат:**

