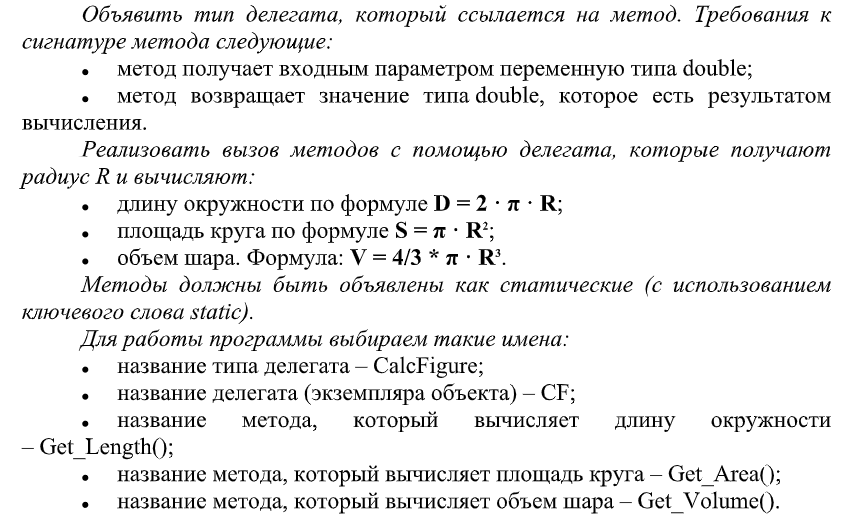
**Механизм наследования. Полиморфизм**

**Задание №1.**



Листинг программы:

tatic void Main(string[] args)

{

const int MIN = 1;

const int MAX = 100;

var length = new CalcFigure(GetLength);

double radius = Faker.RandomNumber.Next(MIN, MAX) \* 0.1;

WriteLine($"Длина окружности: {length(radius):f2} см.");

var area = new CalcFigure(GetArea);

WriteLine($"Площадь круга: {area(radius):f2} см2.");

var volume = new CalcFigure(GetVolume);

WriteLine($"Объём шара: {volume(radius):f2} см3.");

ReadKey();

}

Анализ результатов:

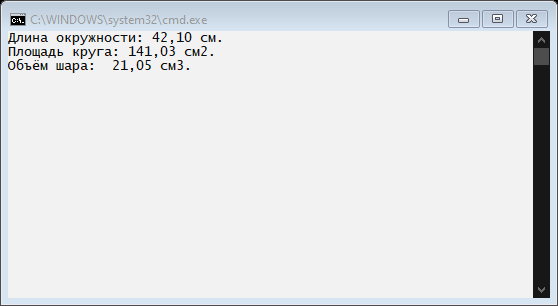


Рисунок 1.1 – Результат работы программы

**Задание № 2.** Используя Visual Studio, создайте проект по шаблону Console Application. Создайте четыре лямбда оператора для выполнения арифметических действий: (Add – сложение, Sub – вычитание, Mul – умножение, Div – деление). Каждый лямбда оператор должен принимать два аргумента и возвращать результат вычисления. Лямбда оператор деления должен делать проверку деления на ноль. Написать программу, которая будет выполнять арифметические действия указанные пользователем.

Листинг программы:

static void Main(string[] args)

{

try

{

Calculate mathematic = new();

Write("Введие а: ");

var a = ToInt32(ReadLine());

Write("Введие b: ");

var b = ToInt32(ReadLine());

WriteLine($"add = {mathematic.Add(a, b)}");

WriteLine($"sub = {mathematic.Sub(a, b)}");

WriteLine($"mul = {mathematic.Mul(a, b)}");

WriteLine($"div = {mathematic.Div(a, b)}");

}

catch (DivideByZeroException)

{

WriteLine("Делить на 0 нельзя!");

}

catch (FormatException)

{

WriteLine("Неверный формат!");

}

ReadKey();

}

}

Анализ результатов:

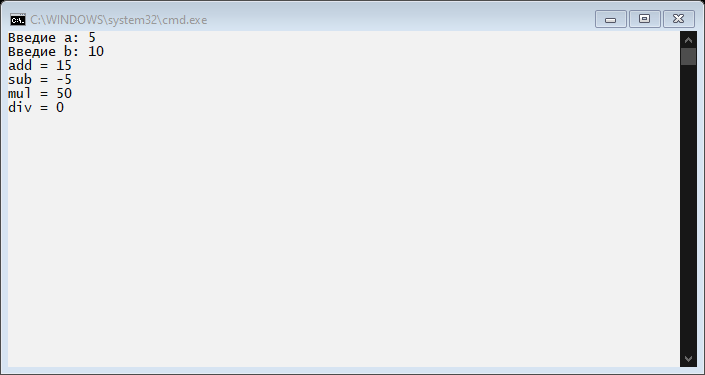


Рисунок 1.2 – Результат работы программы

**Задание № 3.** Создать 3 метода для работы со строкой. Создать делегат. В программе вызывать все три метода при помощи делегата.

Листинг программы:

static void Main(string[] args)

{

StringManager stringManager = new();

var join = new MyStringDelegate(stringManager.JoinString);

Write("Введите строку: ");

var mainString = ReadLine();

var joinedString = join(mainString);

WriteLine($"joined: {joinedString}");

var remove = new MyStringDelegate(stringManager.RemoveString);

var removedString = remove(joinedString);

WriteLine($"removed: {removedString}");

var upper = new MyStringDelegate(stringManager.MakeStringUpper);

var upperedString = upper(removedString);

WriteLine($"uppered: {upperedString}");

ReadKey();

}

Анализ результатов:

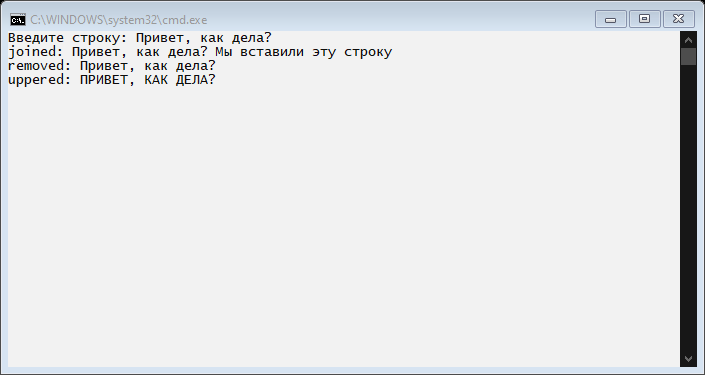


Рисунок 1.3 – Результат работы программы

**Задание № 4.** Используя Visual Studio, создайте проект по шаблону Console Application. Создайте анонимный метод, который принимает в качестве аргумента массив делегатов и возвращает среднее арифметическое возвращаемых значений методов сообщенных с делегатами в массиве. Методы, сообщенные с делегатами из массива, возвращают случайное значение типа int.

Листинг программы:

static void Main(string[] args)

{

var d1 = new Func<Action<int>, int> (DelegateService.GetRandomNum);

var d2 = new Func<Action<int>, int>(DelegateService.GetRandomNum);

var d3 = new Func<Action<int>, int>(DelegateService.GetRandomNum);

var delegates = new List<Func<Action<int>, int>>() { d1,d2,d3};

var average = new Func<List<Func<Action<int>, int>>, Action<int>, double>(DelegateService.GetAverage);

var averageIn = average(delegates, (n) => {WriteLine(n); });

WriteLine($"Среднее: {averageIn}");

ReadKey();

}

Анализ результатов:

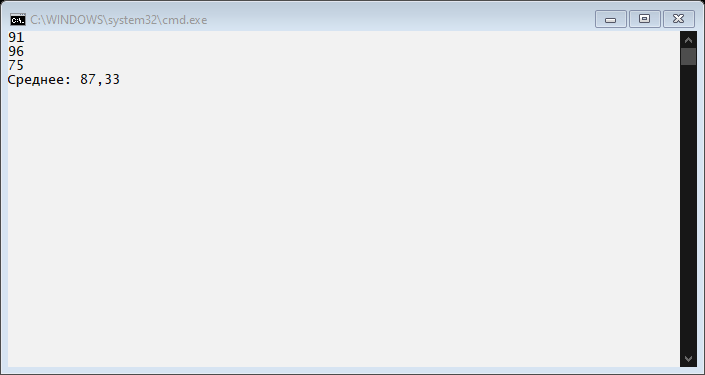


Рисунок 1.4 – Результат работы программы