

Лабораторная работа 1.

В работе определяется абстрактный базовый класс и два производных класса. В одном производном классе данные хранятся в массиве, в другом – в одном из типов коллекций стандартной библиотеки .NET. Привязка типов к данным измерений некоторых физических величин условная, так как типы, которые определяются в лабораторных работах, это “учебные” типы для изучения синтаксических конструкций языка C# и стандартной библиотеки .NET.

Типы, определенные в лабораторной работе, будут использоваться в других лабораторных работах, в том числе в следующем семестре.

Вариант 4

В лабораторной работе надо определить типы для работы с данными измерений двух компонент электромагнитного поля на множестве точек на двумерной плоскости:

- **struct DataItem** для координат (x, y) точки и значений компонент поля в этой точке;
- абстрактный базовый класс **V4Data** и два производных от него класса **V4DataList** и **V4DataArray**;
- **class V4DataList** для значений поля на множестве точек на двумерной плоскости, которые хранятся в виде коллекции **List<DataItem>**;
- **class V4DataArray** для значений поля на равномерной сетке, которые хранятся в прямоугольном двумерном массиве типа **System.Numerics.Vector2[,]**;
- **class V4MainCollection** для коллекции объектов типа **V4DataList** и **V4DataArray**;
- статический класс с методами для вычисления значений поля в точке на двумерной плоскости.

Структура **DataItem** содержит открытые автореализуемые свойства

- типа **System.Numerics.Vector2** с координатами (x,y) двумерной точки; координата **x** хранится в первой компоненте **System.Numerics.Vector2**, координата **y** – во второй;
- типа **System.Numerics.Vector2** для компонент вектора значений поля в этой точке.

В структуре **DataItem** определены открытые

- конструктор с параметрами типа **Vector2** (координаты x и y точки измерения) и **Vector2** (значения компонент поля) для инициализации данных структуры;
- метод **string ToLongString(string format)**, возвращающий строку, которая содержит значения координат точки измерения, значения компонент поля и значение модуля вектора поля; параметр **format** задает формат вывода чисел с плавающей запятой;
- перегруженная (**override**) версия виртуального метода **string ToString()**.

Определить делегат **Vector2 Fv2Vector2 (Vector2 v2)**.

Абстрактный базовый класс **V4Data** имеет открытые автореализуемые свойства

- типа **string** с методом **get** для идентификации объекта данных;
- типа **DateTime** с методом **get** для даты измерений поля.

Класс **V4Data** содержит открытые

- конструктор с параметрами типа **string** и **DateTime**;
- абстрактное свойство **Count** типа **int** (только с методом **get**);
- абстрактное свойство **MaxFromOrigin** типа **float** (только с методом **get**);
- абстрактный метод **string ToLongString(string format)**;
- перегруженную (**override**) версию виртуального метода **string ToString()**;

Класс **V4DataList** является производным от класса **V4Data**. В классе **V4DataList** данные измерений хранятся в коллекции **List<DataItem>**. Каждый элемент коллекции **DataItem** содержит координаты точки, в которой измерено поле, и значения компонент вектора поля.

В классе **V4DataList** определить открытые

- автореализуемое свойство типа **List<DataItem>** с методом **get**;
- конструктор с параметрами типа **string** и **DateTime** для инициализации данных базового класса; в этом конструкторе распределяется память для **List<DataItem>**;
- метод **bool Add(DataItem newItem)** для добавления в список **List<DataItem>** нового элемента **newItem**; метод добавляет новый элемент только в том случае, когда в коллекции **List<DataItem>** нет элемента, у которого координаты точки, в которой измерено поле, совпадают с координатами точки измерения элемента **newItem**; в этом случае метод возвращает значение **true**; в противном случае новый элемент не добавляется и возвращается значение **false**;
- метод **int AddDefaults(int nItems, Fv2Vector2 F)**, в котором (по какому-нибудь алгоритму) выбираются **nItems** точек на плоскости, для этих точек с помощью метода **F** вычисляются значения компонент электромагнитного поля, создаются элементы **DataItem** и добавляются в список **List<DataItem>**; при добавлении элементов **DataItem** надо использовать метод **bool Add(DataItem newItem)**; метод возвращает число элементов, добавленных в список **List<DataItem>**; число добавленных элементов может быть меньше, чем **nItems**, если в списке уже есть данные для выбранных по умолчанию точек измерения;
- реализацию абстрактного свойства **Count**, которое возвращает число точек, в которых измерено поле (число элементов в **List<DataItem>**);
- реализацию абстрактного свойства **MaxFromOrigin** типа **float**, которое возвращает максимальное расстояние между началом координат и точками, в которых измерено поле;
- перегруженную (**override**) версию виртуального метода **string ToString()**, который возвращает строку с именем типа, данными базового класса и числом элементов в списке **List<DataItem>**;
- реализацию абстрактного метода **string ToLongString(string format)**, который возвращает строку с такими же данными, что и метод **ToString()**, и дополнительно информацию о каждом элементе из **List<DataItem>** – координаты точки измерения,

значения компонент вектора поля и значение модуля поля; параметр **format** задает формат вывода чисел с плавающей запятой.

Класс **V4DataArray** является производным от класса **V4Data**. В классе **V4DataArray** данные измерений на двумерной прямоугольной сетке с равномерным шагом по осям *Ox* и *Oy* хранятся в двумерном прямоугольном массиве типа **Vector2[,]**. В классе хранятся параметры сетки – шаг по оси *Ox*, шаг по оси *Oy*, число узлов по оси *Ox*, число узлов по оси *Oy*. Предполагается, что координаты первых узлов сетки по осям *Ox* и *Oy* равны нулю.

Класс **V4DataArray** содержит открытые

- автореализуемые свойства для параметров сетки с методом **get** – типа **int** для числа узлов по оси *Ox*, типа **int** для числа узлов по оси *Oy*, типа **Vector2** для шагов сетки по осям *Ox* и *Oy*;
- автореализуемое свойство с методом **get** типа **Vector2 [,]** для двумерного прямоугольного массива значений поля в узлах сетки;
- конструктор с параметрами типа **string**, **DateTime** для инициализации данных базового класса; в этом конструкторе распределяется память для массива **Vector2[,]** с нулевым числом элементов;
- конструктор с параметрами типа
 - **string**, **DateTime** для инициализации данных базового класса;
 - **int** (число узлов по оси *Ox*), **int** (число узлов по оси *Oy*), **Vector2** (шаги сетки по осям *Ox* и *Oy*) для инициализации сетки;
 - **Fv2Vector2** для вычисления компонент вектора поля в узлах сетки;в конструкторе распределяется память для массива **Vector2 [,]** и для каждого узла сетки вызывается метод типа **Fv2Vector2**, который вычисляет значения компонент вектора поля в узле;
- реализацию абстрактного свойства **Count**, которое возвращает число точек, в которых измерено поле (число узлов сетки);
- реализацию абстрактного свойства **MaxFromOrigin** типа **float**, которое возвращает максимальное расстояние между началом координат и точками, в которых измерено поле;
- перегруженную (**override**) версию виртуального метода **string ToString ()**, который возвращает строку с именем типа, данными базового класса и параметрами сетки;
- реализацию абстрактного метода **string ToLongString (string format)**, который возвращает строку с такими же данными, что и метод **ToString ()**, и дополнительно информацию о каждом узле сетки – координаты, значения компонент вектора поля и значение модуля поля; параметр **format** задает формат вывода чисел с плавающей запятой;
- оператор преобразования типа **V4DataArray** к типу **V4DataList**; в коде оператора преобразования для каждого узла сетки создается объект типа **DataItem**, который добавляется в список **List<DataItem>** класса **V4DataList**.

Класс **V4MainCollection** содержит коллекцию элементов **V4DataList** и **V4DataArray**.

Класс **V4MainCollection** содержит

- закрытое поле типа **List<V4Data>**;
- открытое свойство **Count** типа **int**, которое возвращает число элементов в списке **List<V4Data>**;
- открытый индексатор типа **V4Data** с целочисленным индексом (только с методом **get**), который возвращает элемент списка **List<V4Data>** с заданным индексом;
- открытый метод **bool Contains (string ID)**, который возвращает значение **true**, если среди элементов списка **List<V4Data>** есть элемент, у которого значение свойства типа **string** для идентификации объекта **V4Data** равно ID, и **false** в противном случае;
- открытый метод **bool Add (V4Data v4Data)** для добавления в коллекцию **List<V4Data>** нового элемента; метод добавляет новый элемент только в том случае, когда в коллекции **List<V4Data>** нет элемента с таким же значением свойства типа **string** для идентификации объекта, как и у значения параметра **v4Data** метода; метод возвращает значение **true**, если элемент был добавлен, и значение **false** в противном случае;
- открытый метод **string ToLongString (string format)**, который возвращает строку с информацией о каждом элементе списка **List<V4Data>**; при создании строки для каждого элемента из списка **List<V4Data>** вызывается метод **ToLongString (string format)**;
- перегруженную(**override**) версию виртуального метода **string ToString()**, который возвращает строку с информацией о каждом элементе списка **List<V4Data>**; при создании строки для каждого элемента из списка **List<V4Data>** вызывается метод **ToString()**.

Статический класс содержит статические методы, отвечающие делегату **Fv2Vector2**. Эти методы используются как фактические значения параметров типа **Fv2Vector2** в методе **AddDefaults(int nItems, Fv2Vector2 F)** класса **V4DataList** и в конструкторе класса **V4DataArray**.

В методе **Main()**

1. Создать объект типа **V4DataArray**, вывести его данные с помощью метода **ToLongString (string format)**. С помощью оператора преобразования, определенного в классе **V4DataArray**, преобразовать его в объект типа **V4DataList**, вывести данные **V4DataList** с помощью метода **ToLongString (string format)**. Для исходного объекта **V4DataArray** и для объекта **V4DataList**, полученного в результате преобразования, вывести значения свойств **Count** и **MaxFromOrigin**.
2. Создать объект типа **V4MainCollection**, с помощью метода **bool Add (V4Data v4Data)** добавить в коллекцию два элемента типа **V4DataArray** и два элемента типа **V4DataList** и вывести данные объекта **V4MainCollection** с помощью метода **ToLongString (string format)**.
3. Для всех элементов из **V4MainCollection** вызвать свойства **Count** и **MaxFromOrigin** и вывести их значения.

Срок сдачи лабораторной работы
12 октября