

Лабораторный практикум по дисциплине
«ЯЗЫКИ ПРОГРАММИРОВАНИЯ С СТРУКТУРЫ ДАННЫХ»
направление 020302.62 «Фундаментальная информатика и информационные технологии»
семестр 2

Лабораторная работа №4
Язык программирования C#: Стандартные интерфейсы

Теоретический материал:

1. Павловская Т.А. C#. Программирование на языке высокого уровня.

Задание 1.

Прочитать теоретический материал.

Задание 2.

В классах `ArrayVector` и `LinkedListVector` переопределить унаследованный от класса `Object` метод `Equals()` таким образом, чтобы он сравнивал на равенство любой объект типа `IVectorable`. Вектора считаются равными, если они равны и по числу координат, и по координатам. Также можно переопределить метод `GetHashCode()`.

Задание 3

Сделать классы `ArrayVector` и `LinkedListVector` реализующими интерфейс `Comparable`, и реализовать в них метод `CompareTo()` – метод сравнивает вектора типа `IVectorable` по числу их координат.

Описать дополнительный класс, реализующий интерфейс `IComparer`. Реализовать метод `Compare()` интерфейса `IComparer`, который сравнивает два вектора типа `IVectorable` по их модулю.

В методе `Main()` класса `Program` создать массив векторов (ссылок типа интерфейс), хранящий попеременно вектора разного типа. Найти в этом массиве вектора с минимальным и максимальным числом координат, используя метод `CompareTo()`.

Отсортировать массив векторов по возрастанию их модулей, используя метод `Compare()`.

Задание 4

Сделать классы `ArrayVector` и `LinkedListVector` реализующими интерфейс `ICloneable` и реализовать в них метод `Clone()`, выполняющий глубокое клонирование объектов.

В методе `Main()` класса `Program` выбрать один из векторов в массиве, выполнить его клонирование, продемонстрировать результат клонирования (например, изменив один из векторов – клонируемый или клон – и вывести на экран оба вектора для сравнения, возможно также использование метода `Equals()`).

Задание 5.

Протестировать работу приложения в классе `Program`, разработать адекватный интерфейс пользователя. Необходимо отлавливать и обрабатывать все возможные исключения. Воспользоваться пользовательским интерфейсом из лабораторной работы 3.

Задание 6.

Подготовить отчет о работе.