Задание 5.

Реализуйте обобщённый (generic) класс LinkedList<T>, репрезентирующий коллекцию вида односвязный список. Тип элемента списка LinkedListNode<T> должен быть описан на уровне типа односвязного списка.

Для класса реализуйте следующий функционал:

- Конструктор от односвязного списка LinkedList<T>
- Конструктор от IEnumerable<T>
- Индексатор, принимающий на вход порядковый номер элемента списка
- Объектный метод для вставки значения типа T в начало списка
- Объектный метод для вставки значения типа T в конец списка
- Объектный метод для вставки значения типа T на определённую позицию, задаваемую номером элемента типа int
- Объектный метод для удаления значения типа T из начала списка
- Объектный метод для удаления значения типа T из конца списка
- Объектный метод для удаления значения типа T, расположенного на определённой позиции, задаваемой номером элемента типа int
- Объектный метод для разворота (reverse) списка без модификации исходного списка
- Статический метод operator!, декорирующий вышеописанный метод разворота
- Статический метод для конкатенации двух списков без разрушения исходных списков
- Статический метод operator+, декорирующий вышеописанный метод конкатенации
- Статический метод для пересечения двух списков без разрушения исходных списков, со сравнением элементов списков через подаваемую в метод реализацию интерфейса IEqualityComparer<T>
- Статический метод operator&, декорирующий вышеописанный метод пересечения (hint: используйте свойство EqualityComparer<T>.Default)
- Статический метод для объединения двух списков с удалением дубликатов элементов без разрушения исходных списков, со сравнением элементов списков через подаваемую в метод реализацию интерфейса IEqualityComparer<T>
- Статический метод operator|, декорирующий вышеописанный метод пересечения (hint: используйте свойство EqualityComparer<T>.Default)
- Статический метод для удаления из первого списка элементов, которые имеют во втором списке хотя бы один элемент, равный по переданной в метод реализации интерфейса IEqualityComparer<T>
- Статический метод operator-, декорирующий вышеописанный метод “вычитания” (hint: используйте свойство EqualityComparer<T>.Default)
- Семейство перегруженных объектных методов для сортировки списка по переданному компаратору типа IComparer<T> / Comparer<T> / Comparison<T> (допускается использование реализованных классов из задания 3)
- Объектный метод, принимающий на вход объект делегата типа Action<T>, применяемый к каждому элементу списка
- Статические методы operator== и operator!= для сравнения по значению объектов переданных списков

- Статический метод operator*, возвращающий новый список, имеющий длину, равную меньшей из длин исходных списков, с применением метода operator* для элементов исходных списков, стоящих на одинаковых индексах (hint: используйте тип dynamic)

Для Вашего класса односвязного списка необходимо переопределить методы: object.GetHashCode, object.Equals, object.ToString. Также для класса необходимо реализовать интерфейсы: IEquatable<LinkedList<T>>, ICloneable, IEnumerable<T>.

Продемонстрируйте выполнение реализованного функционала (при демонстрации все потенциально возможные исключения должны быть обработаны; каждый тип обрабатываемого исключения необходимо поместить в отдельный блок catch).