SIT221 –DATA STRUCTURES AND ALGORITHMS

LAB4: HOW LINKED LISTS WORK?

LAB OBJECTIVE:

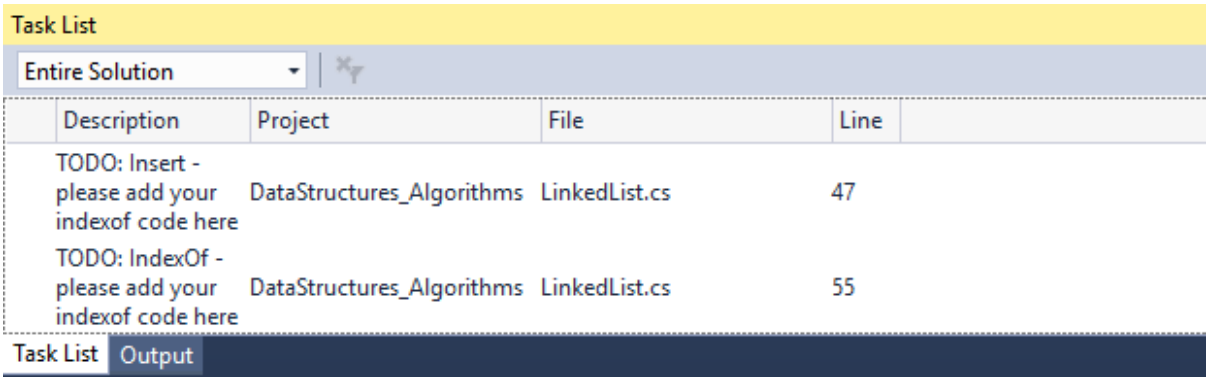
Understand how Linked Lists work

SUBMISSION INSTRUCTIONS

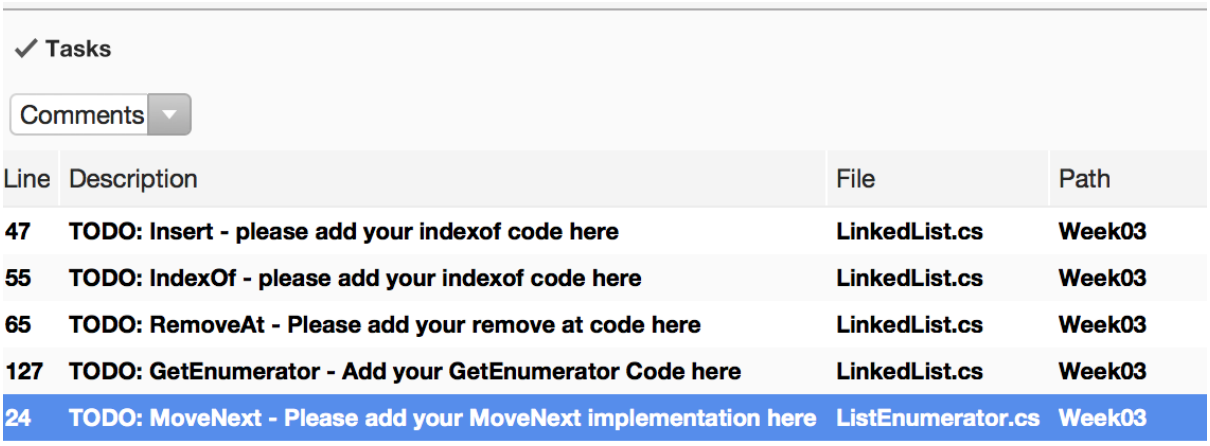
Please submit your work to Week04 assignment folder. Make sure to include your implementation in Week4 folder, and zip the whole solution, and submit it.

PREPARATION

1. Before you get started please have a look on the .NET framework Linked List class: https://msdn.microsoft.com/en-us/library/he2s3bh7(v=vs.110).aspx
2. Download the template project available in week04 resources folder. The solution has two projects: **DataStructures_Algorithms & Runner** projects.
3. In the Runner project, there is a Data Folder for Week01 – Note that the dataset has three files (1H.txt, 1T.txt, 1M.txt) [1H = 100 points, 1T = 1000 points, 10T = 10,000 points]
4. How to find what methods you need to complete? You will not be able to complete until you understand the whole solution, but here is a new way you can try, if you want
   In visual studio – You can from the View menu open the Task list and filter comments: you should see something like the below screen:



In Xamarin Studio: You should have similar window:



LAB TASKS

1. LINKEDLIST CLASS

In this task, we want to complete the **LinkedList** class to support the following methods, please check week04 folder for the given files:

1. **public int IndexOf (T element):** This method traverses the linked list looking for the first match of element in the linked list. You may need to use EqualityComparer, see here: https://msdn.microsoft.com/en-us/library/ms132123(v=vs.110).aspx
2. **public int RemoveAt(int index):** This method traverses the linked list until element number **index, it then removes it and** updates the elements before and after, as well as the count.
3. **public void Insert(T element, int index):** This method should insert the input element at the specified index in the list and adjust linked before this element, as well as the count.
4. Currently the LinkedList class does not support foreach. In order to enable foreach, we need to implement IEnumerable & IEnumerator interfaces, see here: https://msdn.microsoft.com/enus/library/system.collections.ienumerable(v=vs.110).aspx
   and
   https://msdn.microsoft.com/en-us/library/system.collections.ienumerator(v=vs.110).aspx

5. Our current LinkedList implements IEnumerable<T> interface. You need to implement the **GetEnumerator()** method which returns Enumerator object.
6. **ListEnumerator** class which implements the IEnumerator interface is given to you. Most methods of this class have been implemented. You are required to implement the **MoveNext** method.

2. TEST LINKEDLIST

In this task, you need to test the methods of the LinkedList that you just have implemented.