CSci 1933 Lab 7

October 27, 2015

1. Introduction

The purpose of this lab is to deepen your understanding of the machinery of LinkedList. This lab is designed to be a very straightforward one: you are asked to implement a basic singly linked list class with its relevant methods.

2. Your tasks

2.1. Recall the structure of the LinkedList: it is a collection of nodes that represents a sequence. Every node consists of 2 parts - the first part is the data and the second part is the reference to the next node. The LinkedList starts with the first node of the sequence and the last node will point to the terminator to signify the end of the LinkedList.

2.2. The linked list class you developed will be named `LinkedList1933<T>`. `LinkedList1933<T>` will implement the List interface. However, to reduce the amount of the work you will do, you are only asked to implement the following methods in detail:

Return type	Function name	Description
`T`	`remove(int index)`	Removes the element at the specified position in this list (optional operation).
`int`	`size()`	Returns the number of elements in this list.
`T`	`get(int index)`	Returns the element at the specified position in this list.
`void`	`add(int index, T obj)`	Inserts the specified element at the specified position in this list (optional operation).
`boolean`	`add(T object)`	Appends the specified element to the end of this list (optional operation).
`int`	`indexOf(T object)`	Returns the index of the first occurrence of the specified element in this list, or -1 if this list does not contain the element.
`boolean`	`contains(T object)`	Returns true if this list contains the specified element.

	`LinkedList1933<T>()`	An constructor that will instantiate the object and properly set the class fields..

For all other methods that are required by the `List` interface but are not specified above, they should be implemented to throw an UnsupportedOperationException. Hint: recall from previous lab how IntelliJ can help you auto implement the methods.

2.3. For the fields of the `LinkedList1933<T>`, you should have only two:

- LinkedListNode<T> head
- int listCount;

When you implement, please think carefully regarding what is a meaningful "terminator" of your LinkedList.

2.4. `LinkedList1933<T>` contains a sequence of nodes. For the data structure of the backing node, we've provided you with the `LinkedListNode` in `LinkedListNode.java` file. Please use this class as the node type.

3. Testing

We've provided you with a couple of JUnit test to help you verify your implementation.

3.1. Add JUnit to your project. To do this, do the following:

- Select "Project structure" from the "File" menu.
- Go to the "Libraries" group, click the little green plus (look up), and choose "From Maven...".
- Search for "junit" -- you're looking for something like "junit:junit:4.8.1" (latest stable option), and click OK.

3.2. Create Run/Debug configuration for JUnit:

- On the main menu, please "Run"
- Click "Edit Configuration"
- Click the plus sign in upper left corner and select "JUnit"
- Fill out the configuration information. Necessarily, put the ListTest in the "Class" filed.