Lab 2

Recursive LinkedCollection

In this lab you will implement the CollectionInterface with a recursive linked-based solution. You will use GenericTester.java to test your solution.

1. Create a new Java Project in Eclipse called lab2.

2. You have been given 3 Java files, CollectionInterface.java, Contact.java and

   GenericTester.java. Put those 3 files in the src folder of your Eclipse workspace and

   refresh the project in Eclipse.

3. Implement the CollectionInterface given to you. Make sure to read the Javadoc comments above each abstract method to implement correctly. The class implementation should be called RecursiveLinkedCollection<T>.
4. RecursizeLinkedCollection should have 2 instance variables – a front node and an integer size.

5. The following methods **must** all use recursion in its implementation:
   a. add – a recursive add method adds to the end of the list and returns an LLNode<T>. You must use the implementation taught in lecture.
      private LLNode<T> recAdd(LLNode<T> node, T info) {}
   b. remove **–** recursive method must return an LLNode<T>
      private LLNode<T> recAdd(LLNode<T> node, T info) {}
   c. get
   d. contains
   e. size

6. You must also implement a toString method in the RecursiveLinkedCollection class. It does not have to use recursion. toString is used in the tester class.
7. Use the GenericTester class to test your implementation.
8. Submit RecursiveLinkedCollection.java only.

Assumptions:

You will only be asked to remove elements that are already in the list.

Grading

90 points - Implementation

isFull, isEmpty and toString are worth 5 points each.

The recursive methods are worth 15 points each.

10 points

Your name: Add your name to the top of the .java file after the Javadoc annotation @author

Package: Do not put your code in a package. This makes it difficult for the TAs to grade your submission.

Readability: Make sure your code is indented and neatly commented. You can use cntl a, cntl i to have Eclipse automatically indent your code properly. Do not leave commented out code in your submission.

Compilation: If your code does not compile, you will receive a 0 on the lab. Once notified, you will have 3 days to fix the lab to receive partial credit, up to 75%.

Late Policy: There is a 3 point per late penalty. You do not need to ask for an extension. Late penalties are strictly enforced.