Lab 3

Objectives:
- Sign up for the lab
- How to implement an interface
- How to implement a singly linked list
- Submit one file: *IntegerSLL.java* at the end of the lab. Click the "Submit" button

1. INode.java
Note that INode.java is a singly linked node. It is the building block for a singly linked list.

2. Interface
An interface provides a way to specify methods and constants, but supplies no implementations. There are two reasons to use interface: one reason for different performances. For example, the IntegerList interface in assignment 2 and in today's lab is implemented in three classes: array based (assignment 2), singly linked list (lab 3) and doubly linked list (assignment 2). The performance of the addFront() method differs among them. Why?

In a2tester.java, find the line where an interface is created. We can reuse the tester to test the *IntegerSLL.java* by a minor modification. Find it out and see how convenient an interface it. You are not going to pass all the test cases since the "removeAll()" method is not going to be implemented in the lab.

The lab instructor is going to draw a lot of diagrams on the whiteboard to explain how a singly linked list works, how an INode object is manipulated.

Use Connex to submit one file: *IntegerSLL.java* at the end of the lab. Make sure you click the "Submit" button.

Extra notes:
The second reason to use an interface is that it enables you to specify some desired common behavior that may be useful over many different types of objects. For example, the java.lang.Comparable<Object> interface enables two objects to be compared. If we implement the interface in Student.java, two Students objects can be compared. If it is implemented in Song.java, or Patient.java classes, two Song or two Patient objects can be compared. Why is it important? It means that we can sort a collection of those objects. Though we are not going to implement the Comparable interface today, we can see the usefulness of interfaces.

Learn to use online the API for help. For example, find the Comparable interface.
http://docs.oracle.com/javase/7/docs/api/

The "javadoc" style comments are added to INode.java. Learn to more, go to:
http://www.oracle.com/technetwork/java/javase/documentation/index-137868.html#examples