

## CS 1316 – Homework 7 – Linked Lists

**Due: Friday October 23<sup>rd</sup>, 2009 before 6pm.**

**Out of 100 points**

This is a pair programming problem! You are expected to work with the person you have been paired with in class, and you are both responsible for submitting the exact same code to T-Square. You may collaborate with other students in this class. Collaboration means talking through problems, assisting with debugging, explaining a concept, etc. You should not exchange code or write code for others. For pair programming assignments, you and your partner should turn in identical assignments. Collaboration at a reasonable level will not result in substantially similar code (with other groups).

### Linked Lists

For this assignment, you will be completing several methods in the (currently incomplete) `OurLinkedList.java` class file. Specifically, you will fill in the implementation of:

- `getFirst()`
- `addFirst(item )`
- `contains( item )`
- `set( int n, item)`
- `removeNthNode( int n)`
- `addNode( int n, item)`

The specifications and behavior for each method in this assignment are located in the Javadoc comments inside the incomplete `OurLinkedList.java` class file. There are several completed methods in addition to the ones that you need to implement. It is your task to implement each of the missing methods.

While implementing the above methods, you may use existing methods in the provided `OurLinkedList` class and the related `Node<Type>` class, but you may not use any methods from `java.util` or any other Java libraries.

NOTE: The `Node.java` class is using Java Generics, so that your linked list can contain objects of any type. The `main()` method in the `OurLinkedList.java` file instantiates a

linked list that can hold Strings, but your code should work with any objects stored within the list.

There is a main method in the LinkedList.java file that you can use to test your code. As you work on each method, you can test it out by commenting out sections of the main method that use the methods you have not yet written.

After completing all the methods, your output should look like:

```
> java LinkedList
LinkedList is empty: true
The size is: 0
Linked list:

LinkedList is empty: false
The size is: 3
Linked list:
    Node: I
    Node: Love
    Node: CS 1316

LinkedList is empty: false
The size is: 3
Linked list:
    Node: We
    Node: Love
    Node: Weekends

LinkedList is empty: false
The size is: 5
Linked list:
    Node: Yes,
    Node: We
    Node: All
    Node: Love
    Node: Weekends

LinkedList is empty: false
The size is: 3
Linked list:
    Node: We
    Node: All
    Node: Love

false
The data in the first node is: We
```

## Grading Criteria

GetFirst() .....	10 pts
addFirst(item) .....	10 pts
contains( item ) .....	20 pts
set( int n, item) .....	10 pts
Checks for bad index .....	3 pts
removeNthNode( int n) .....	25 pts
Checks for bad index.....	7 pts
Calls removeFirst() when needed .....	5pts
addNode( int n, item) .....	25 pts
Checks for bad index .....	7 pts
Calls addFirst(data) when needed .....	5 pts