

## **CMSC 132 Quiz 2 Worksheet**

The second quiz for the course will be on Friday, Jun 15. The following list provides more information about the quiz:

- The quiz will be a written quiz (no computer).
- Closed book, closed notes quiz.
- Answers must be neat and legible. We recommend that you use pencil and eraser.

The following exercises cover the material to be included in this quiz. Solutions to these exercises will not be provided, but you are welcome to discuss your solutions with the TA or instructor during office hours. **We strongly recommend you do not use Eclipse to write the code associated with these exercises.** Try to answer the exercises in a piece of paper and then use Eclipse to verify your solutions. This approach will better prepare you for the quiz. **You cannot use any Java API class during the implementation of the methods below.**

### **Exercises**

Implement the methods below based on the following Java class definitions.

```
public class MyLinkedList<T> {  
  
    private class Node<E> {  
        private E data;  
        private Node<E> next;  
    }  
  
    private Node<T> head;  
}
```

1. Define a constructor for the **MyLinkedList** class that creates an empty list.
2. Define a method called **addFirst** that adds an element to the beginning of the list.
3. Define a method named **size** that returns the number of nodes in the list.
4. Implement a method named **removedLastNode** that removes the last node from the list.
5. Define a method named **find** that determines whether a particular element is part of the list. The method will return true if the element is found in the list and false otherwise. You can assume the appropriate **equals** method has been defined for data elements of the list.
6. Define a method called **insert** that has the following prototype:

```
public void insert(T element);
```

The method will insert the element after the middle element of list. We define the middle element of the list as the node with "index" value equal to (size of the list)/2.

7. Define a method called **delete** that has the following prototype:

```
public void delete(T element);
```

The method will delete the middle element of the list. We define the middle element of the list as the node with "index" value equal to (size of the list)/2.