**CS 311 Data Structures and Algorithms**

**Assignment 2**

This assignment is based on the linked list data structure from assignment 1. You must have a working We will implement the following two functions on input linked lists:

1. Assume two linked lists that represent Set A and Set B respectively. Implement the following function to calculate their union A U B and return the result as a new linked list. Note that a SET should not contain duplicate elements (e.g., integers), but its elements are not assumed to be sorted in the list. Add a function unionLinkedList ().

LinkedList unionLinkedList (  
const LinkedList& LA,   
 const LinkedList& LB);

Example:

LA = (3, 5, 8, 11)

LB = (2, 6, 8, 9, 22, 24)

Then unionLinkedList(LA, LB) = (3, 5, 8, 11, 2, 6, 9, 22, 24) // The list isn’t sorted.

The mergeLinkedList and unionLinkedList functions are declared as friends of the Linked List class, so that they may access the private members of the class. You may add additional functions to the LinkedList to facilitate the implementation of those functions.

2. Assume two input linked lists, LA and LB, whose elements are both in the non-descending order. Implement the following function to merge LA and LB into a new linked list (as the return value). The elements in the new list should still be in the non-descending order. (50 Points)

LinkedList mergeLinkedList (  
const LinkedList& LA,   
 const LinkedList& LB);

Example:

LA = (3, 5, 8, 11)

LB = (2, 6, 8, 9, 22, 24)

Then mergeLinkedList (LA, LB) = (2, 3, 5, 6, 8, 8, 9, 11, 22, 24) // same value may repeat