CS 311 Data Structures and Algorithms

Assignment 1

Due on Wednesday, February 9, 11:59PM.

1. Assume we use two linked lists that represent Set A and Set B respectively. Implement the following function to calculate A U B and return the result as a new linked list. Note that a SET should not contain duplicate elements (e.g., integers). (50 Points)

*Node * unionLL (Node * LA, Node * LB);*

2. There are two linked lists, LA and LB. Their 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)

*Node * mergeLL (Node * LA, Node * LB);*

Example:
LA = (3, 5, 8, 11)
LB = (2, 6, 8, 9, 22, 24)
unionLL(LA, LB) = (3, 5, 8, 11, 2, 6, 9, 22, 24) // The order of the elements may change.
mergeLL (LA, LB) = (2, 3, 5, 6, 8, 8, 9, 11, 22, 24)

Additional requirements:

1. The input linked lists, LA and LB, should not be changed in either of these two functions.
2. Create a *main* function and print out the numbers in the linked lists before and after executing each method above.
3. Include your code in a single cpp file and submit it to Cougar Course.
4. At the beginning of your cpp file, add the following comment:

```
// Name: XXX XXX
// Student ID: XXXXXX
// Email: XXXXXX
```

5. Assume *Node* is defined as follows:

```
struct Node{
    int num;
    Node * next;
};
```