CS5403 Data Structures and Algorithms

Assignment 2

Assigned date: 09/16/2015

Due date: 09/24/2015

Do not use a separate header file, application file and implementation file. Instead, put all your code in a single file called hw2.cpp and submit it via the Blackboard website.

Please include the return 0; statement in main() in this assignment and all future assignments.

Do not use global variables.
Do not overload operators.
Do not use friend functions.
Do not use STL.
Do not use the <vector> library.

Problem:

In this assignment you will implement a linked-list data structure designed to store integers. Each node of the linked-list should store one integer.

Hint: you can implement a node as a C struct containing the stored integer and the pointer to the next node.

Tasks:

Once you implement your linked-list data structure

1.Write a function to concatenate two linked lists. Given lists l1 = (2, 3, 1) and l2 = (4, 5), after return from concatenate(l1,l2) the list l1 should be changed to be l1 = (2, 3, 1, 4, 5).

2.Write a function to insert a number into a sorted linked list. Assume the list is sorted from smallest to largest value. After insertion, the list should still be sorted. Given the list l1 = (3, 17, 18, 27) and the value 20, on return l1 be the list (3, 17, 18, 20, 27).

3.Write a function to return the median value in a sorted linked list. If the length n of the list is odd, then the median is the the element at the (n+1)/2 th node. For example, given the list (1, 2, 2, 5, 7, 9, 11) as input, your function should return the value 5. If the length of the list is even, then the median is the mean of the n/2 and (n/2)+1 members. Thus, the median of the sorted list (2, 4, 8, 9) is (4+8)/2.

4.Write a function to reverse the nodes in a linked list. Your function should have time complexity O(n), where n is the length of the list. You should create no new nodes. You shouldn't overwrite any node values. You should do this by just changing the pointers.

5) Write a main function that interacts with the user to support the above functions. Your main function should behave as follows:

Sample input:

```
Please tell me what to do: concatenate(C), insert(I), median(M), reverse(R), end(E)?
C
Please input the list1:
2 3 1
Please input the list2:
4 5
The outcome is:
2 3 1 4 5

Please tell me what to do: concatenate(C), insert(I), median(M), reverse(R), end(E)?
I
Please input the list:
3 17 18 27
Please input the value:
20
The outcome is:
3 17 18 20 27

Please tell me what to do: concatenate(C), insert(I), median(M), reverse(R), end(E)?
M
Please input the list:
1 2 2 5 7 9 11
The median is 5
Please please tell me what to do: concatenate(C), insert(I), median(M), reverse(R),
end(E)?
R
Please input the list:
1 2 3 4 5
The outcome is:
5 4 3 2 1

Please please tell me what to do: concatenate(C), insert(I), median(M), reverse(R),
end(E)?
E
```