CS5403 Data Structures and Algorithms

Assignment 3

Assigned date: 09/25/2015

Due date: 10/07/2015

Do not use a separate header file, application file and implementation file. Instead, put all your code in a single file called hw3.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 STL.

Tasks:

1) Implement a Stack to store characters using a linked-list. Your stack should provide pop() and push() functions. You can use the linked-list that you implemented for the previous homework. You need to modify your linked-list so that it will store characters rather than integers this time. Or, optionally, you can use Templates of Lecture 7.

2) Using only one stack object that you implemented above, write a function, which inverts the order of characters of the words in a given sentence. Notice that the word ordering should not change. For instance if the input is:

   "doing homework is fun"

   then the output should be:

   "gniod krowemoh si nuf"

   Don't forget that you are allowed to use only 1 stack.

3) In this part, you will deal with a classic problem of implementing a queue with two stacks. First describe how this could be done. Basically explain step-by-step how you can support insert() and get_front() operations of a queue using only two stacks. Assuming you have already inserted N elements in the queue, what is the runtime complexities (in O(..) notation) of your insert()

and get_front() functios. You should submit a separate document file for this part.

4) Then implement a queue class using two stacks. You can utilize the stack class that you implemented in the previous assignment.

5) In your main function, test your queue implementation. Basically insert a sentence character by character into your queue. Then pull each character from your queue and print to verify that the ordering is preserved (i.e. FIFO property.)