CS 180: Data Structures, Fall 2012
Homework 1

Due *via email* by 11:59pm on Sunday, Sept. 23

For this program, you'll modify the SLinkedList.h and SLinkedList.tcc files that are posted on the schedule page; *all* of the problems are designed to be added to that class. In addition, I've posted testSLL.cpp, which is a (very simple) test file for the class; you are welcome to download and modify it as well for your test file. Please don't forget to add appropriate comments to the functions and to the main, as well.

1. Write the `operator=` function for the SLinkedList class. Be sure to make a deep copy and deallocate any memory that is no longer needed.

2. Write a function `size` which takes no input arguments and returns an integer which is the size of the linked list.

   Note: You may do this by counting the nodes explicitly, or by keeping a private variable to track the size. Note that if you keep a separate size variable, you will have to alter the other functions as well to update this value correctly.

3. Write a function `nextToLast` which takes no inputs and returns the value stored in the second to last element of the list. If the list has no second to last element, throw an appropriate error.

4. Write a function `maximum` that takes no input parameters and returns the maximum value stored in your linked list.

5. Finally, write a main function to test all of your functions. Please comment and output appropriately, so that by looking at your code and running your main, I can see exactly where and how you are testing each problem.

6. Extra credit: Write a *fast* algorithm for reversing a singly linked list, so that the order of the nodes is the opposite of the original list. (Hint: This should be an $O(n)$ time algorithm.)