PIC 10B SPRING 2013 HOMEWORK 8

Assignment

P12.5 Provide a linked list of integer by modifying the `Node`, `List`, and `Iterator` classes of Section 12.2 to hold integers instead of strings.

P12.6 Write a member function `List::reverse()` that reverses the nodes in a list.

P12.7 Write a member function `List::push_front()` that adds a value to the beginning of a list.

P12.8 Write a member function `List::swap(List& other)` that swaps the elements on this list and `other`. Your method should work in $O(1)$ time.

P12.9-10 Add a `size` field to the `List` class. Modify the `insert` and `erase` functions to update the `size` field so that it always contains the correct size. Write a member function `get_size()` that returns the number of elements in the list.

P12.1 Write a function `void downsize(List& names)` that removes every second value from a linked list.

P12.2 Write a function `maximum` that computes the largest element in a `List`.

P12.3 Write a function `sort` that sorts the elements of a linked list (without copying them into a vector).

P12.4 Write a function `merge` that merges two Lists into one, alternating elements from each list until the end of one of the lists has been reached, then appending the remaining elements of the other list. For example, merging the lists containing `1 7 12` and `8 3 11 2 2 1` should yield the list `1 8 7 3 12 11 2 2 1`.

(1) Overload the ++ operator for the `Iterator` class (both prefix and postfix) to perform the same function as the `next()` member function, but returns the Iterator by value in the postfix form and an Iterator by reference (&) in the prefix form (i.e., similar behavior as before with Polynomials and Rationals).

(2) Overload the $--$ operator for the `Iterator` class (both prefix and postfix) to perform the same function as the `previous()` member function, analogous to the ++ operator defined above.

(3) Overload the == operator for the `Iterator` class to perform the same function as the `equals(Iterator b)` member function.

Place your code in a source file labeled *hw8.cpp*. ***If your file is not named this exactly, your homework will not be collected***. As with all programs in this course, your code should contain useful comments. In particular, your name, the date, and a brief description of what the program does should appear at the top of your source file.

What to Turn in

Place in your Submit folder the source file *hw8.cpp* with exactly this name (all lowercase, no spaces). The files will be automatically collected on Friday 5/24/13 at 5:00pm.

Date: April 4, 2013.

Grading		
Correctness	No errors, input/output correct, output presented nicely	5 points
Linked List	Correctly implements all Linked list operations	10 points
Style	Variable names, comments, indentation	5 points
	TOTAL	20 points

Note on grading: There is an automatic 5 point penalty for any homework that does not compile.

In the code below, you may assume the user will always enter a valid index for insertion.

Note! You may receive a compiler error if you try to define the classes before main sequentially. This is because the Node class depends on the Iterator class which depends on the Node class. There is a simple solution the book uses, see page 485.

```
Please input a set of nonnegative numbers for a List (Enter -1 when you are finished):
3
6
5
3
3
9
7
6
12
54
-1
Your list is
(3,6,5,3,3,9,7,6,12,54)

Select an index for insertion (enter -1 when finished): 10
Select a value for insertion: 111
Select an index for insertion (enter -1 when finished): 5
Select a value for insertion: 123
Select an index for insertion (enter -1 when finished): 8
Select a value for insertion: 1
Select an index for insertion (enter -1 when finished): 0
Select a value for insertion: 18
Select an index for insertion (enter -1 when finished): -1

The augmented List is
(18,3,6,5,3,3,123,9,7,1,6,12,54,111)

The maximum of the List is: 123
The size of the List is: 14
When we remove every second value from this list we are left with
(18,6,3,123,7,6,54)
When we sort this downsized list we obtain
(3,6,6,7,18,54,123)
And this sorted list in reverse order is
(123,54,18,7,6,6,3)
If we merge this list with the list (2,3,5,7,11) we obtain
(123,2,54,3,18,5,7,7,6,11,6,3)
```