PIC 10B SPRING 2013 HOMEWORK 9

Assignment

(1) Provide a binary search tree of integers by modifying the `TreeNode` and `BinarySearchTree` classes of Sections 13.2 and 13.3 to hold integers instead of strings.

P13.6 Write a member function `int BinarySearchTree::smallest()` that returns the smallest element of a tree. (The book has this function as `const` but just ignore it if you run into difficulties.)

(2) Write a member function of the `int BinarySearchTree::largest()` that returns the largest element of a tree.

P13.11 In the `BinarySearchTree` class, modify the `erase` member function so that a node with two children is replaced by the largest child of the left subtree.

∼P13.12 (With key differences!) Add a pointer to the parent node to the `TreeNode` class. Modify the `insert` and `erase` functions to properly set those parent nodes. Then define a `TreeIterator` class that contains a pointer to a `TreeNode`. The tree's begin member function returns an iterator that points to the root node. The iterator's `get` member function simply returns the data value of the node to which it points.

(3) (!!) Overload the ++ operator for the `TreeIterator` class as follows: the prefix version updates the Iterator by pointing to the left child, and the postfix version updates the Iterator by pointing to the right child. Both versions should return an Iterator by value (similar to the postfix versions previously).

(3) Overload the −− operator for the `TreeIterator` class (both prefix and postfix) to update the Iterator to point to the parent node. Here the standard differences between postfix and prefix apply.

(4) Make a `string BinarySearchTree::BinaryCode(int k)` function that returns the binary code discussed in lecture. For example, if the element 6 is the root node, with right child 12, and 12 has left child 10, and 10 has left child 8, and 8 has right child 9, then the binary code for 9 is 1001. If the element is the root node, return the empty string. If it is not found then return the string "2".

Place your code in a source file labeled *hw9.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 *hw9.cpp* with exactly this name (all lowercase, no spaces). The files will be automatically collected on Friday 5/31/13 at 5:00pm.

Date: April 4, 2013.

Grading		
Correctness	No errors, input/output correct, output presented nicely	5 points
Tree Structure	Correctly implements all Tree 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 non-duplicate positive integer for insertion.

```
Please input a set of distinct nonnegative numbers for a Tree (Enter -1 when you are finished):
7
5
6
3
12
9
54
-1

The maximum of the entries is: 54
The minimum of the entries is: 3
The size of the Tree is: 7

Select a value for insertion (enter -1 when finished): 111
Select a value for insertion (enter -1 when finished): 123
Select a value for insertion (enter -1 when finished): 1
Select a value for insertion (enter -1 when finished): 18
Select a value for insertion (enter -1 when finished): -1

The maximum of the entries is: 123
The minimum of the entries is: 1
The size of the Tree is: 11

The binary code for 1 is: 000
The binary code for 2 is: 2
The binary code for 3 is: 00
The binary code for 4 is: 2
The binary code for 5 is: 0
The binary code for 6 is: 01
The binary code for 7 is:
The binary code for 8 is: 2
The binary code for 9 is 10

Select a value to erase (enter -1 when finished): 5
Select a value to erase (enter -1 when finished): 7
Select a value to erase (enter -1 when finished): -1

The binary code for 1 is: 00
The binary code for 2 is: 2
The binary code for 3 is: 0
The binary code for 4 is: 2
The binary code for 5 is: 2
The binary code for 6 is:
The binary code for 7 is: 2
The binary code for 8 is: 2
The binary code for 9 is 10
```