CMSC 341 Data Structures

Binary Search Tree Review

These questions will help test your understanding of the binary search tree material presented in class and in the text. These questions are only a study guide. Questions found here may be found on your exam, although perhaps in a different format. Questions NOT found here may also be on your exam. The necessary class definitions will be provided with your exam. Please refer to the BST and BST node class definitions in the text.

1. Define ***binary search tree***.

2. Write the syntactically correct Java code for the private BST member function `find(T x)` whose function prototype is found below and cannot be changed. This private function is initially called by the public `find(T x)` with a reference to the root. If x is in the tree, return element in the node; otherwise return NULL.
   ```
   private T find(BinaryNode<T> root, T x)
   ```

3. Describe how deletion is performed in a BST. Be sure to mention all relevant cases.

4. The number of comparisons to build a BST of n elements is $O(n^2)$ in the worst case and $O(n \lg n)$ in the best case. Describe the best and worst case and explain why this is so.

5. Insert the following Java reserved words into an initially empty BST in the order given: *break, operator, if, typedef, else, case, while, do, return, byte, for, true, double, void.*

   a. Show the output from the pre-order, post-order, in-order, and level-order traversals of the BST.
   b. Which word is the predecessor of *while*?
   c. Which word is the successor of *for*?
   d. Draw the new tree that results from each of the following operations. Each problem is separate and begins with the tree created above.
      i. Insert *int* and *float*, in that order.
      ii. Delete *double*, *case*, and *typedef*, in that order.

6. Prove that if a node in a BST has two children, its successor has at most one child.

7. Suppose the result of a post-order traversal of a BST is 1, 3, 2, 8, 10, 5, 4. Draw the original BST.