CS2134 Homework 8 Spring 2016

Today? 2016

Written Part:



1)

2a) Recursive function  
 Check to make sure node isn’t a nullptr  
 if X < node, recursively search the left pointer  
 if X > node, recursively search the right pointer  
 otherwise, x == node so return true

2b) Recursive function  
 Check to make sure node isn’t a nullptr  
 if the min<node<max print and check left and right  
 if min== node or max == node print  
 if the node>=max check left   
 if the min>=node check right  
 2c) use a recursive function to create an ordered queue  
 Recursive function  
 if the left != nullptr, recursively call left  
 push the node onto the queue  
 if the right!= nullptr, recursively call right  
 set the first queue as the root  
 set n+1 as the right value of n for all of the queue and null as left

2d) use a recursive function to find the depth of each node and add them   
 together  
 use a recursive function to count how many nodes there are  
 divide the sum of all the depths by the amount of nodes.

2)

a) O(h)  
b)O(k+h)  
c)O(n)  
d)O(n)

3)assuming the use of NULL is valid in this case,this code should work properly. It checks through all the left nodes until the next left node is nullptr. The difference is that it checks if the current or original node is a nullptr.

4)this code might perform properly, but it sets the size of the new node created to 1, even though it’s already being incremented later on. Also, it does nothing if there’s a duplicate which is fine, but it’d be better if it threw an exception or at least output a warning.

So t = new Node{ x, nullptr, nullptr, 1 };   
Should be  
t = new Node{ x, nullptr, nullptr, 0 };

and in the else case there should be   
throw DuplicateItemException( );

5)

a)14,17,18,22,30

b)22,17,14,18,30

c) 14,18,17, 30, 22