Binary Tree Assignment

Marks: 25

Modify the BTree class developed in class to implement the following methods. The methods should be as easy to use as possible, for example you should not need to pass in the root of the tree to use a display method. As with the Linked Lists you will need three classes again and I only want to see the BTree class.

1. **depth**

   takes an integer and returns the depth that the integer was found at. (2 marks)

2. **display**

   prints all elements in the tree. This method must be overloaded to take either no parameters or a single integer. The integer must be defined as a constant in the BinaryTree class. These constants must be IN, PRE and POST and control if the tree is to be displayed in-order, pre-order or post-order. If no parameter is specified then display the tree in-order. (2 marks)

3. **countLeaves**

   returns the number leaves (have no children) in the tree. (2 marks)

4. **height**

   returns the maximum depth of the leaves in the tree. (3 marks)

5. **isAncestor**

   takes two integers and tells if the first integer is an ancestor of the second integer. (4 marks)

6. **delete**

   takes either a number or a TreeNode reference as the parameter and removes the node from the tree. (4 marks)

7. **isBalanced**

   returns true if the tree is balanced, and false otherwise. (4 marks)

8. **add**

   overload add so that it takes a BinTree. Add copies of all of the nodes of the BinTree parameter to the current tree. (4 marks)