Trees 1 - Binary Search Trees

Lab Description : Write a binary search tree class. For the base lab, you must write the following methods :: **preOrder, postOrder, inOrder, revOrder, getNumLeaves, getNumLevels, getWidth, getHeight, getNumNodes, toString, isFull.**

In addition, you may work with partners to complete the following tasks:

Task 1 :: write a method to search the tree for a value and return true or false

Task 2 :: write getLargest and getSmallest methods to return the smallest and largest tree values

Task 3 :: write a level order traversal using the Java LinkedList as a Queue

Task 4 :: write a method to remove a node from the tree – (best if recursive, will accept other methods)

Task 5 :: write a method to display the tree like a tree – level order traversal might be useful

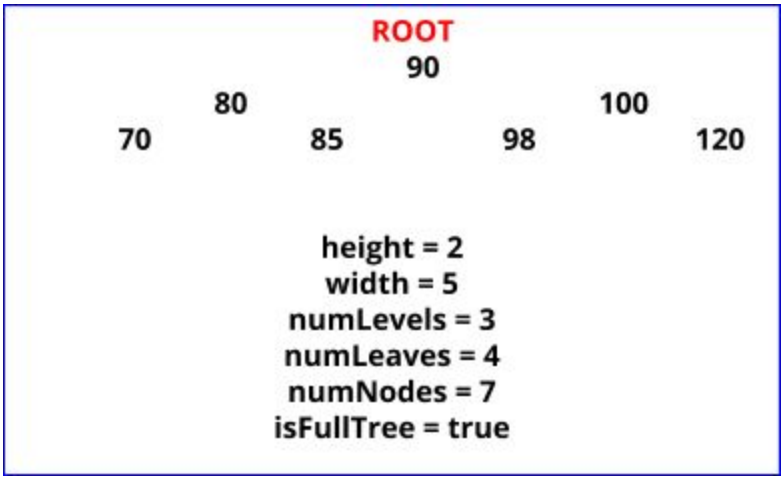
Sample Output :

```
IN ORDER
70 80 85 90 98 100 120


PRE ORDER
90 80 70 85 100 98 120


POST ORDER
70 85 80 98 120 100 90


REVERSE ORDER
120 100 98 90 85 80 70
```




```
Tree height is 2
Tree width is 5
Number of leaves is 4
Number of nodes is 7
Number of levels is 3
Tree as a string   70  80  85  90  98  100  120
The tree is full.
```

PARTNER TASK SECTION

```
The tree contains 100!

The does not contain 114!

The smallest tree node 70

The largest tree node 120



Tree before removing any nodes - using level order traversal.
90 80 100 70 85 98 120



Tree after removing 90.
98 80 100 70 85 120



Tree after removing 70.
98 80 100 85 120



Tree after removing 85.
98 80 100 120



Tree after removing 98.
100 80 120



Tree after removing 80.
100 120



Tree after removing 120.
100



Tree after removing 100.
```