Given a binary search tree (BST) with duplicates, find all the [mode(s)](https://en.wikipedia.org/wiki/Mode_(statistics)) (the most frequently occurred element) in the given BST.

Assume a BST is defined as follows:

* The left subtree of a node contains only nodes with keys **less than or equal to** the node's key.
* The right subtree of a node contains only nodes with keys **greater than or equal to** the node's key.
* Both the left and right subtrees must also be binary search trees.

For example:  
Given BST [1,null,2,2],

1

\

2

/

2

return [2].

**Note:** If a tree has more than one mode, you can return them in any order.

**Follow up:** Could you do that without using any extra space? (Assume that the implicit stack space incurred due to recursion does not count).