Smart Coding & Interview Series
Top-20 Basic Program
(BST Problems)

First, understand the solution building strategies and coding for the problems in LIVE/VIDEO session and then you apply those strategies discussed in LIVE/VIDEO session to solve the following problems. Use your favourite language(C/C++/Java/C#/Python/Scala) for coding.

1) Validate Binary Search Tree: Given a binary tree, determine if it is a valid binary search tree (BST). Assume a BST is defined as follows:

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

Example:

```
  2
 / \
1   3
```

Input: *[2,1,3]*

Output: *true*

Source : *https://leetcode.com/problems/validate-binary-search-tree/description/*

2) Balanced Binary Tree: Given a binary tree, determine if it is height-balanced. For this problem, a height-balanced binary tree is defined as: a binary tree in which the left and right subtrees of *every* node differ in height by no more than 1.

Example:

Given the following tree [3,9,20,null,null,15,7]:

```
  3
 / \
9  20
  /  \
 15   7
```

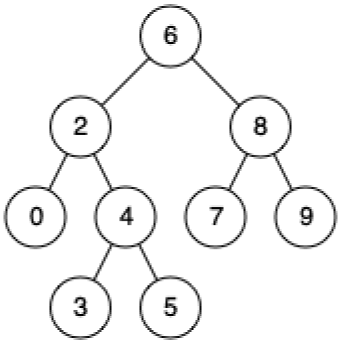
Return true.

Source: *https://leetcode.com/problems/balanced-binary-tree/description/*

3) Lowest Common Ancestor of a Binary Search Tree: Given a binary search tree (BST), find the lowest common ancestor (LCA) of two given nodes in the BST. The lowest common ancestor is defined between two nodes p and q as the lowest node in T that has both p and q as descendants (where we allow a node to be a descendant of itself).

Example:

Given binary search tree: root = [6,2,8,0,4,7,9,null,null,3,5]



Input: root = [6,2,8,0,4,7,9,null,null,3,5], p = 2, q = 8
Output: 6

Source:*https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-search-tree/description/*

4) Find Mode in Binary Search Tree

Given a binary search tree (BST) with duplicates, find all the mode(s) (the most frequently occurred element) in the given BST. If a tree has more than one mode, you can return them in any order. 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.

Example:

Given BST [1,null,2,2],

```
  1
   \
    2
   /
  2
```

return [2].

Source: https://leetcode.com/problems/find-mode-in-binary-search-tree/description/

5) Minimum Distance Between BST Nodes: Given a Binary Search Tree (BST) with the root node root, return the minimum difference between the values of any two different nodes in the tree.

Example :

Input: root = [4,2,6,1,3,null,null]
Output: 1

Source: *https://leetcode.com/problems/minimum-distance-between-bst-nodes/description/*