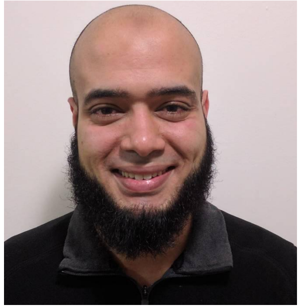
Data Structures

BST Homework 2

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

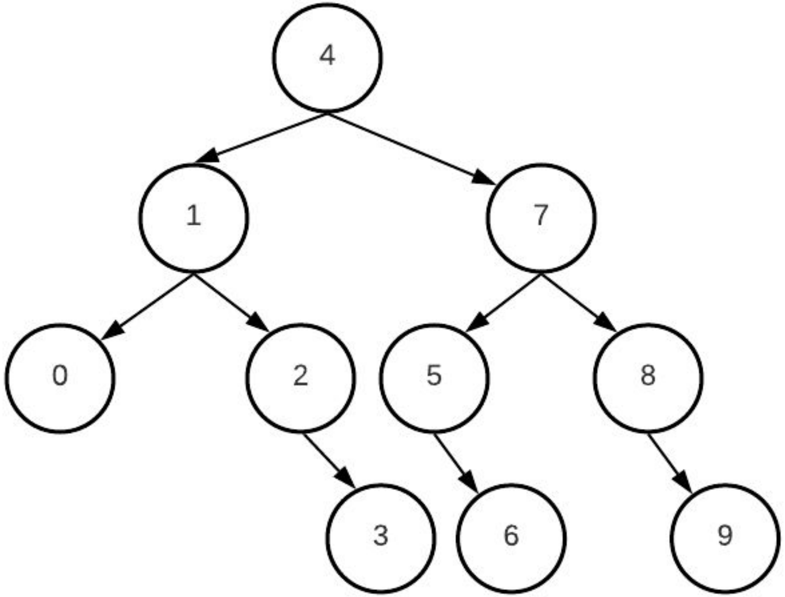
Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



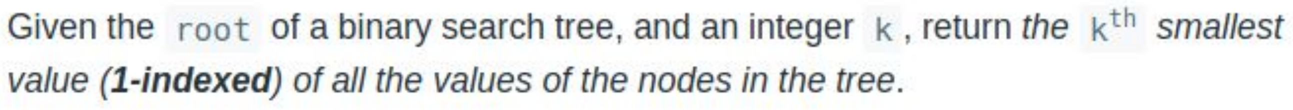
Problem #1: LeetCode 98 - Validate Binary Search Tree

- Given the root of a binary tree, determine if it is a valid binary search tree (BST).
    - Consider this: A binary tree with **duplicate** value is not BST
- Describe 2 ***fundamentally different*** approaches to check that
    - Important! Don't simply code recursive and iterative versions of code following identical logic.
    - I want two distinct implementations

- Example: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- On the right side, there is only one way to make it a balanced BST

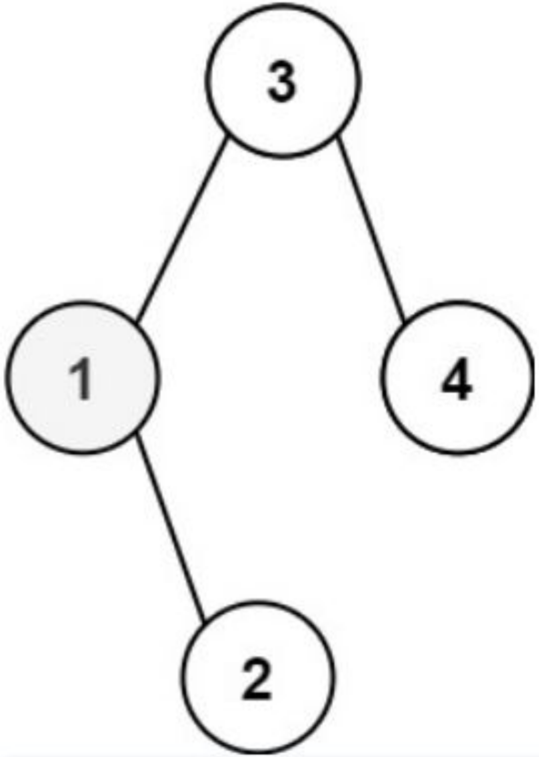


Problem #2: LeetCode 230 - Kth Smallest Element in a BST

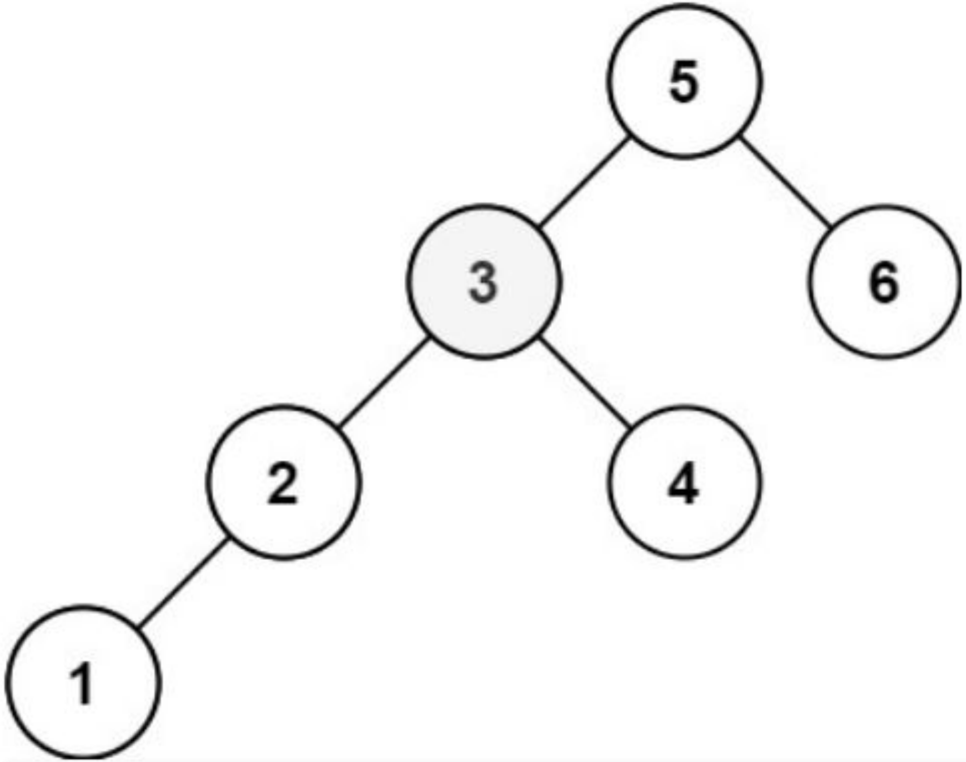


Given the `root` of a binary search tree, and an integer `k`, return *the* `k`th *smallest value (**1-indexed**) of all the values of the nodes in the tree.*

- A trivial way: compute inorder traversal, output is inorder[k-1]
  - Do something more efficient



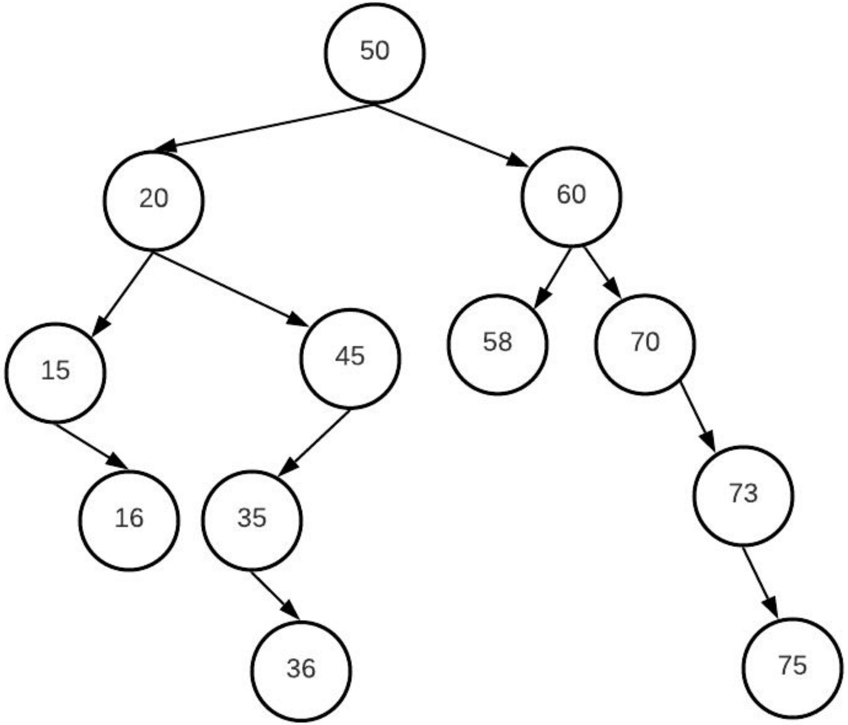
```
Input: root = [3,1,4,null,2], k = 1
Output: 1
```



```
Input: root = [5,3,6,2,4,null,null,1], k = 3
Output: 3
```

Problem #3: LeetCode 235 - Lowest Common Ancestor of a Binary Search Tree

- Given **2 nodes**, find their LCA
- LCA(x, y): the farthest node from the root that is an ancestor for both x and y.
  - The root is common ancestor for any pair,, but we want to find the farest from root
  - LCA(16, 45) = 35
  - LCA(45, 36) = 45
  - LCA(15, 70) = 50
  - LCA(58, 70) = 60
  - LCA(36, 75) = 50
  - LCA(70, 75) = 70



"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."