

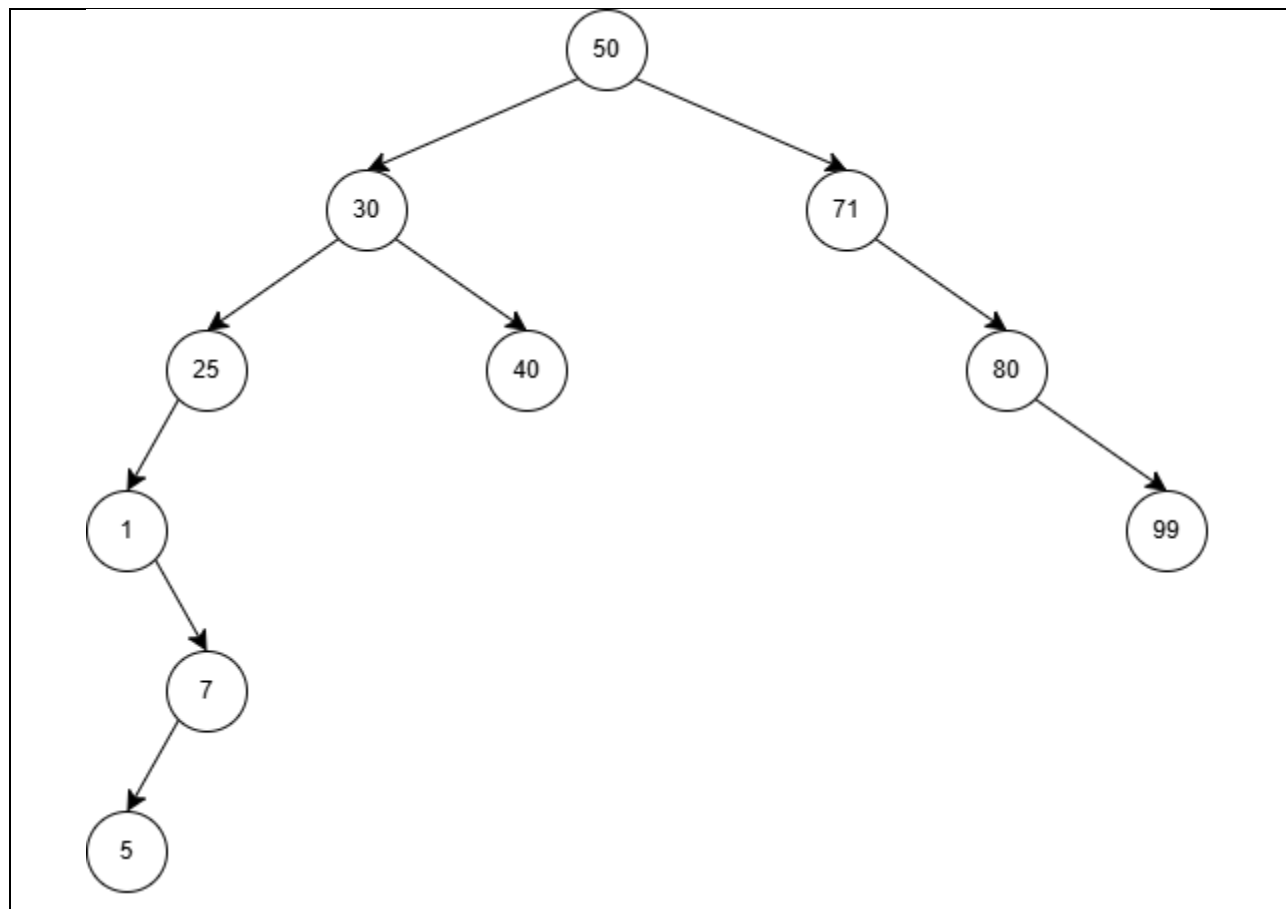
### Tutorial: Binary Search Tree

1. What is a binary search tree (BST)?

Binary search tree is a special type of binary tree with no duplicate elements where for every node in the tree, the value of any node in its left subtree is less than the value of the node and the value of any node in its right subtree is greater than the value of the node. In BST, each node can only have at most two children, enabling efficient search by eliminating half the remaining tree at each step which similar to binary search.

2. Build a BST based on the input 50, 30, 25, 71, 80, 99, 40, 1, 7, 5. Draw the final tree.

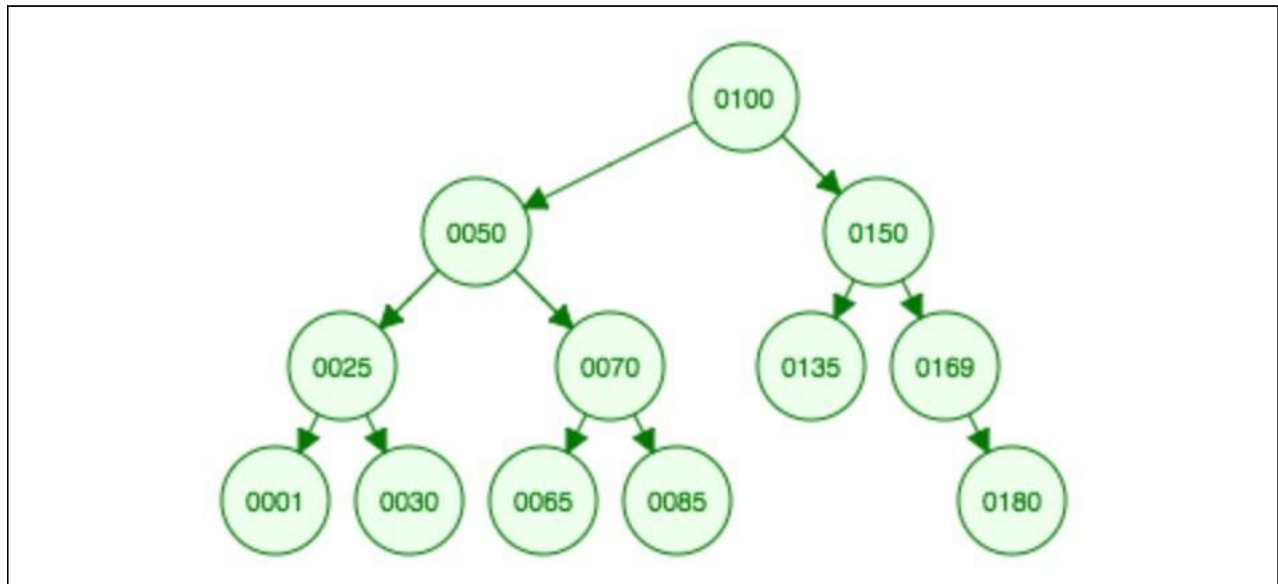
Rule: Left child < Parent < Right child



3. What is the height of the tree built in Question 2?

5 ( 50 → 30 → 25 → 1 → 7 → 5 )

4. Given the following BST, list the items in the order of:



(a) Pre-order traversal

(Depth-first search)

0100 → 0050 → 0025 → 0001 → 0030 → 0070 → 0065 → 0085 → 0150 → 0135 →  
0169 → 0180

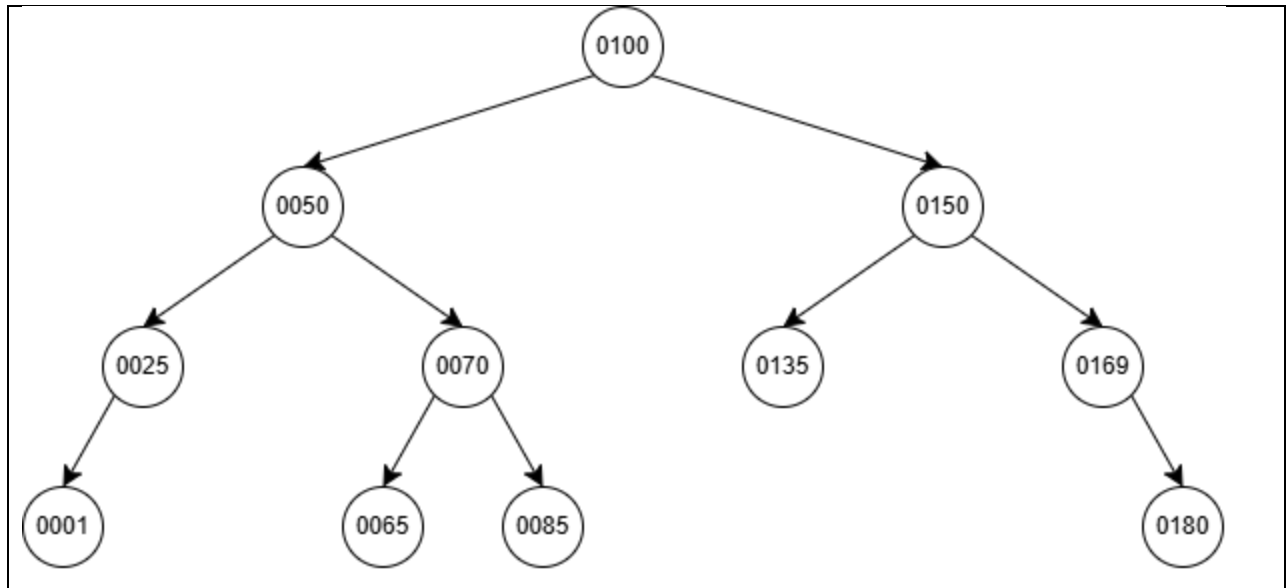
(b) In-Order traversal

0001 → 0025 → 0030 → 0050 → 0065 → 0070 → 0085 → 0100 → 0135 → 0150 →  
0169 → 0180

(c) Post-order traversal

0001 → 0030 → 0025 → 0065 → 0085 → 0070 → 0050 → 0135 → 0180 → 0169 →  
0150 → 0100

5. Using the same BST in Question 4, delete the element '0030'. Draw the resulting tree.



6. Again, using the same BST in Question 3 (i.e., ignoring the deletion of '0030' in Question 5), delete the element '0050'. Draw the resulting tree.

