Data Structures

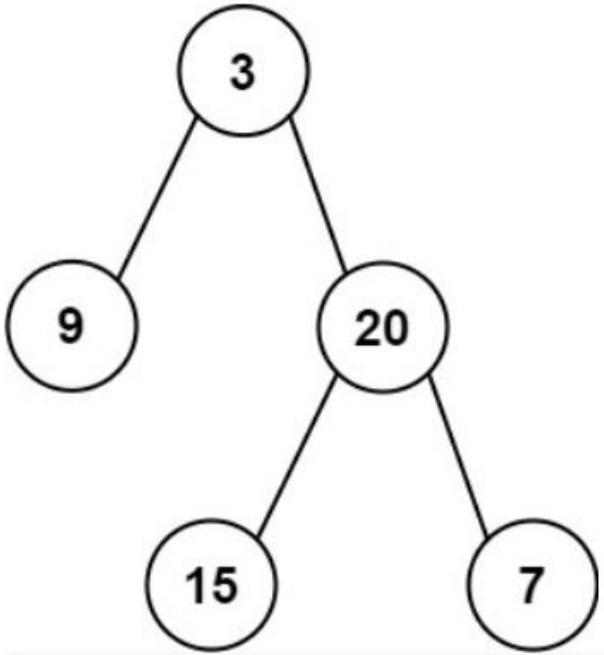
Binary Tree Homework 4

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 105 - Construct Binary Tree from Preorder and Inorder Traversal

- Given two **integer lists** preorder and inorder of **unique** values where preorder is the preorder traversal of a binary tree and inorder is the inorder traversal of the same tree, construct and return the binary tree.
- Use the lecture slides recursive approach
  - Feel free to keep the code simple, even if is very slow



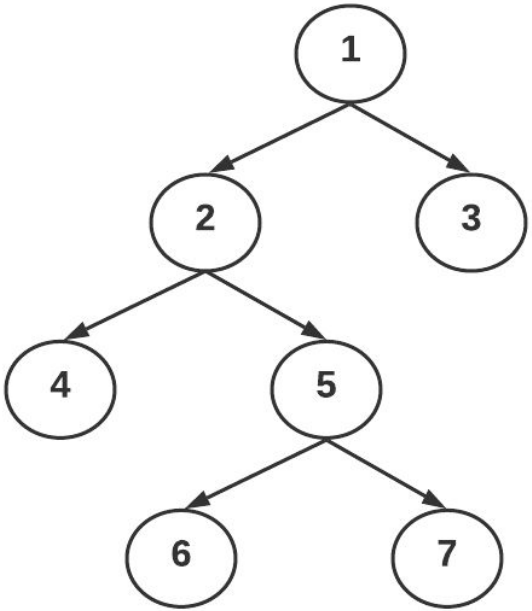

```
Input: preorder = [3,9,20,15,7], inorder = [9,3,15,20,7]
Output: [3,9,20,null,null,15,7]
```

Example 2:

```
Input: preorder = [-1], inorder = [-1]
Output: [-1]
```

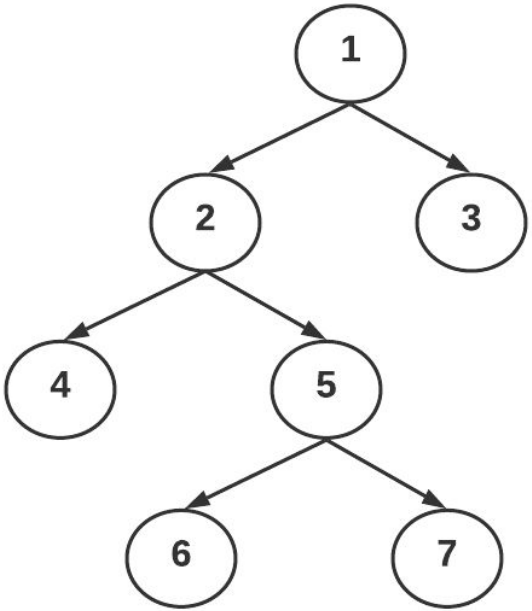
Problem #2: Generate a full binary tree

- **Sub-task**: Given a binary tree, implement
  def preorder_leaf(self)
- The function returns a **deque** of the tree preorder. However, it is actually deque of list: the inner list 2 is elements
  [preorder node value, is leaf]
  - Output for this tree is: [[1, False], [2, False], [4, **True**], [5, False], [6, **True**], [7, **True**], [3, **True**]]



Problem #2: Generate a full binary tree

- Implement: **def** create_from_preorder(self, preorder_dq)
- It takes the preorder in the previous format
  - Dequeue of lists each [value, is_leaf]
- The input will be from a **full** binary tree
- **Return** a new **binary tree** structure using this data



"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."