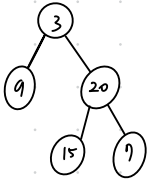


## Binary Tree Level Order Traversal (BFS)

: Given the root of a binary tree, return the level order traversal of its nodes' values.

Ex)



Input: root = [3, 9, 20, null, null, 15, 7]

Output: [[3], [9, 20], [15, 7]]

Algorithm : BFS

\* Key concept: 'Level Order' Traversal

# base case: reaching new level

if len(out) == level:

out.append([])

# Store values on each level

out[level].append(node.val)

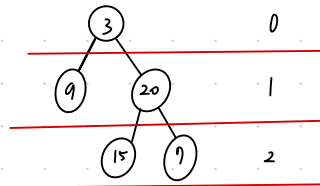
# traverse left to right

if node.left:

traverse(node.left, level+1)

if node.right:

traverse(node.right, level+1)



traverse(3, 0)

→ out.append([]) → out[0].append(3) → traverse(9, 1), stack(traverse(20, 1))

→ traverse(9, 1) → out.append([]) → out[1].append(9)

→ traverse(20, 1) → out[1].append(20) → traverse(15, 2), stack(traverse(7, 2))

→ traverse(15, 2) → out.append([]) → out[2].append(15)

→ traverse(7, 2) → out[2].append(7)