

```

1  /*Given a binary tree, return the zigzag level order traversal of its nodes' values.
   (ie, from left to right, then right to left for the next level and alternate between).
2
3  ..... For example:
4  ..... Given binary tree [3,9,20,null,null,15,7],
5  ..... 3
6  ..... / \
7  ..... 9  20
8  ..... / \
9  ..... 15  7
10 ..... return its zigzag level order traversal as:
11 ..... [
12 ..... [3],
13 ..... [20,9],
14 ..... [15,7]
15 ..... ]*/
16
17
18 import java.util.ArrayList;
19 import java.util.LinkedList;
20 import java.util.List;
21 import java.util.Queue;
22
23 class TreeNode {
24     .... int val;
25     .... TreeNode left;
26     .... TreeNode right;
27
28     .... TreeNode(int x) {
29     ..... val = x;
30     .... }
31 }
32
33
34 class Solution {
35
36     .... public List<List<Integer>> zigzagLevelOrder(TreeNode root) {
37
38     ..... List<List<Integer>> result = new ArrayList<>();
39
40     ..... if (root == null) {
41     .....     return result;
42     ..... }
43
44     ..... Queue<TreeNode> queue = new LinkedList<>();
45     ..... queue.offer(root);
46
47     ..... while (!queue.isEmpty()) {
48
49     .....     LinkedList<Integer> currentLayer = new LinkedList<>();
50     .....     int size = queue.size();
51
52     .....     if (result.size() % 2 == 0) {
53     .....         for (int i = 0; i < size; i++) {
54     .....             .... TreeNode tempNode = queue.poll();
55     .....             .... currentLayer.addLast(tempNode.val);
56     .....             .... if (tempNode.left != null) {
57     .....                 ..... queue.offer(tempNode.left);
58     .....             }
59     .....             .... if (tempNode.right != null) {
60     .....                 ..... queue.offer(tempNode.right);
61     .....             }
62     .....         }
63     .....     } else {
64     .....         for (int i = 0; i < size; i++) {
65     .....             .... TreeNode tempNode = queue.poll();
66     .....             .... currentLayer.addFirst(tempNode.val);
67     .....             .... if (tempNode.left != null) {
68     .....                 ..... queue.offer(tempNode.left);
69     .....             }
70     .....             .... if (tempNode.right != null) {
71     .....                 ..... queue.offer(tempNode.right);
72     .....             }

```

```
73 .....}
74 .....}
75
76 .....result.add(currentLayer);
77 .....}
78
79 .....return result;
80 .....}
81 }
```