

09/05/24

leetcode - Binary Tree Zigzag Level Order Traversal

```
int** zigzagLevelOrder(struct TreeNode* root,
int* returnSize, int** returnColumnSize){
```

```
int** ans = malloc(2000 * sizeof(int));
*returnColumnSize = malloc(2000 * sizeof(int));
*returnSize = 0;
```

```
struct TreeNode* tmp[2000] = {0};
```

```
int top = -1, start = 0;
```

```
tmp[start + top] = root;
```

```
while (tmp[start]) {
```

```
int tmp_top = top;
```

```
ans[(xreturnSize)] = malloc((top - start + 1) * sizeof(int));
```

```
(*returnColumnSize)[(xreturnSize)] = (top - start + 1);
```

```
int idx = (xreturnSize) * 2 / (top - start + 1) - 1; 0;
```

```
int dir = (xreturnSize) * 2 % 2 - 1; 1;
```

```
while (start <= tmp_top) {
```

```
ans[(xreturnSize)][idx] = tmp[start] -> val;
```

```
if (tmp[start] -> left) {
```

```
tmp[start + top] = tmp[start] -> left;
```

```
if (tmp[start] -> right) {
```

```
tmp[start + top] = tmp[start] -> right;
```

```
start++;
```

```
idx++;
```

```
(*returnSize)++;
```

```
return ans;
```

Output

Case 1

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

O/P: [[3], [9,20], [15,7]]

Case 2

IP: root = [1]

O/P: [[1]]

Case 3

IP: root = null

O/P: []

9/5/24