09/05/24. Week-2

Leetcode 2: Binary Tree zigzag level order traversal.

```
int** zigzagLevelOrder(struct TreeNode* root,
  int* returnSize, int** returnColumnSizes)
{
int **ans = malloc(2000* sizeof(int*));
*returnColumnSizes = malloc (2000*sizeof(int));
*returnSize=0;
struct TreeNode *tmp[2000]={0};
int top=-1, start=0;
tmp[++top]=root;
while(tmp[start])
{ int tmp_top=top;
 ans[(*returnSize)]=malloc((top-start+1)* sizeof(int));
(*returnColumnSizes)[(*returnSize)] = (top-start+1);
int idx=(*returnSize)%2 ? (top-start+1)-1 : 0;
int step = (*returnSize)%2 ? -1:1;
 while (start<=tmp_top).
{
  ans[(*returnSize)][idx] =tmp[start]->val;
  if(tmp[start]->left)
       tmp[++top]=tmp[start]->left;
  if(tmp[start]->right)
       tmp[++top]=tmp[start]->right;
  start++;
  idx+=step;
  }
 (*returnSize)++;
 }
 return ans;
}
```

output

case1 :

input = [3, 9, 20, null, null, 15, 7]
output = [[3], [20, 9], [15, 7]]

case2 :

input = [1]
output = [[1]]

case 3 :

input = []
output [ ]

9/5/24