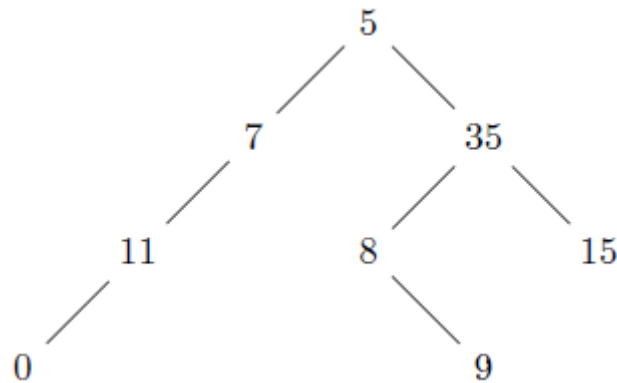


1. Ecrire une fonction qui prend en entrée un arbre binaire implémenté de manière dynamique et affiche les valeurs de l'arbre par niveau suivant l'ordre zigzag (c'est-à-dire de gauche à droite, puis de droite à gauche pour le niveau suivant et alterner entre eux).

Exemple :



La sortie du parcours par niveau en ordre zigzag de l'arbre ci-dessus est : 5 35 7 11 8 15 9 0

Astuce : utilisez deux piles : une pour les niveaux pairs et une pour les niveaux impairs.

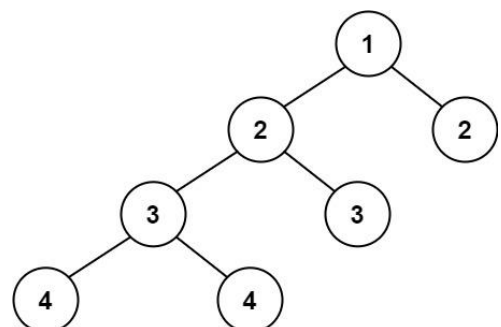
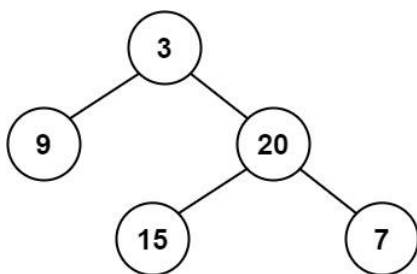
Write a function that takes as input a dynamically implemented binary tree and displays the values of the tree level by level following the zigzag order (i.e., from left to right, then from right to left for the next level, and alternating between them)

The output of the level order traversal in zigzag order for the above tree is: 5 35 7 11 8 15 9 0

Hint: Use two stacks, one for even levels and one for odd levels or a counter that keeps track of the level is odd or even.

2. Ecrire une fonction pour vérifier si un arbre binaire est équilibré ou non. Un arbre binaire est équilibré si les hauteurs des sous-arbres de gauche et de droite diffèrent d'au plus un pour chaque nœud.

Voici deux exemples (gauche → équilibré ; droite → non-équilibré) :

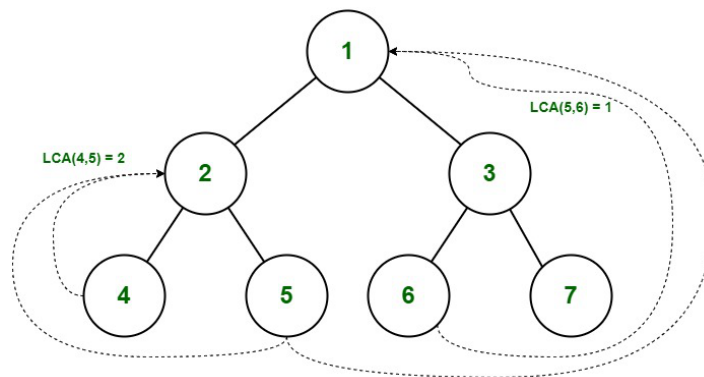


Write a function to check if a binary tree is balanced or no. A binary tree is balanced if the heights of the left and right subtrees differ by at most one for every node.

You find an example in figures above (left one is balanced and the right one is not)

3. Dans cet exercice, nous voulons implémenter le problème du plus petit ancêtre commun dans les arbres binaires. L'ancêtre commun le plus bas est le nœud le plus bas de l'arbre qui a à la fois $n1$ et $n2$ comme descendants, où $n1$ et $n2$ sont les nœuds pour lesquels nous souhaitons trouver le LCA. Par conséquent, le LCA d'un arbre binaire avec des nœuds $n1$ et $n2$ est l'ancêtre partagé de $n1$ et $n2$ qui est situé le plus loin de la racine.

Dans la figure ci-dessus, le LCA des nœuds 4 et 5 est 2. Le LCA des nœuds 6 et 5 est 1.



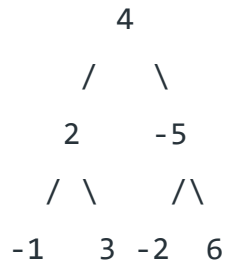
In this exercise, we want to implement the lowest common ancestor problem in binary trees. The **lowest common ancestor** is the lowest node in the tree that has both $n1$ and $n2$ as **descendants**, where $n1$ and $n2$ are the nodes for which we wish to find the LCA. Hence, the LCA of a binary tree with nodes $n1$ and $n2$ is the shared ancestor of $n1$ and $n2$ that is located farthest from the root.

In the figure above, the LCA of the nodes 4 and 5 is 2. The LCA of nodes 6 and 5 is 1.

4. Étant donné un arbre binaire ayant des valeurs positives et négatives, écrire une fonction qui imprime la somme des valeurs à chaque niveau et renvoie la somme maximale.

Given a Binary Tree having positive and negative values, write a function that prints the sum of values at each level and returns the highest one.

Input :



Output: 6

Explanation :

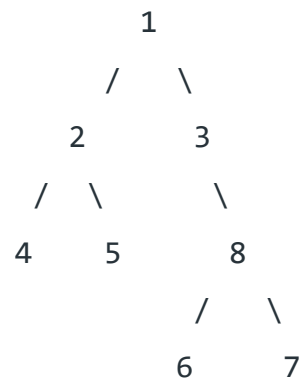
Sum of all nodes of 0'th level is 4

Sum of all nodes of 1'th level is -3

Sum of all nodes of 2'th level is 6

Hence maximum sum is 6

Input :



Output : 17