

## 606. Construct String from Binary Tree

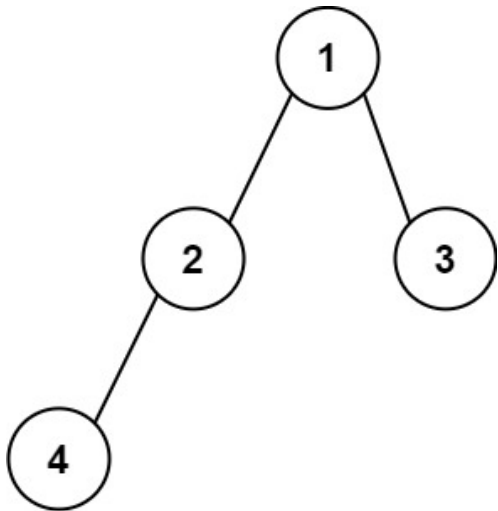
Medium  Topics  Companies

Given the `root` node of a binary tree, your task is to create a string representation of the tree following a specific set of formatting rules. The

- **Node Representation:** Each node in the tree should be represented by its integer value.
- **Parentheses for Children:** If a node has at least one child (either left or right), its children should be represented inside parentheses. Specifically:
  - If a node has a left child, the value of the left child should be enclosed in parentheses immediately following the node's value.
  - If a node has a right child, the value of the right child should also be enclosed in parentheses. The parentheses for the right child should follow those of the left child.
- **Omitting Empty Parentheses:** Any empty parentheses pairs (i.e., `()`) should be omitted from the final string representation of the tree, with the following exceptions: empty parentheses are not omitted when a node has a right child, nor when a node has only a right child.

In summary, empty parentheses pairs should be omitted when a node has only a left child or no children. However, when a node has a right child, an empty

Example 1:

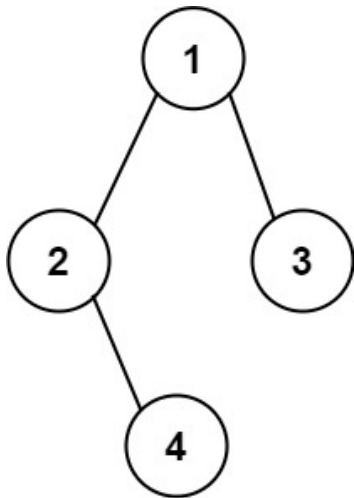


Input: `root = [1,2,3,4]`

Output: `"1(2(4))(3)"`

Explanation: Originally, it needs to be `"1(2(4))()(3())()"`, but you need to omit all the empty parentheses pairs to make it as short as possible.

Example 2:



Input: `root = [1,2,3,null,4]`

Output: `"1(2()(4))(3)"`

Explanation: Almost the same as the first example, except the `()` after 2 is necessary to indicate the absence of a left child of node 2.

Constraints:

- The number of nodes in the tree is in the range `[1, 104]`.