1490. Clone N-ary Tree Premium

Medium | Topics | Companies | Hint

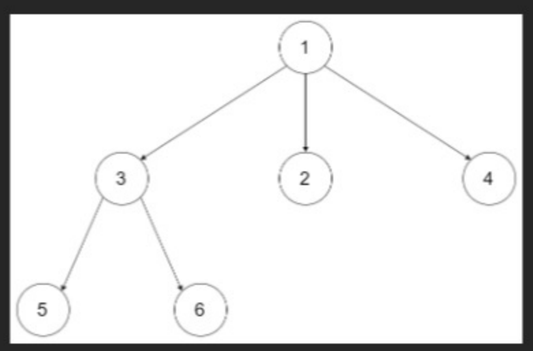
Given a `root` of an N-ary tree, return a **deep copy** (clone) of the tree.

Each node in the n-ary tree contains a val (`int`) and a list (`List[Node]`) of its children.

```
class Node {
    public int val;
    public List<Node> children;
}
```

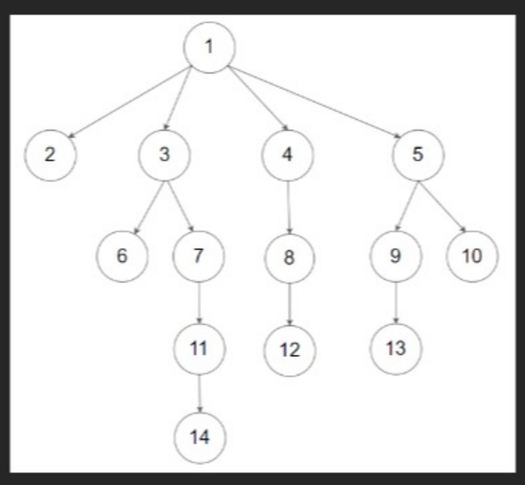
Nary-Tree input serialization is represented in their level order traversal, each group of children is separated by the null value (See examples).

Example 1:




```
Input: root = [1,null,3,2,4,null,5,6]
Output: [1,null,3,2,4,null,5,6]
```

Example 2:




```
Input: root = [1,null,2,3,4,5,null,null,6,7,null,8,null,9,10,null,null,11,null,12,null,13,null,null,14]
Output: [1,null,2,3,4,5,null,null,6,7,null,8,null,9,10,null,null,11,null,12,null,13,null,null,14]
```

Constraints:

- The depth of the n-ary tree is less than or equal to `1000`.
- The total number of nodes is between `[0, 10^4]`.

Follow up: Can your solution work for the graph problem?

Seen this question in a real interview before? 1/5

Yes No

Accepted 30.5K | Submissions 36.7K | Acceptance Rate 83.1%

Topics

Hash Table Tree Depth-First Search Breadth-First Search

Companies

0 - 6 months

Amazon 2

Hint 1

Traverse the tree, keep a hashtable with you and create a clone node for each node in the tree.

Hint 2

Start traversing the original tree again and connect each child pointer in the cloned tree the same way as the original tree with the help of the hashtable.

Similar Questions

Clone Graph	Medium
Copy List with Random Pointer	Medium
Clone Binary Tree With Random Pointer	Medium

Discussion (6)