< Back

Python O(n) recursive both functions

yorkshire ★ 684 Last Edit: October 20, 2018 6:27 AM 3.1K VIEWS

37

Serialize with preorder traversal where sentinel "#" indicates the final child of a node has been processed, so the function returns to its parent call.
Deserialize by creating a deque (could also use an iterator with next() instead of popleft()).
While the next item is not "#", create a child with the item, add the child to the list of children and recurse to create its subtree.
Repeat until there are no more children, then ignore the "#".

```
class Codec:
    def serialize(self, root):
        serial = []

        def preorder(node):

            if not node:
                return

            serial.append(str(node.val))

            for child in node.children:
                preorder(child)

            serial.append("#")      # indicates no more children, continue serialization from parent

        preorder(root)
        return " ".join(serial)

    def deserialize(self, data):
        if not data:
            return None

        tokens = deque(data.split())
        root = Node(int(tokens.popleft()), [])

        def helper(node):

            if not tokens:
                return

            while tokens[0] != "#": # add child nodes with subtrees
                value = tokens.popleft()
                child = Node(int(value), [])
                node.children.append(child)
                helper(child)

            tokens.popleft()        # discard the "#"

        helper(root)
        return root
```

Comments: 5 Best Most Votes Newest to Oldest Oldest to Newest

Type comment here... (Markdown is supported)

Post

amitrk23 ★ 4 July 7, 2019 6:30 AM

Amazing solution.
Easy to understand.

4 Reply

fzy1995 ★ 12 Last Edit: October 20, 2018 6:33 AM

Neat! Thank you for the solution!

1 Reply

charleszhou327 ★ 359 Last Edit: October 20, 2018 6:34 AM

Interesting solution .. thanks !