Problem List

Description | Accepted | Editorial | Solutions | Submissions

3294. Convert Doubly Linked List to Array

Solved

Premium

Medium | Topics | Hint

You are given an **arbitrary** `node` from a **doubly linked list**, which contains nodes that have a next pointer and a previous pointer.

Return an integer array which contains the elements of the linked list **in order**.

Example 1:

Input: head = [1,2,3,4,5], node = 5

Output: [1,2,3,4,5]

Example 2:

Input: head = [4,5,6,7,8], node = 8

Output: [4,5,6,7,8]

Constraints:

- The number of nodes in the given list is in the range `[1, 500]`.
- `1 <= Node.val <= 1000`
- All nodes have unique `Node.val`.

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

Yes | No

Accepted 2,893 /3.5K | Acceptance Rate 81.6%

Topics

Hint 1

Similar Questions

Discussion (8)

Copyright © 2025 LeetCode. All rights reserved.

9 | 8 | 0 Online

Submit

Code

Python3 • Auto

```python
class Node:
    def __init__(self, val, prev=None, next=None):
        self.val = val
        self.prev = prev
        self.next = next
"""
class Solution:
    def toArray(self, node: 'Optional[Node]') -> List[int]:

        res = []

        root = node

        while root:
            res.append(root.val)
            root = root.prev

        res = res[::-1]

        root = node
        root = root.next

        while root:
```

Saved

Ln 9, Col 16

Testcase | Test Result

Accepted Runtime: 39 ms

Case 1 | Case 2

Input

head =
[1,2,3,4,5]

node =
5

Output