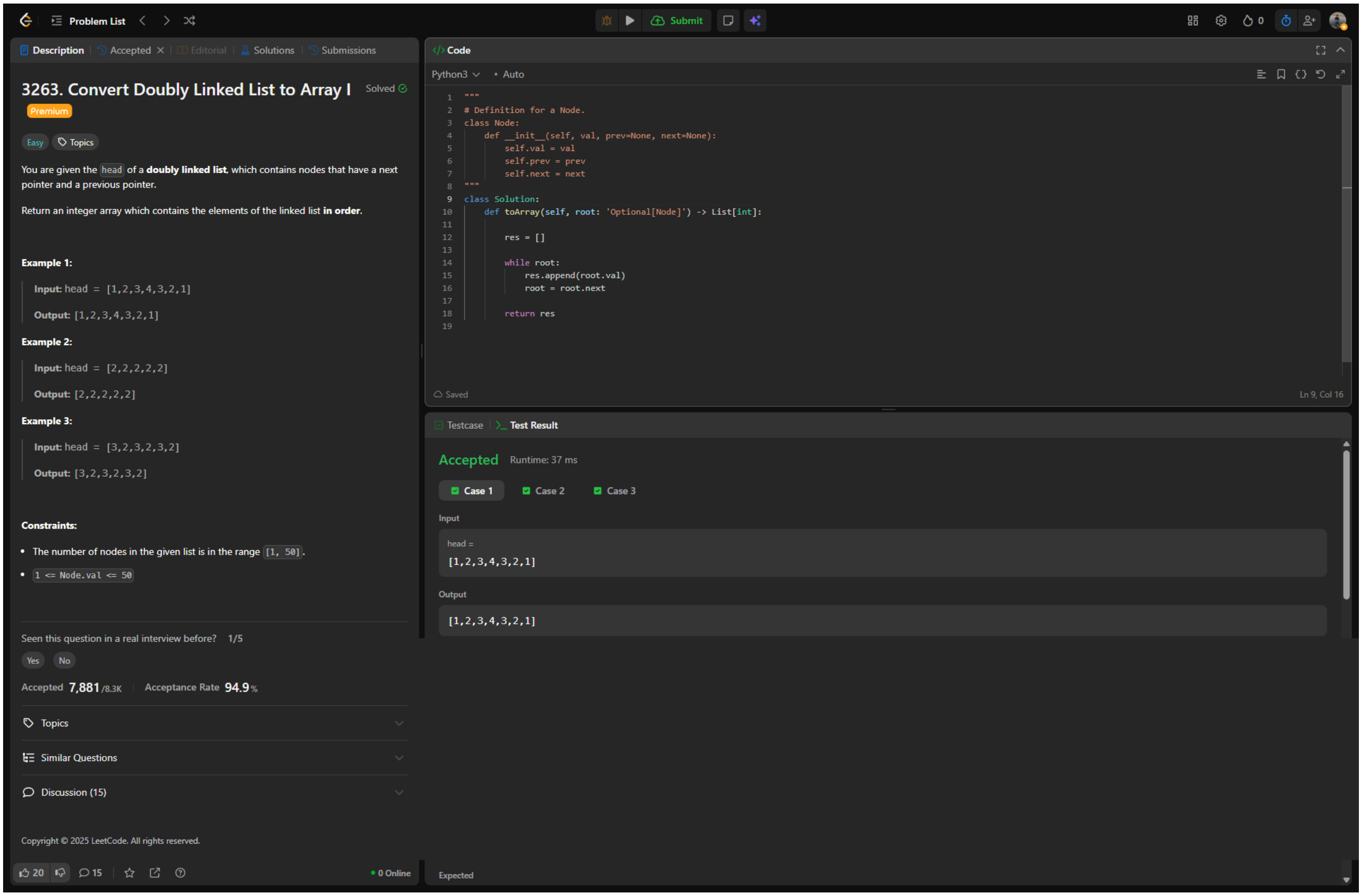
Problem List
Submit
Description | Accepted | Editorial | Solutions | Submissions
Code
Python3
• Auto

## 3263. Convert Doubly Linked List to Array I

Solved

You are given the `head` of 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 = `[1,2,3,4,3,2,1]`
>
> **Output:** `[1,2,3,4,3,2,1]`

**Example 2:**

> **Input:** head = `[2,2,2,2,2]`
>
> **Output:** `[2,2,2,2,2]`

**Example 3:**

> **Input:** head = `[3,2,3,2,3,2]`
>
> **Output:** `[3,2,3,2,3,2]`

**Constraints:**

- The number of nodes in the given list is in the range `[1, 50]`.
- `1 <= Node.val <= 50`

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

Accepted 7,881/8.3K | Acceptance Rate 94.9%

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

        res = []

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

        return res

```

Accepted Runtime: 37 ms

Input

head =

```
[1,2,3,4,3,2,1]
```

Output

```
[1,2,3,4,3,2,1]
```

Expected