



Description

Editorial

Solutions (10.4K)

Submission

i Java

Auto



## 876. Middle of the Linked List



Easy

9.9K

291

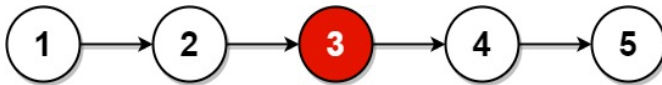


Companies

Given the `head` of a singly linked list, return *the middle node of the linked list*.

If there are two middle nodes, return **the second middle** node.

### Example 1:



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

**Output:** `[3,4,5]`

**Explanation:** The middle node of the list is node 3.

### Example 2:



**Input:** `head = [1,2,3,4,5,6]`

**Output:** `[4,5,6]`

**Explanation:** Since the list has two middle nodes with values 3 and 4, we return the second one.

### Constraints:

- The number of nodes in the list is in the range `[1, 100]`.

```
1 /**
2  * Definition for singly-linked list.
3  * public class ListNode {
4  *     int val;
5  *     ListNode next;
6  *     ListNode() {}
7  *     ListNode(int val) { this.val = val; }
8  *     ListNode(int val, ListNode next) { this.val = val; this.next = next; }
9  * }
10 */
11 class Solution {
12     public ListNode middleNode(ListNode head) {
13
14     }
15 }
```

Console



Run

Subr