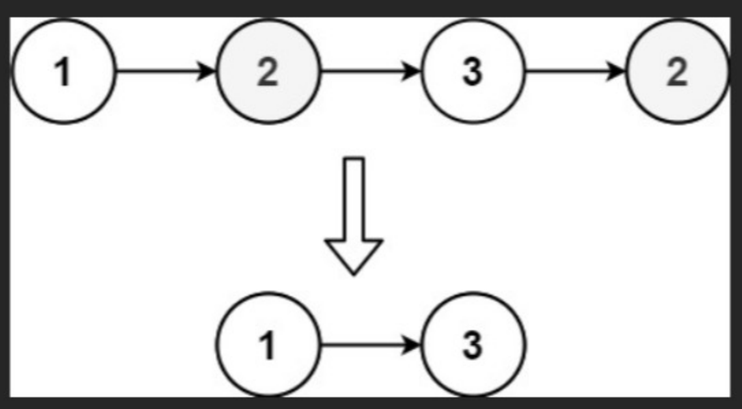
1836. Remove Duplicates From an Unsorted Linked List Premium

Medium | Topics | Companies | Hint

Given the `head` of a linked list, find all the values that appear **more than once** in the list and delete the nodes that have any of those values.

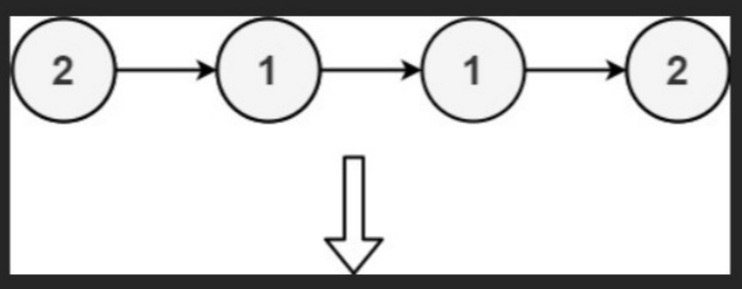
Return *the linked list after the deletions.*

Example 1:



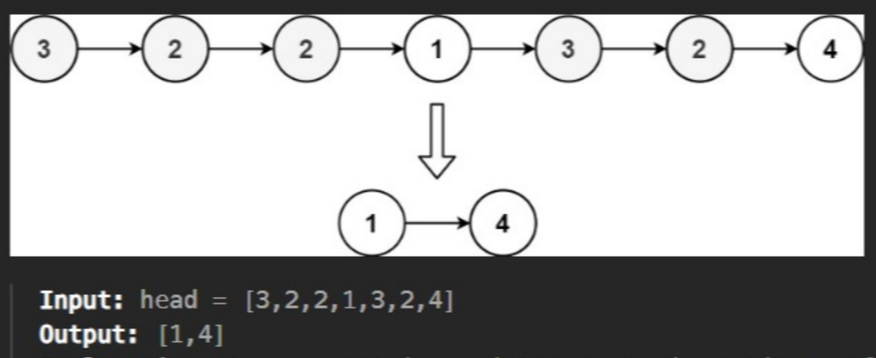

```
Input: head = [1,2,3,2]
Output: [1,3]
Explanation: 2 appears twice in the linked list, so all 2's should be deleted. After deleting all 2's, we are left with [1,3].
```

Example 2:




```
Input: head = [2,1,1,2]
Output: []
Explanation: 2 and 1 both appear twice. All the elements should be deleted.
```

Example 3:




```
Input: head = [3,2,2,1,3,2,4]
Output: [1,4]
Explanation: 3 appears twice and 2 appears three times. After deleting all 3's and 2's, we are left with [1,4].
```

Constraints:

- The number of nodes in the list is in the range `[1, 10^5]`
- `1 <= Node.val <= 10^5`

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

Yes No

Accepted 36K | Submissions 48.2K | Acceptance Rate 74.8%

Topics

Hash Table Linked List

Companies

0 - 6 months

Microsoft 2

Hint 1

Is there a way we can know beforehand which nodes to delete?

Hint 2

Count the number of appearances for each number.

Similar Questions

Remove Duplicates from Sorted List II	Medium
Remove Duplicates from Sorted List	Easy

Discussion (7)