3217. Delete Nodes From Linked List Present in Array

Medium

Topics

Companies

Hint

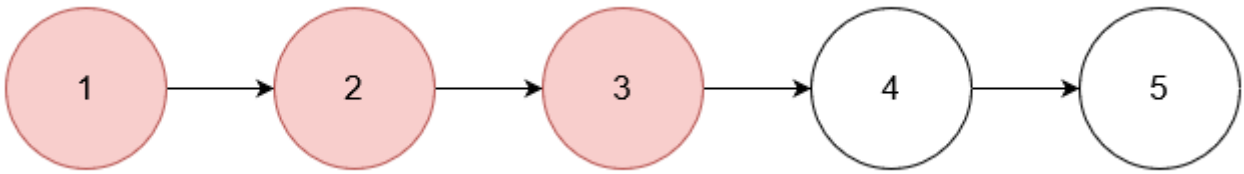
You are given an array of integers `nums` and the `head` of a linked list. Return the `head` of the modified linked list after **removing** all nodes from the linked list that have a value that exists in `nums`.

Example 1:

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

Output: [4,5]

Explanation:



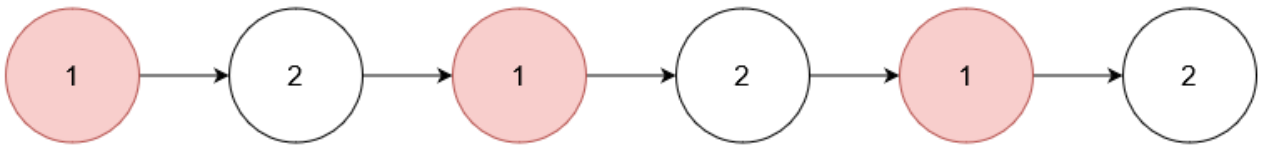
Remove the nodes with values 1, 2, and 3.

Example 2:

Input: nums = [1], head = [1,2,1,2,1,2]

Output: [2,2,2]

Explanation:



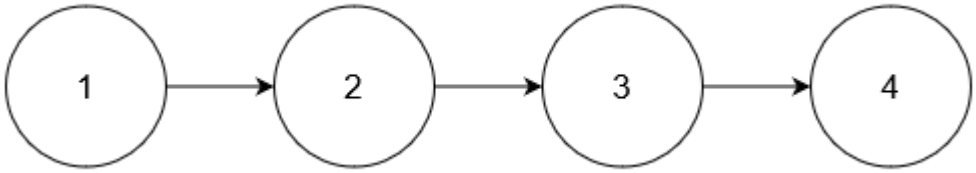
Remove the nodes with value 1.

Example 3:

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

Output: [1,2,3,4]

Explanation:



No node has value 5.

Constraints:

- `1 <= nums.length <= 105`
- `1 <= nums[i] <= 105`
- All elements in `nums` are unique.
- The number of nodes in the given list is in the range `[1, 105]`.
- `1 <= Node.val <= 105`
- The input is generated such that there is at least one node in the linked list that has a value not present in `nums`.

Solution:

```
/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     int val;
 *     ListNode next;
 *     ListNode() {}
 *     ListNode(int val) { this.val = val; }
 *     ListNode(int val, ListNode next) { this.val = val; this.next = next;
}
 * }
 */
class Solution {
```

```java
    public ListNode modifiedList(int[] nums, ListNode head) {
        Set<Integer> valuesSet = new HashSet<>();
        for (int value : nums) {
            valuesSet.add(value);
        }

        ListNode dummy = new ListNode(0);
        dummy.next = head;
        ListNode current = dummy;

        while(current != null && current.next != null){
            if(valuesSet.contains(current.next.val)){
                current.next = current.next.next;
            }else{
                current = current.next;
            }
        }
        return dummy.next;
    }
}
```