83. Remove Duplicates from Sorted List ★

Question | Editorial Solution

My Submissions (/problems/remove-duplicates-from-sorted-list/submissions/)

Total Accepted: **139027** Total Submissions: **365778** Difficulty: **Easy**

Notes

Given a sorted linked list, delete all duplicates such that each element appear only *once*.

For example,

Given `1->1->2`, return `1->2`.

Given `1->1->2->3->3`, return `1->2->3`.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Have you met this question in a real interview? Yes No

Discuss (https://leetcode.com/discuss/questions/oj/remove-duplicates-from-sorted-list)

Pick One (/problems/random-one-question/)

C++

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    ListNode* deleteDuplicates(ListNode* head) {
        if (!head) return head;
        ListNode dummyBegin(head->val-1); // 故意跟頭不一樣
        ListNode* curNode = &dummyBegin;

        while (head) {
            if (head->val != curNode->val) {
                curNode->next = head;
                curNode = curNode->next;
                head = head->next;
            }
            else {
                ListNode* prev = head;
                head = head->next;
                delete prev;
```

Send Feedback (mailto:admin@leetcode.com?subject=Feedback)

```
26              }
27          }
28          curNode->next = NULL;
29          return dummyBegin.next;
30      }
```

Custom Testcase

Run Code | Submit Solution

Notes