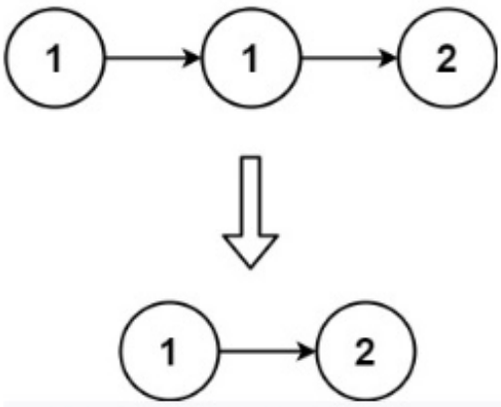
Remove duplicate from sorted list

83. Remove Duplicates from Sorted List

Easy 7529 252 Add to List Share

Given the `head` of a sorted linked list, *delete all duplicates such that each element appears only once.* Return *the linked list* ***sorted*** *as well.*

Example 1:



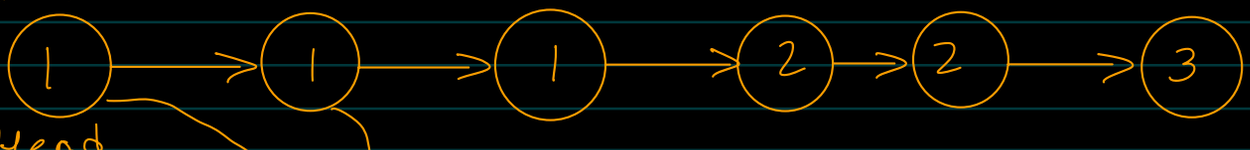

```
Input: head = [1,1,2]
Output: [1,2]
```

Understand the logic :-

The advantage we have here is Linked List (LL) is Sorted so duplicate number will in contiguous format

example

temp -> value temp -> next -> value



Head

temp = head

If temp -> value == temp -> next -> value

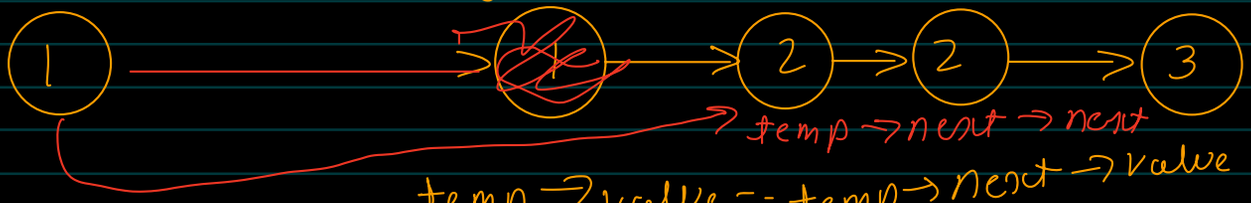
then we need to delete temp -> next node

temp -> next = temp -> next -> next;

Don't lose the head until we get different value

temp->value

temp->next->value



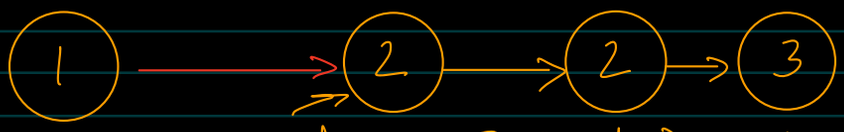
temp->next->next

temp->value == temp->next->value

delete(temp->next)

temp->next = temp->next->next

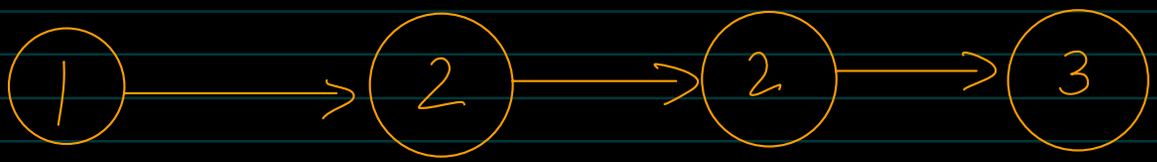
temp->value



temp->next->value

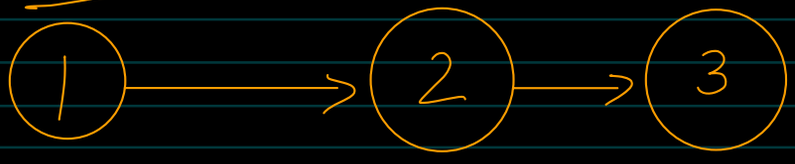
are not equal move temp

temp = temp->next



Is duplicate removed

Result



Code

```
class Solution {
public:
    ListNode* deleteDuplicates(ListNode* head) {
        ListNode* temp=head;
         while(temp!=NULL && temp->next!=NULL){
             if(temp->val==temp->next->val){
                 ListNode* DeleteNode=temp->next;
                 temp->next=temp->next->next;
                 delete(DeleteNode);
             }
             else{
                 temp=temp->next;
             }
         }
         return head;
    }
};
```

T.C = O(n)

S.C = O(1)