

47. Remove Duplicates from Sorted List

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.

Code:

```
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

def deleteDuplicates(head):
    current = head

    while current and current.next:
        if current.val == current.next.val:
            current.next = current.next.next
        else:
            current = current.next

    return head

def printList(node):
    while node:
        print(node.val, end=" -> ")
        node = node.next
    print("None")

head = ListNode(1)
head.next = ListNode(1)
head.next.next = ListNode(2)
head.next.next.next = ListNode(3)
head.next.next.next.next = ListNode(3)

print("Original list:")
printList(head)

head = deleteDuplicates(head)

print("List after removing duplicates:")
printList(head)
```

Output:

```
Original list:  
1 -> 1 -> 2 -> 3 -> 3 -> None  
List after removing duplicates:  
1 -> 2 -> 3 -> None
```

Time Complexity:

- $T(n) = O(n)$