All Contests > Assignment 01 | Basic Data Structures | Batch 04 > Remove Duplicate

# **Remove Duplicate**

Problem Submissions Leaderboard Discussions

#### **Problem Statement**

You will be given a singly linked list of integer values as input. You need to remove duplicate values from the linked list and finally print the linked list.

The process is, for each node N, traverse from that node and delete all nodes where the values are same with N.

Note: You must use singly linked list, otherwise you will not get marks.

### **Input Format**

• First line will contain the values of the singly linked list, and will terminate with -1.

#### Constraints

- 1. 1  $\leq$  N  $\leq$  1000; Here N is the maximum number of nodes of the linked list.
- 2.  $0 \le V \le 1000$ ; Here V is the value of each node.

#### **Output Format**

• Output the final linked list where there will be no duplicate values.

#### Sample Input 0

1 2 3 4 5 -1

## Sample Output 0

1 2 3 4 5

### Sample Input 1

1 2 4 2 3 5 1 4 5 2 6 1 -1

#### Sample Output 1

1 2 4 3 5 6

#### Sample Input 2

5 5 1 1 2 4 2 4 1 3 5 0 -1

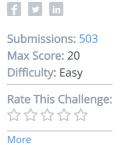
## Sample Output 2

5 1 2 4 3 0

# Sample Input 3

#### Sample Output 3

10 20



```
C++20
                                                                                                X | Ø
 1 #include <bits/stdc++.h>
   using namespace std;
 3
 4 ▼class Node{
 5
        public:
        int val;
 6
 7
        Node* next;
 8 1
        Node(int val){
           this->val = val;
 9
10
            this->next = NULL;
11
        }
12
   };
13
14 void print_linked_list(Node *head){
        Node* temp = head;
15
        while(temp != NULL){
16 🔻
17
            cout << temp->val << " ";</pre>
18
            temp = temp->next;
19
        }
20
   }
21
   void insert_tail(Node* &head, Node *&tail, int val){
23
        Node *newNode = new Node(val);
24
        if(head == NULL){
25
            head = newNode;
            tail = newNode;
26
27
            return;
        }
28
29
        tail->next = newNode;
30
        tail = newNode;
31
   }
32
33 void find_func(Node *head, int val){
        Node* temp2 = head;
34
35
        while(temp2->next != NULL){
36 ▼
37 '
            if(val == temp2->next->val){
38
                Node* delete_data = temp2->next;
39
                temp2->next = delete_data->next;
40
                delete delete_data;
41 1
            }else{
                temp2 = temp2->next;
42
43
44
45
        }
46
47
   }
48
49 vint main(){
50
```

```
51
        Node* head = NULL;
52
        Node* tail = NULL;
53
54
        int val;
55
        while (true)
56 🔻
57
            cin >> val;
            if(val == -1){
58 ▼
                break;
59
60
            insert_tail(head, tail, val);
61
        }
62
63
        Node *temp = head;
64
65 1
        while(temp->next != NULL){
66
            find_func(temp,temp->val);
67
            temp = temp->next;
68
        }
69
70
        print_linked_list(head);
71
        return 0;
72
   }
73
                                                                                              Line: 67 Col: 27
```

<u>♣ Upload Code as File</u> Test against custom input

Run Code

Submit Code