

Code

C Auto

```
1  /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     struct ListNode *next;
6  * };
7 */
8 struct ListNode* removeElements(struct ListNode* head, int val) {
9     // Handle nodes at the head with the target value
10    while (head != NULL && head->val == val) {
11        struct ListNode* temp = head;
12        head = head->next;
13        free(temp);
14    }
15
16    struct ListNode* curr = head;
17
18    // Traverse the list and remove nodes with the target value
19    while (curr != NULL && curr->next != NULL) {
20        if (curr->next->val == val) {
21            struct ListNode* temp = curr->next;
22            curr->next = curr->next->next;
23            free(temp);
24        } else {
25            curr = curr->next;
26        }
27    }
28
29    return head;
30 }
```

Saved Ln 5, Col 30

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

```
head =
[]
```

Val

```
1
```

Output

```
[]
```

Expected

```
[]
```

Contribute a testcase

Problem List < > ⌛

Testcase | Test Result

**Accepted** Runtime: 0 ms

Case 1  Case 2  Case 3

Input

```
head =  
[1,2,6,3,4,5,6]
```

val =  
6

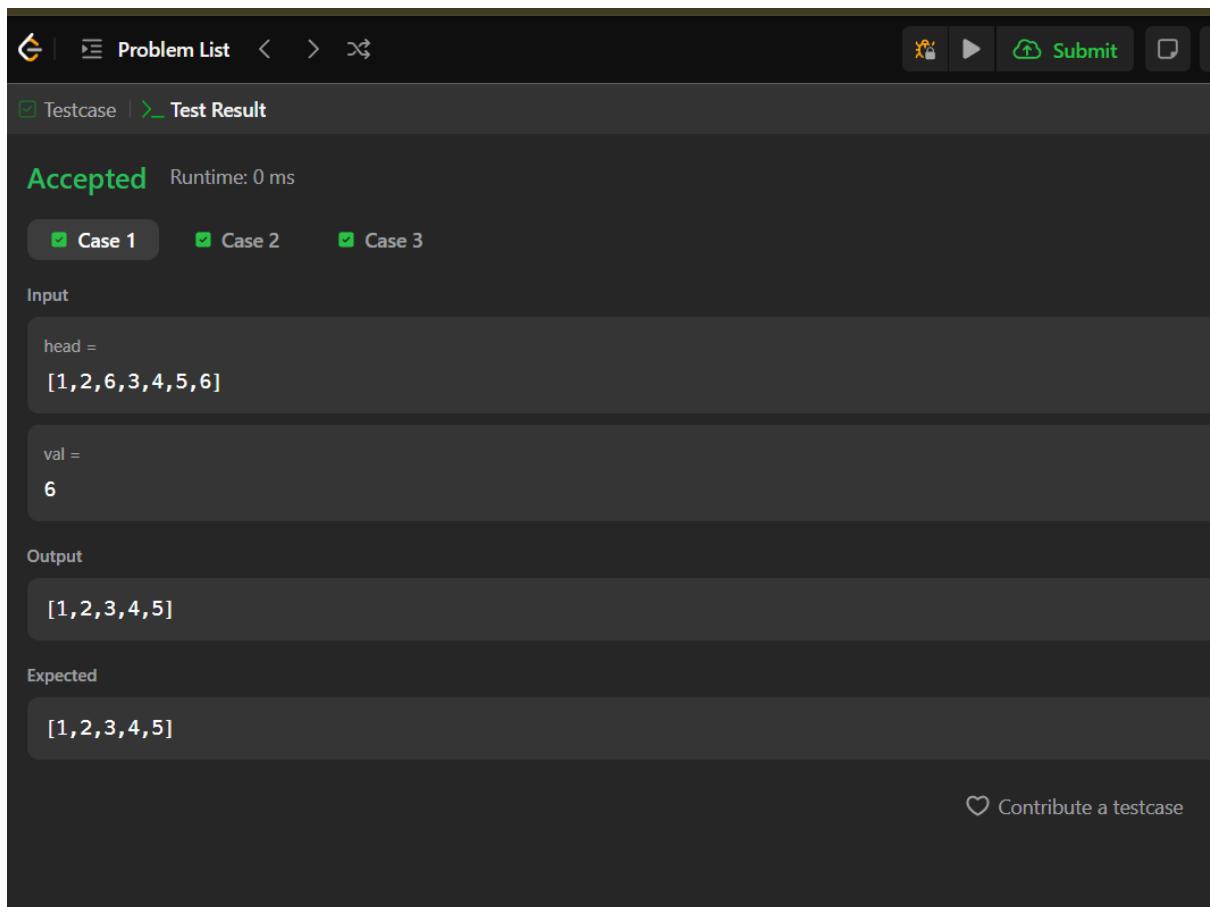
Output

```
[1,2,3,4,5]
```

Expected

```
[1,2,3,4,5]
```

Heart icon: Contribute a testcase



Problem List < > ⌛

Testcase | Test Result

**Accepted** Runtime: 0 ms

Case 1  Case 2  Case 3

Input

```
head =  
[7,7,7,7]
```

val =  
7

Output

```
[]
```

Expected

```
[]
```

