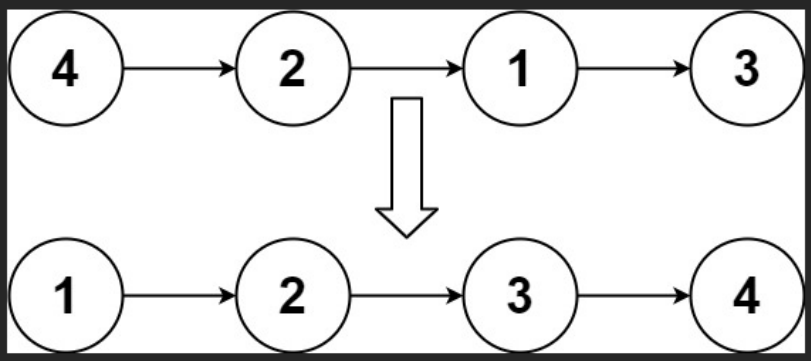
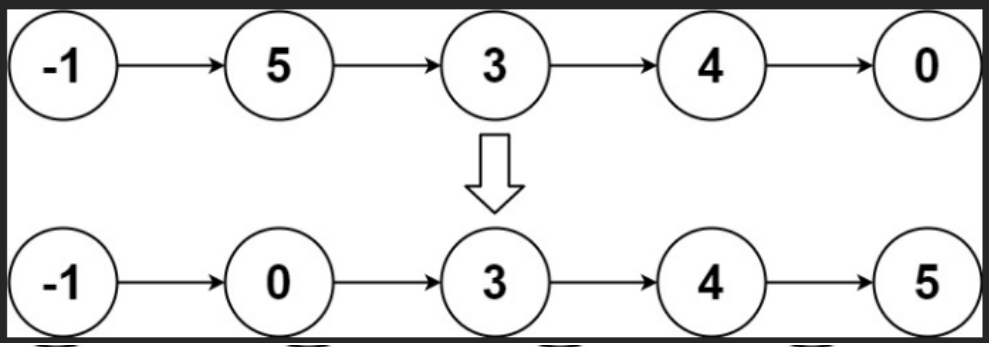
Given the `head` of a linked list, return *the list after sorting it in* ***ascending order***.

Example 1:




```
Input: head = [4,2,1,3]
Output: [1,2,3,4]
```

Example 2:




```
Input: head = [-1,5,3,4,0]
Output: [-1,0,3,4,5]
```

Example 3:

```
Input: head = []
Output: []
```

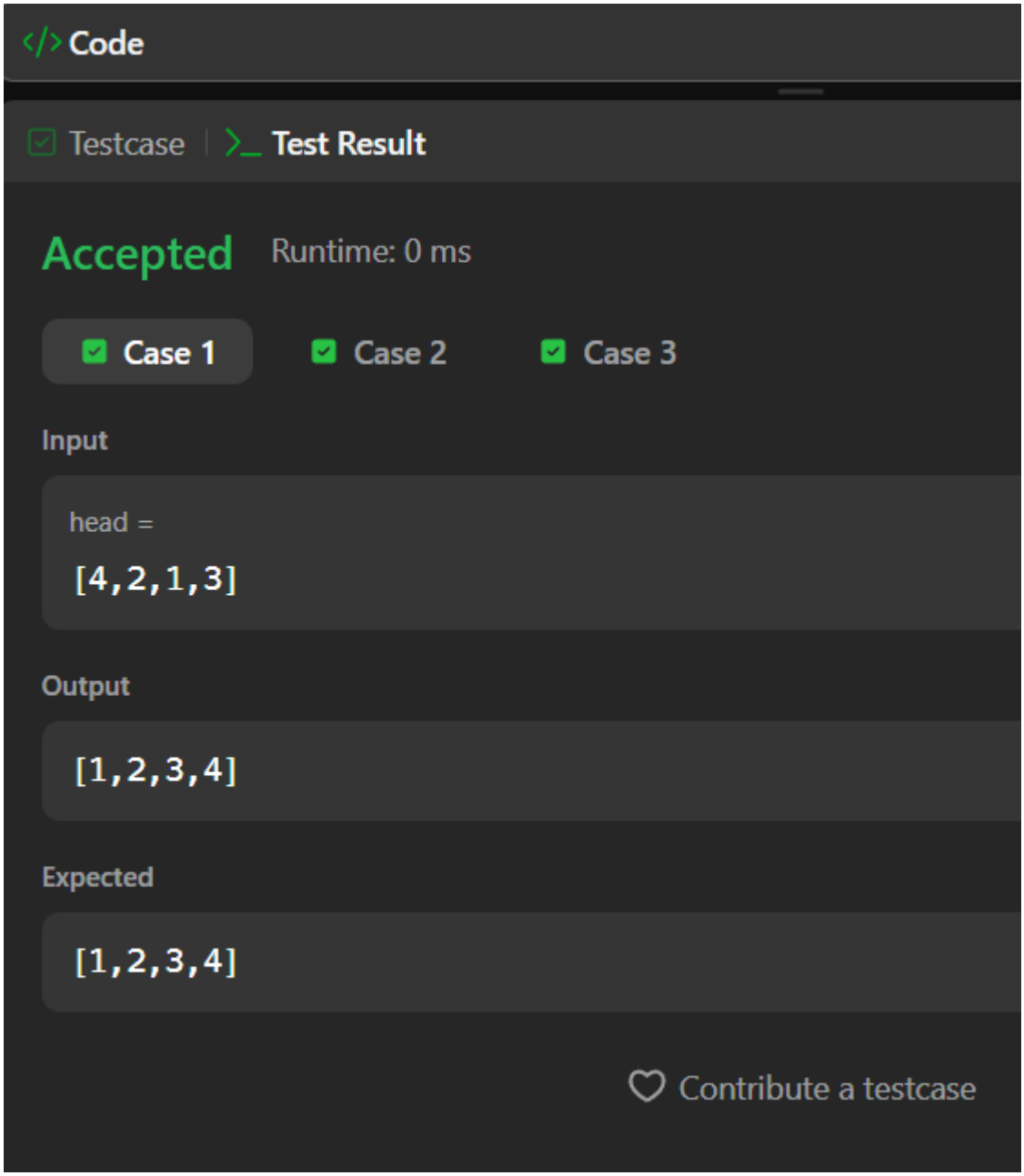
Constraints:

- The number of nodes in the list is in the range `[0, 5 * 10^4]`.
- `-10^5 <= Node.val <= 10^5`

Code

C Auto

```c
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     struct ListNode *next;
 * };
 */
struct ListNode* sortList(struct ListNode* head) {
    if (head == NULL || head->next == NULL)
        return head;

    struct ListNode *slow = head, *fast = head, *prev = NULL;

    while (fast != NULL && fast->next != NULL) {
        prev = slow;
        slow = slow->next;
        fast = fast->next->next;
    }

    prev->next = NULL;

    struct ListNode* left = sortList(head);
    struct ListNode* right = sortList(slow);

    struct ListNode dummy;
    struct ListNode* tail = &dummy;
    dummy.next = NULL;

    while (left && right) {
        if (left->val <= right->val) {
            tail->next = left;
            left = left->next;
        } else {
            tail->next = right;
            right = right->next;
        }
        tail = tail->next;
    }

    tail->next = (left) ? left : right;

    return dummy.next;
}
```


Code
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Case 3
Input
head =
[4,2,1,3]
Output
[1,2,3,4]
Expected
[1,2,3,4]
Contribute a testcase