Experiment 2

Student Name: Amit Kumar Sahu **UID:** 22BCS50073
Branch: BE-CSE **Section/Group:** 22BCS_NTPP_IOT_602'A'
Semester: 6th **Date of Performance:** 16-01-25
Subject Name: AP Lab-2 **Subject Code:** 22CSP-351

1. **Aim: Merge Two Sorted Linked Lists**

2. **Objective:**
   You are given the heads of two sorted linked lists list1 and list2.

   Merge the two lists into one sorted list. The list should be made by splicing together the nodes of the first two lists.

   Return the head of the merged linked list.
   Example 1:

   Input: list1 = [1,2,4], list2 = [1,3,4]
   Output: [1,1,2,3,4,4]

3. **Implementation/Code:**

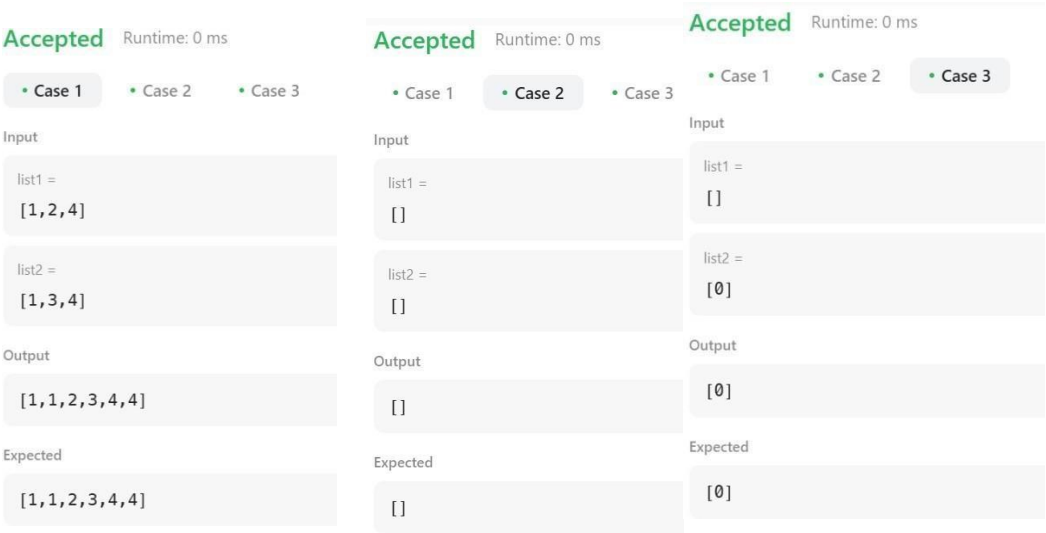
```
ListNode* mergeTwoLists(ListNode* list1, ListNode* list2) {

  ListNode dummy(0);
  ListNode* tail = &dummy;

  while (list1 != NULL && list2 != NULL) {
     if (list1->val < list2->val) {
        tail->next = list1;
        list1 = list1->next;
```

```
        } else {
            tail->next = list2;
            list2 = list2->next;
        }
        tail = tail->next;
    }
    if (list1 != NULL) {
        tail->next = list1;
    } else {
        tail->next = list2;
    }
    return dummy.next;
    }
```

4. Output



5. Learning Outcome:

i. We Learn About the use of Dummy Node.
ii. We Learn About the use of ll.
iii. We Learn About the use of indexing.
iv. We learn About the ordered Next Node.
v. We Learn About the Calling For the function.

Question 2.

1. **Aim: Remove Duplicates from a Sorted List**

2. **Objective:**
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]

3. **Implementation/Code:**

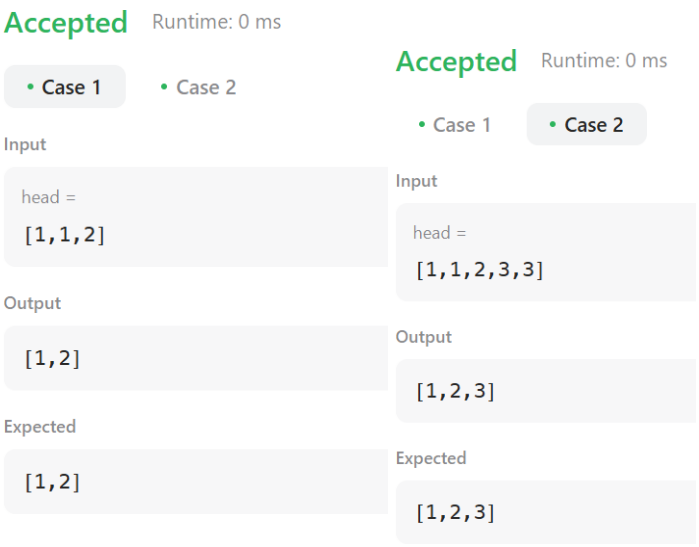
```
ListNode* deleteDuplicates(ListNode* head) {
if (head == NULL) return NULL;
ListNode* current = head;

while (current->next != NULL) {
   if (current->val == current->next->val) {

      ListNode* temp = current->next;
      current->next = current->next->next;
      delete temp;
```

```
            } else {
                current = current->next;
            }
        }
        return head;
    }
```

4. Output



5. Learning Outcome:

I. We Learn About the use of Running LL.
II. We Learn About the use of temp ll.
III. We Learn About the use of Delete function.
IV. We learn About the move from one node.
V. We Learn About the use of while loop.