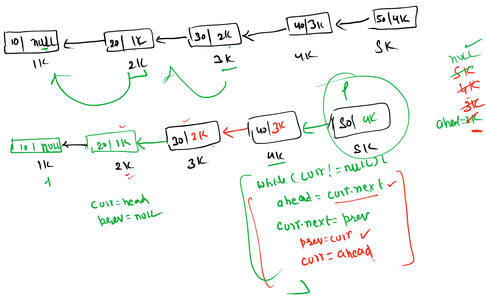
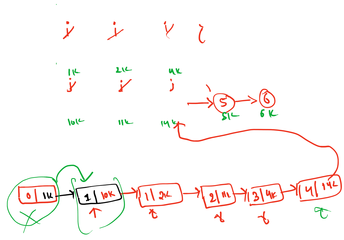
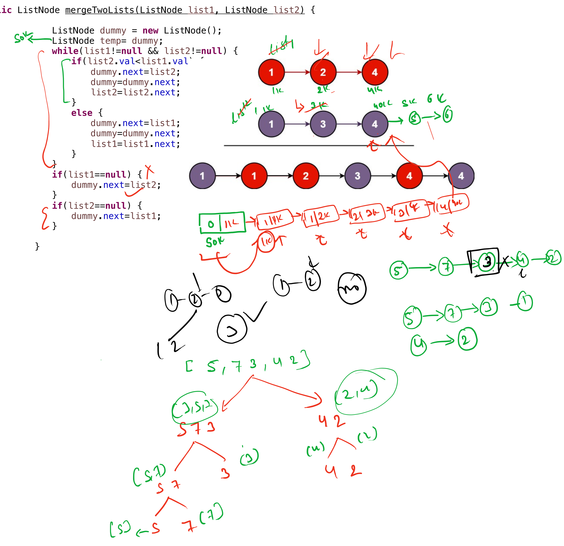
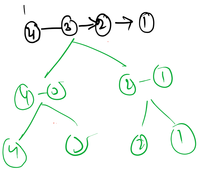




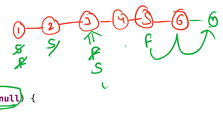
```
public ListNode mergeTwoLists(ListNode list1, ListNode list2) {
    ListNode dummy = new ListNode();
    ListNode temp= dummy;
    while(list1!=null && list2!=null) {
        if(list2.val<list1.val)
            dummy.next=list2;
            dummy=dummy.next;
            list2=list2.next;
        }
        else {
            dummy.next=list1;
            dummy=dummy.next;
            list1=list1.next;
        }
    }
    if(list1==null) {
        dummy.next=list2;
    }
    if(list2==null) {
        dummy.next=list1;
    }
}
```

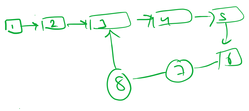


```
public ListNode sortList(ListNode head) {
        if (head == null || head.next == null) {
            return head;
        }
        ListNode mid = middleNode(head);
        ListNode headb = mid.next;
        mid.next = null;
        ListNode f = sortList(head);
        ListNode s = sortList(headb);
        return mergeTwoLists(f, s);
    }
```



```
public ListNode middleNode(ListNode head) {
        ListNode slow = head;
        ListNode fast = head;
        while (fast != null && fast.next != null) {
            slow = slow.next;
            fast = fast.next.next;
        }
        return slow;
    }
```

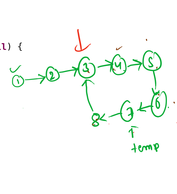


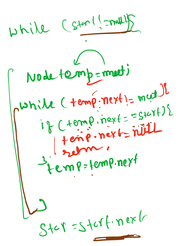


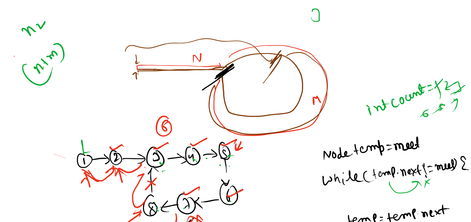
```
public Node hasCycle() {
    Node slow = head;
    Node fast = head;
    while (fast != null && fast.next != null) {
        slow = slow.next;
        fast = fast.next.next;
        if (slow == fast) {
            return slow;
        }
    }
    return null;
}

public void RemoveCycle() {
    Node meet = hasCycle();
    if (meet == null) {
        return;
    }

}
```







```
public void RemoveCycle2() {
    Node meet = hasCycle();
    if (meet == null) {
        return;
    }
    // cycle ki lenght
    int c = 1;
    Node temp = meet;
    while (temp.next != meet) {
        c++;
        temp = temp.next;
    }
    // fast move kra do c distance
    Node fast=head;
    for (int i = 0; i < c; i++) {
        fast=fast.next;
    }
}
```