

```

#include <iostream>
using namespace std;

class LinkedList {
    class Node {
    public:
        int val;
        Node* next;
        Node(int x) {
            val=x;
            next=NULL;
        }
    };
    Node* head;
    Node* tail;
public:
    LinkedList() {
        head = NULL;
        tail = NULL;
    }
    void append(int val) {
        Node* new_node = new Node(val);
        if (head == NULL)
            head = new_node;
        else
            tail->next=new_node;
        tail=new_node;
    }
    void mergeList(LinkedList list1, LinkedList list2) {
        Node* current_node1 = list1.head;
        Node* current_node2 = list2.head;
        while (current_node1 != NULL && current_node2 != NULL) {
            if (current_node1->val <= current_node2->val) {
                append(current_node1->val);
                current_node1 = current_node1->next;
            }
            else {
                append(current_node2->val);
                current_node2 = current_node2->next;
            }
        }
        while (current_node1 != NULL) {
            append(current_node1->val);
            current_node1 = current_node1->next;
        }
        while (current_node2 != NULL) {
            append(current_node2->val);
            current_node2 = current_node2->next;
        }
    }
    void display() {
        Node* tmp = head;
        while (tmp != NULL) {
            cout << tmp->val << " ";
            tmp = tmp->next;
        }
        cout << endl;
    }
};

int main() {
    LinkedList list1;
    list1.append(1);
    list1.append(3);
    list1.append(5);
    LinkedList list2;
    list2.append(2);
    list2.append(4);
    list2.append(6);
    LinkedList merged_list;
    merged_list.mergeList(list1, list2);
    merged_list.display();
    return 0;
}

```