```cpp
#include <iostream>
using namespace std;

struct Node {
    int data;
    Node* next;

    Node(int value) {
        data = value;
        next = nullptr;
    }
};

class LinkedList {
public:
    Node* head;

    LinkedList() {
        head = nullptr;
    }

    // Function to append data to the linked list
    void append(int value) {
        Node* newNode = new Node(value);
        if (head == nullptr) {
            head = newNode;
        } else {
            Node* temp = head;
            while (temp->next != nullptr) {
                temp = temp->next;
            }
            temp->next = newNode;
        }
    }

    // Function to display the linked list
    void display() {
        Node* temp = head;
        while (temp != nullptr) {
            cout << temp->data << " ";
            temp = temp->next;
        }
        cout << endl;
    }
};

LinkedList mergeLists(LinkedList& list1, LinkedList& list2) {
    if (list1.head == nullptr) return list2;
    if (list2.head == nullptr) return list1;

    Node* temp = list1.head;
    while (temp->next != nullptr) {
        temp = temp->next;
    }
    temp->next = list2.head;

    return list1;
}

int main() {
    // Create Linked List A
    LinkedList listA;
    listA.append(1);
    listA.append(3);
    listA.append(5);
```

```
    // Create Linked List B
    LinkedList listB;
    listB.append(2);
    listB.append(4);
    listB.append(6);

    // Merge A and B
    LinkedList mergedList = mergeLists(listA, listB);

    // Display the merged list
    cout << "Merged Linked List: ";
    mergedList.display();

    return 0;
}
```