Merge Two Sorted Linked List LeetCode

Write a C++ program to merge two sorted linked lists.

Approach 1: Iterative approach to merge two sorted linked lists

- Initialize pointers to keep track of the merged list and the previous node.
- Traverse both linked lists simultaneously, comparing their values.
- Connect the smaller value node to the merged list, updating the merged list and the previous node pointers.
- Continue this process until both lists are exhausted.
- Finally, append any remaining nodes from both lists to the merged list.

Time Complexity: O(m + n), where m and n are the lengths of the two linked lists.

Space Complexity: O(1), as only a constant amount of additional space is used.

Approach 2: Recursive approach to merge two sorted linked lists

- Base cases: If one of the lists is empty, return the other list.
- Compare the values of the current nodes in the two lists.
- Recursively merge the rest of the lists and connect the nodes accordingly.
- Return the merged list.

Time Complexity: O(m + n), where m and n are the lengths of the two linked lists.

Space Complexity: O(m + n), as the recursive call stack can grow to accommodate both lists' lengths.