**Question:**

Given two sorted singly linked lists, write a method called merge\_sorted\_lists(list1, list2) that will merge them into a single sorted linked list using recursion. If one list runs out before the other, append the rest of the remaining list.

| **Sample Input** | **Sample Output** |
| --- | --- |
| **List 1: 2 -> 4 -> 7 -> None**  **List 2: 1 -> 3 -> 5 -> 6 -> 8 -> None** | **1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 8 -> None** |

**Question:**

Given two singly linked lists as inputs, write a method called merge\_alternate(listA, listB) that merges them by alternating nodes from each list using recursion. If one list runs out before the other, append the rest of the remaining list.

| **Sample Input** | **Sample Output** |
| --- | --- |
| List A: 10 -> 20 -> 30 -> None  List B: 5 -> 15 -> 25 -> 35 -> 45 -> None | 10 -> 5 -> 20 -> 15 -> 30 -> 25 -> 35 -> 45 -> None |