Linked List Problems

Problem 1: Reverse a Linked List

Problem Statement:
Given the head of a singly linked list, reverse the list and return the new head.

Input Format:
- A singly linked list represented by its head node.

Output Format:
- The head node of the reversed linked list.

Example:
Input: 1 -> 2 -> 3 -> 4 -> 5
Output: 5 -> 4 -> 3 -> 2 -> 1

Problem 2: Find the Middle Element of a Linked List

Problem Statement:
Given the head of a singly linked list, find and return the middle node.
If there are two middle nodes, return the second middle node.

Input Format:
- A singly linked list represented by its head node.

Output Format:
- The middle node of the linked list.

Example:
Input: 1 -> 2 -> 3 -> 4 -> 5
Output: 3

Input: 1 -> 2 -> 3 -> 4 -> 5 -> 6
Output: 4

Problem 3: Merge Two Sorted Linked Lists

Problem Statement:
Given the heads of two sorted singly linked lists, merge them into a single sorted linked list and return the head.

Input Format:
- Two sorted singly linked lists represented by their head nodes.

Output Format:
- The head node of the merged sorted linked list.

Example:
Input: List1 = 1 -> 2 -> 4, List2 = 1 -> 3 -> 4
Output: 1 -> 1 -> 2 -> 3 -> 4 -> 4

Problem 4: Find Intersection Point of Two Linked Lists

Problem Statement:
Given the heads of two singly linked lists, return the node where the two lists intersect.
If the two linked lists have no intersection, return null.

Input Format:
- Two singly linked lists represented by their head nodes.

Output Format:
- The intersecting node or null if no intersection exists.

Example:
Input: ListA = 4 -> 1 -> 8 -> 4 -> 5, ListB = 5 -> 6 -> 1 -> 8 -> 4 -> 5
Output: Node with value 8

Input: ListA = 2 -> 6 -> 4, ListB = 1 -> 5
Output: null

Problem 5: Detect and Delete Duplicates in a Linked List

Problem Statement:
Given the head of a sorted singly linked list, remove all duplicates such that each element appears only once.

Input Format:

- A sorted singly linked list represented by its head node.

Output Format:
- The head node of the linked list with duplicates removed.

Example:
Input: 1 -> 1 -> 2 -> 3 -> 3
Output: 1 -> 2 -> 3