

## 1. Reverse Linked List

Reverse a singly linked list and return the new head. You must reverse it **in-place** (no arrays or extra list).

**Example:**

**Input:** 1 → 2 → 3 → 4 → 5

**Output:** 5 → 4 → 3 → 2 → 1

## 2. Merge Two Sorted Lists

Given two sorted linked lists, merge them into one sorted list.

**Example:**

**Input:**

List1 = 1 → 3 → 5

List2 = 2 → 4 → 6

**Output:** 1 → 2 → 3 → 4 → 5 → 6

## 3. Middle of the Linked List

Find the middle node of a singly linked list. If there are two middle nodes, return the second one.

In even nodes, you can consider any node from two as middle.

**Example:**

Input: 1 → 2 → 3 → 4 → 5

Output: 3

Input: 1 → 2 → 3 → 4 → 5 → 6

Output: 3 or 4

## 4. Remove Nth Node from End

Remove the **nth node from the end** of a linked list.

**Example:**

**Input:** head = [1, 2, 3, 4, 5], n = 2

**Output:** [1, 2, 3, 5]

## 5. Linked List Cycle Detection

Detect if a linked list contains a cycle (loop).

**Example:**

**Input:** head = [3, 2, 0, -4], tail connects to node index 2

**Output:** true

**Input:** head = [3, 2, 0, -4], (3 → 2 → 0 → -4 → NULL )

**Output:** false

## 7. Add Two Numbers (Linked Lists)

Each linked list represents a number in reverse order. Add the two numbers and return the sum as a linked list.

**Example:**

**Input:**

L1 = 2 → 4 → 3 (represents 342)

L2 = 5 → 6 → 4 (represents 465)

**Output:** 7 → 0 → 8 (represents 807)

## 8. Reorder List

1. Find the middle node.
2. Reverse the second half.
3. Merge both halves alternately.

**Example:**

**Input:** 1 → 2 → 3 → 4 → 5

**Output:** 1 → 5 → 2 → 4 → 3

## 9. Remove Duplicates from Sorted Linked List

Given a **sorted** linked list, delete all duplicate elements so that each element appears only once.

**Example:**

**Input:** 1 → 1 → 2 → 3 → 3

**Output:** 1 → 2 → 3

## 10. Reverse Nodes in k-Group:

Given the head of a singly linked list, reverse the nodes of the list **k at a time** and return the modified list. Nodes that remain fewer than **k** at the end should stay as they are (not reversed). Example:

**Input:** head = [1, 2, 3, 4, 5], k = 2

**Output:** [2, 1, 4, 3, 5]

**NOTE:** Your code should work for any value of K.