



## Assignment : Linkedlist – 3

**Q1** Given the head of a sorted linked list, delete all nodes that have duplicate numbers, leaving only distinct numbers from the original list. Return the linked list sorted as well. [Leetcode 82].

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

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

**Q2** Given the head of a singly linked list, sort the list using insertion sort, and return the sorted list's head. [Leetcode 147].

The steps of the **insertion sort** algorithm:

1. Insertion sort iterates, consuming one input element each repetition and growing a sorted output list.
2. At each iteration, insertion sort removes one element from the input data, finds the location it belongs within the sorted list and inserts it there.
3. It repeats until no input elements remain

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

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

**Q3** Given the head of a singly linked list, group all the nodes with odd indices together followed by the nodes with even indices, and return the reordered list.

The **first** node is considered **odd**, and the **second** node is **even**, and so on.

Note that the relative order inside both the even and odd groups should remain as it was in the input.

You must solve the problem in  $O(1)$  extra space complexity and  $O(n)$  time complexity. [Leetcode 328]

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

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

**Q4** You are given two **non-empty** linked lists representing two non-negative integers. The digits are stored in **reverse order**, and each of their nodes contains a single digit. Add the two numbers and return the sum as a linked list. [Leetcode 2]

You may assume the two numbers do not contain any leading zero, except the number 0 itself.

**Input:** l1 = [2,4,3], l2 = [5,6,4]

**Output:** [7,0,8]

**Explanation:** 342 + 465 = 807.

**Q5** Given head which is a reference node to a singly-linked list. The value of each node in the linked list is either 0 or 1. The linked list holds the binary representation of a number.

Return the decimal value of the number in the linked list.

The **most significant bit** is at the head of the linked list. [Leetcode 1290]

**Input:** head = [1,0,1]

**Output:** 5

**Explanation:** (101) in base 2 = (5) in base 10

**Q6** In a linked list of size n, where n is **even**, the ith node (**0-indexed**) of the linked list is known as the **twin** of the (n-1-i)th node, if  $0 \leq i \leq (n / 2) - 1$ .

- For example, if n = 4, then node 0 is the twin of node 3, and node 1 is the twin of node 2. These are the only nodes with twins for n = 4.

The **twin sum** is defined as the sum of a node and its twin.

Given the head of a linked list with even length, return the **maximum twin sum** of the linked list. [Leetcode 2130]

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

**Output:** 6

**Explanation:** Nodes 0 and 1 are the twins of nodes 3 and 2, respectively. All have twin sum = 6. There are no other nodes with twins in the linked list. Thus, the maximum twin sum of the linked list is 6.

**Q7** Given the head of a linked list, reverse the nodes of the list k at a time, and return the modified list.

k is a positive integer and is less than or equal to the length of the linked list. If the number of nodes is not a multiple of k then left-out nodes, in the end, should remain as it is.

You may not alter the values in the list's nodes, only nodes themselves may be changed. [Leetcode 25]

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

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