Linked Lists - Problems

Problem - 1: Implement Stack using Linked List.

Problem - 2: Find nth node from the end of a Linked List.

Problem - 3: Check whether the given linked list is either NULL-terminated or ends in a cycle (cyclic).

Problem - 4: Check whether the given linked list is NULL-terminated or not. If there is a cycle find the start node of the loop.

Problem - 5: Insert node in sorted linked list.

Problem - 6: Reverse singly linked list.

Problem - 7: Find the intersect node of two linked lists.

Problem - 8: Find the middle node of the linked list.

Problem - 9: Display linked list from the end.

Problem - 10: If the head of a linked list is pointing to Kth element, then how will you get the elements before Kth element?

Problem - 11: Given two sorted lists, we need to merge them into the third list in sorted order.

Problem - 12: Reverse the linked list in pairs.

Problem - 13: Given a binary tree, convert into Doubly linked list.

Problem - 14: How do we sort the Linked lists?

Problem - 15: Check if linked list is palindrome or not?

Problem - 16: Exchange the adjacent elements in list.

Problem - 17: For given K value (K > 0), reverse block of K nodes in a list.

Problem - 18: JosephusCircle: N people have decided to elect a leader by arranging themselves in a circle and eliminating every Mth person around the circle, closing ranks as each person drops out. Find which person will be the last one remaining (with rank 1)

Problem - 19: Copy List with Random Pointer

Problem - 20: Find modular node: Given a singly linked list, write a function to find the last element from the beginning whose n%k == 0, where n is the number no nodes in list and k is a constant integer. Ex: if n = 19, k = 3 then we should return 18^{th} node.

Problem - 21: Reorder list: Given a singly linked list L: L1 -> L2 -> L3 ->……->LN, reorder it to L1 -> LN -> L2 -> LN-1…….

Problem - 22: Implement Insertion sort for Linked list.

Problem - 23: Add two numbers.

Problem - 24: Merge K sorted lists and return is as one sorted list.

Problem - 25: Given a list, rotate the list to the right by k places.

Problem - 26: Given a linked list with even and odd numbers, create an algorithm for making changes to the list in such a way that all even numbers appear at beginning.

Problem - 27: Print common elements from 2 linked lists.

Problem - 28: Remove the duplicate elements from sorted list.

Problem - 29: Swapping Nodes in a Linked List

Please feel to add any question to this list if you feel it deserve the place.