DSA Topics

Linkedlist

# All basic operations like insertion,deletion,traversal,search,find,etc.

# Insert a node in a sorted linked list.

1. Reverse linked list (iterative & recursive)

2. Reverse linked list in a group of given size k (with & w/o recursion)
For a given K value (K > 0) reverse blocks of K nodes in a list.
Example: Input: 1 2 3 4 5 6 7 8 9 10. Output for different K values:
For K = 2: 2 1 4 3 6 5 8 7 10 9
For K = 3: 3 2 1 6 5 4 9 8 7 10
For K = 4: 4 3 2 1 8 7 6 5 9 10

# Reverse the linked list in pairs. If you have a linked list that holds 1 → 2 → 3 → 4 → X, then after the function has been called the linked list would hold 2 → 1 → 4 →
3 → X.

3. Detect and delete loop in LL,isloop/iscycle,issorted /
Check whether the given linked list is either NULL-terminated or ends in a cycle (cyclic).,Floyd cycle finding algorithm.. If there is a cycle, find the length of the loop.

OR

Are given a pointer to the first element of a linked list L. There are two possibilities for L: it either ends (snake) or its last element points back to one of the earlier elements in the list (snail). Give an algorithm that tests whether a given list L is a snake or a snail.

4. Find starting point/node of loop

5. Remove duplicates in unsorted,sorted LL,combine duplicates in LL,Doubly LL

6. Write a Program to Move the last element to Front in a Linked List.

7. Add “1” to a number represented as a Linked List.

8. Add two numbers represented by linked lists.

9. Intersection of two Sorted Linked List.

# Union of two linked lists.

10. Intersection Point of two Linked Lists.

11. Merge Sort For Linked lists.,merge k sorted lists.[Very Important]

# Merge two sorted linked lists such that merged list is in reverse order.

12. Quicksort for Linked Lists.[Very Important]

13. Find the middle Element of a linked list.& delete middle of the LL

14. Check if a linked list is a circular linked list.

15. Split a Circular linked list into two halves.(without counting no of nodes in LL)

# Split a Circular Linked List into two equal parts. If the number of nodes in the list are odd then make first list one node extra than second list.

16. Write a Program to check whether the Singly Linked list is a palindrome or not.
    (Check if the LL of strings form a palindrome)/
    Function to check if a singly linked list is palindrome.

17. Deletion from a Circular Linked List.

# Given only a pointer/reference to a node to be deleted in a singly list.

18. Reverse a Doubly Linked list.

# Given a binary tree convert it to doubly linked list.

19. Find pairs with a given sum in a DLL.

# Pairwise swap elements of a given linked list.

20. Count triplets in a sorted DLL whose sum is equal to given value “X”.

21. Sort a “k”sorted Doubly Linked list.[Very IMP]

22. Rotate DoublyLinked list by N nodes.

23. Rotate a Doubly Linked list in group of Given Size.[Very IMP]

24. Flatten a Linked List

25. Sort a LL of 0's, 1's and 2's

26. Clone a linked list with next and random pointer.
Given a linked list consists of data, a next pointer and also a random pointer which points to a random node of the list. Give an algorithm for cloning the list.

27. Merge K sorted Linked list.

# Given two sorted Linked Lists, how to merge them into the third list in sorted order?

28. Multiply 2 no. represented by LL

29. Delete nodes which have a greater value on right side

30. Segregate even and odd nodes in a Linked List

31. Program for n’th node from the end of a Linked List

32. Find the first non-repeating character from a stream of characters

33. Polynomials operations using LL:Add,multiply,expressions
(coefficient,exponent,pointer)

34. Count and sum of node elements in LL

35. Doubly linked list operations : Insertion & deletion

36. Circular linked list operations : Insertion and deletion

# Check if a linked list is Circular Linked List.

# Sorted insert for circular linked list.

37. Realizing linkedlist using arrays and pointers

38. Max and min element node in LL

39. String operations and reversal using LL and pointers

40. Implementation of stacks and queues.

41. Implementation of graphs: Adjacency list representation of graphs is most popular which uses
linked list to store adjacent vertices.

42. Dynamic memory allocation: We use linked list of free blocks.

43. Maintaining directory of names. Performing arithmetic operations on long integers

44. Manipulation of polynomials by storing constants in the node of linked list. Representing sparse
matrices using LL(Sparse matrix representations using arrays & LL)

45. Circular Doubly Linked Lists are used for implementation of advanced data structures
like Fibonacci Heap.

46. Remove nth node from the end of the list.: Use two pointers pNthNode and pTemp.

# Remove every k-th node of the linked list.

47. Length of longest palindrome list in a linked list using O(1) extra space.

48. Round robin algo using circular LL.

49. Counting Nodes in a Circular Linked List

50. Printing the Contents of a Circular Linked List.

51. Suppose there are two singly linked lists both of which intersect at some point and become a single linked list. The head or start pointers of both the lists are known, but
the intersecting node is not known. Also, the number of nodes in each of the lists before
they intersect is unknown and may be different in each list. List1 may have n nodes before
it reaches the intersection point, and List2 might have m nodes before it reaches the
intersection point where m and n may be m = n,m < n or m > n. Give an algorithm for finding the merging point.

(FIND INTERSECTING NODE)

Also use stacks to solve the scenario.

52. How will you display a Linked List from the end?

53. Check whether the given Linked List length is even or odd?

54. If the head of a Linked List is pointing to kth element, then how will you get the
elements before kth element?
(Memory Efficient Linked Lists [XOR Linked Lists])

55. If we want to concatenate two linked lists which of the following gives O(1) complexity?
1) Singly linked lists

2) Doubly linked lists
3) Circular doubly linked lists

56. Josephus Circle: 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).

57.