

# Linked List

## *Intermediate Level Questions:*

1. Write a program to get the “N<sup>th</sup>” Node from the end of the Singly Linked List.

[Practice here: <https://practice.geeksforgeeks.org/problems/nth-node-from-end-of-linked-list/1> ]

2. Write a Program to check whether the Singly Linked list is a palindrome or not.

[Practice here: <https://practice.geeksforgeeks.org/problems/check-if-linked-list-is-pallindrome/1> ]

3. Write a Program to reverse the Linked List. (Both Iterative and recursive)

[Practice here: <https://practice.geeksforgeeks.org/problems/reverse-a-linked-list/1> ]

4. Reverse a Linked List in group of Given Size. [Very Imp]

[Practice here: <https://practice.geeksforgeeks.org/problems/reverse-a-linked-list-in-groups-of-given-size/1> ]

5. Write a program to Detect loop in a linked list.

[Practice here: <https://practice.geeksforgeeks.org/problems/detect-loop-in-linked-list/1> ]

6. Write a program to find the length of loop in the linked list.

[Practice here: <https://practice.geeksforgeeks.org/problems/find-length-of-loop/1> ]

7. Write a function to delete the Linked List.

[Follow: <https://www.geeksforgeeks.org/write-a-function-to-delete-a-linked-list/> ]

8. Remove Duplicates in a sorted Linked List.

[Practice here: <https://practice.geeksforgeeks.org/problems/remove-duplicate-element-from-sorted-linked-list/1> ]

9. Remove Duplicates in a Unsorted Linked List.

[Practice here: <https://practice.geeksforgeeks.org/problems/remove-duplicates-from-an-unsorted-linked-list/1> ]

10. Write a Program to Move the last element to Front in a Linked List.  
[Follow: <https://www.geeksforgeeks.org/move-last-element-to-front-of-a-given-linked-list/> ]
11. Add "1" to a number represented as a Linked List.  
[Practice here: <https://practice.geeksforgeeks.org/problems/add-1-to-a-number-represented-as-linked-list/1> ]
12. Add two numbers represented by linked lists.  
[Practice here: <https://practice.geeksforgeeks.org/problems/add-two-numbers-represented-by-linked-lists/1> ]
13. Intersection of two Sorted Linked List.  
[Practice here: <https://practice.geeksforgeeks.org/problems/intersection-of-two-sorted-linked-lists/1> ]
14. Intersection Point of two Linked Lists.  
[Practice here: <https://practice.geeksforgeeks.org/problems/intersection-point-in-y-shapped-linked-lists/1> ]
15. Merge Sort For Linked lists.**[Very Important]**  
[Follow: <https://www.geeksforgeeks.org/merge-sort-for-linked-list/> ]
16. Quicksort for Linked Lists.**[Very Important]**  
[Follow: <https://www.geeksforgeeks.org/quicksort-on-singly-linked-list/> ]
17. Find the middle Element of a linked list.  
[Practice here: <https://practice.geeksforgeeks.org/problems/finding-middle-element-in-a-linked-list/1> ]
18. Check if a linked list is a circular linked list.  
[Practice here: <https://practice.geeksforgeeks.org/problems/circular-linked-list/1> ]
19. Split a Circular linked list into two halves.  
[Practice here: <https://practice.geeksforgeeks.org/problems/split-a-circular-linked-list-into-two-halves/1> ]

20. Deletion from a Circular Linked List.  
[Follow here: <https://www.geeksforgeeks.org/deletion-circular-linked-list/> ]
21. Count Nodes in a Circular Linked List.  
[Follow here: <https://www.geeksforgeeks.org/count-nodes-circular-linked-list/> ]
22. Exchange first and last nodes in a linked list.  
[Follow here: <https://www.geeksforgeeks.org/exchange-first-last-node-circular-linked-list/> ]
23. Reverse a Doubly Linked list.  
[Practice here: <https://practice.geeksforgeeks.org/problems/reverse-a-doubly-linked-list/1> ]
24. Find pairs with a given sum in a DLL.  
[Follow here: <https://www.geeksforgeeks.org/find-pairs-given-sum-doubly-linked-list/> ]
25. Count triplets in a sorted DLL whose sum is equal to given value "X".  
[Follow here: <https://www.geeksforgeeks.org/count-triplets-sorted-doubly-linked-list-whose-sum-equal-given-value-x/> ]
26. Sort a "k" sorted Doubly Linked list.[**Very IMP**]  
[Follow here: <https://www.geeksforgeeks.org/sort-k-sorted-doubly-linked-list/> ]
27. Rotate Doubly Linked list by N nodes.  
[Follow here: <https://www.geeksforgeeks.org/rotate-doubly-linked-list-n-nodes/> ]
28. Rotate a Doubly Linked list in group of Given Size.[**Very IMP**]  
[Follow here: <https://www.geeksforgeeks.org/reverse-doubly-linked-list-groups-given-size/> ]
29. Can we reverse a linked list in less than  $O(n)$  ?  
[Study : <https://www.geeksforgeeks.org/can-we-reverse-a-linked-list-in-less-than-on/> ]
30. Why Quicksort is preferred for. Arrays and Merge Sort for Linked Lists ?  
[Study : <https://www.geeksforgeeks.org/why-quick-sort-preferred-for-arrays-and-merge-sort-for-linked-lists/> ]