

Assignment-58 : [A Job Ready Bootcamp in C++, DSA and IOT](#)

DSA Circular linked list

1. Given a circular doubly linked list and a position n . The task is to delete the node at the given position n from the beginning. Initial circular doubly linked list.
2. Given a sorted circular doubly linked list of distinct nodes (no two nodes have the same data) and a value x . Count triplets in the list that sum up to a given value x .
3. Given a doubly singly linked list, find if the linked list is circular or not.
4. Given a circular linked list, count the number of nodes in it.
5. Given a doubly linked list, we have to convert it into a circular linked list.
Time Complexity: $O(n)$, As we need to move through the whole list to get hold of the last node.
Auxiliary Space: $O(1)$, As constant extra space is used.
6. Given two circular linked lists $L1$ and $L2$, the task is to find if the two circular linked lists are identical or not.
7. Given a circular singly linked list containing N nodes, the task is to delete all the odd nodes from the list.
8. Given a circular doubly-linked list, find the largest node in the doubly linked list.
9. Given a circular doubly linked list containing n nodes. The problem is to reverse every group of k nodes in the list.
10. Write a program to reverse the given circular Doubly Linked List.