Circular Linked List and Operations in Data Structures (With Notes)

🌐 **codewithharry.com**/videos/data-structures-and-algorithms-in-hindi-19

Till now, we have covered linked lists, which consist of a head, the body, and an end pointing to NULL. Basically, it was linear. We could do traversal, insertion, deletion, searching, and many more operations while traversing to the end of it. Today, we'll see a variant of it, circular linked lists. We'll also see all those operations that we could do in a linear linked list and their implementations in a circular linked list.

Introduction:

A circular linked list is a linked list where the last element points to the first element (head) hence forming a circular chain. There is no node pointing to the NULL, indicating the absence of any end node. In circular linked lists, we have a head pointer but no starting of this list.

Refer to the illustration of a circular linked list below:
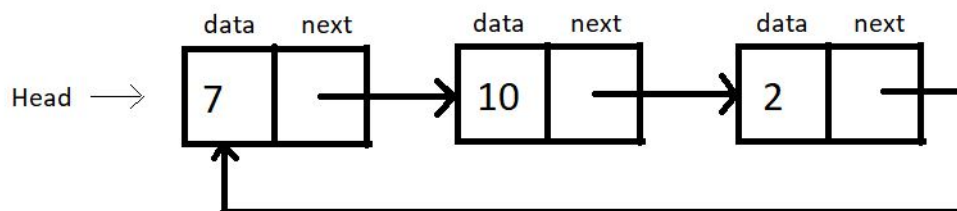
Operations on a Circular Linked List:

Operations on circular linked lists can be performed exactly like a singly linked list. It's just that we have to maintain an extra pointer to check if we have gone through the list once.

Traversal:

- Traversal in a circular linked list can be achieved by creating a new struct Node* pointer p, which starts from the head and goes through the list until it points again at the head. So, this is how we go through this circle only once, visiting each node.
- And since traversal is achieved, all the other operations in a circular linked list become as easy as doing things in a linear linked list.

- One thing that may have sounded confusing to you is that there is a head but no starting of this circular linked list. Yes, that is the case; we have this head pointer just to start or incept in this list and for our convenience while operating on it. There is no first element here.

So this was introducing you all to this variant of linked lists. I would like you all to implement other operations on your own. Our next tutorial will cover them anyway.

Thank you for being with me throughout. I hope you enjoyed the tutorial. If you appreciate my work, please let your friends know about this course too. Don't forget to download the notes from the link below. If you haven't checked out the whole playlist yet, move on to codewithharry.com or my YouTube channel to access it. See you all in the next tutorial, where we'll try to code some operations on a circular linked list in C. Till then, keep learning.

Download Notes here