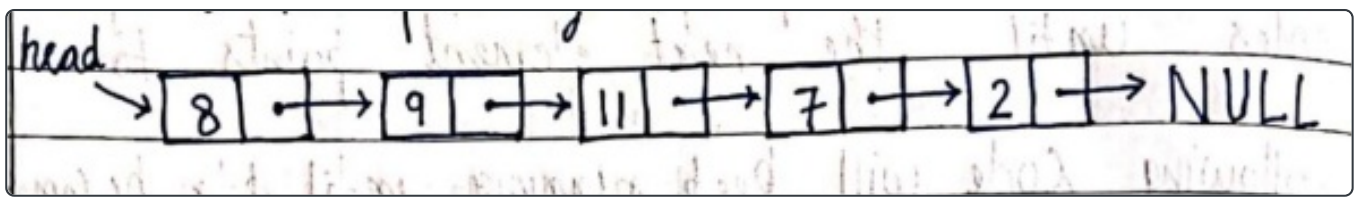
Insertion of a Node in a Linked List Data Structure

Course Content Hide Player

Overview Q&A Files Announcements

Insertion of a Node in a Linked List Data Structure

Consider the following Linked List,



Insertion in this list can be divided into the following categories:

Case 1: Insert at the beginning

Case 2: Insert in between

Case 3: Insert at the end

Case 4: Insert after the node

For insertion of any type we first need to create that extra node, we want to insert and then rewire the connections to the job done!

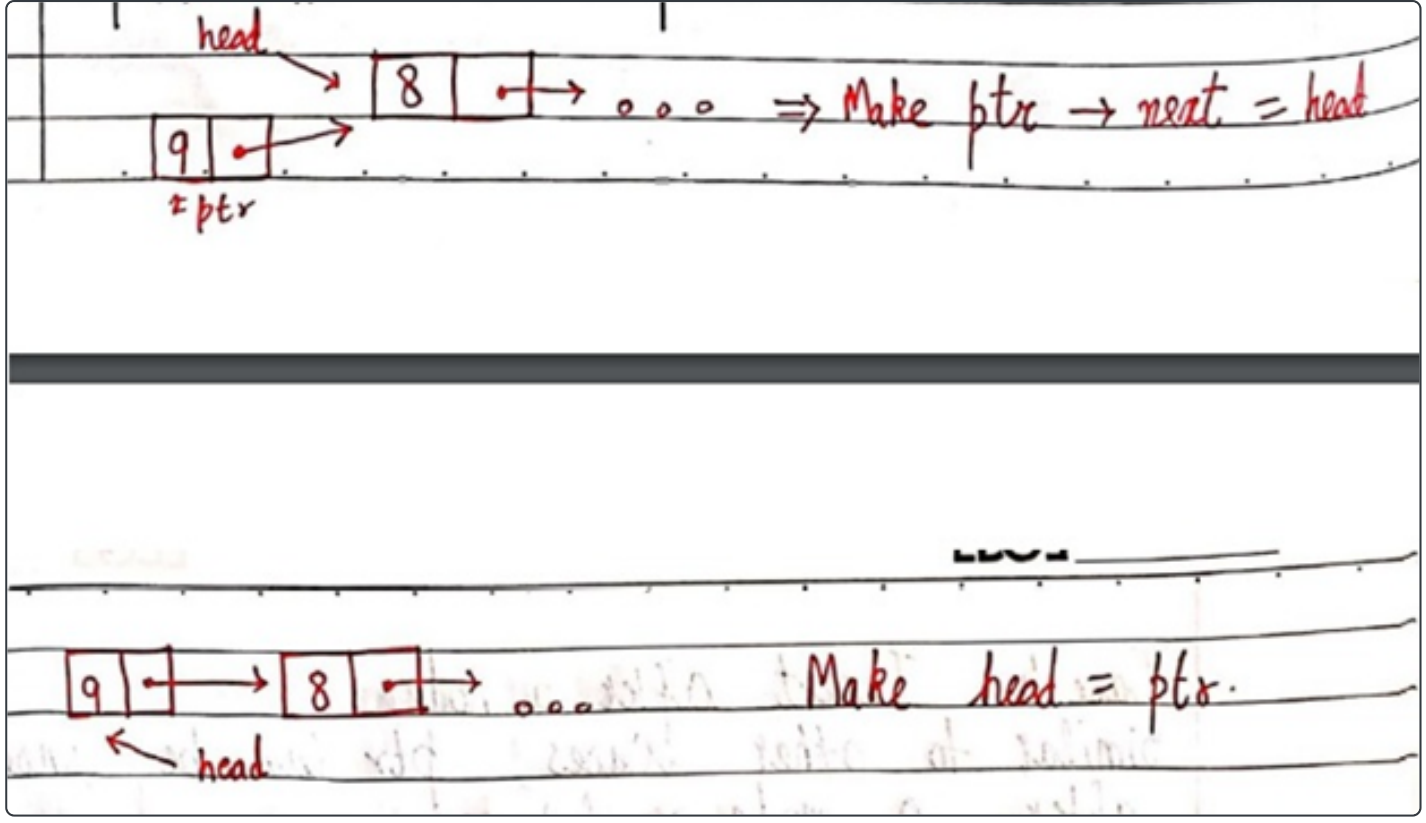
struct Node *ptr = (struct Node*) malloc (sizeof (struct Node))

The above syntax will create a node

ptr -> data = 9 // It will set the data

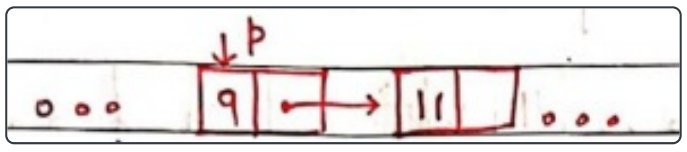
Case 1: Insert at the beginning

In order to insert the node at the beginning, we need to point the node to be inserted (ptr) to head and update the he to ptr.



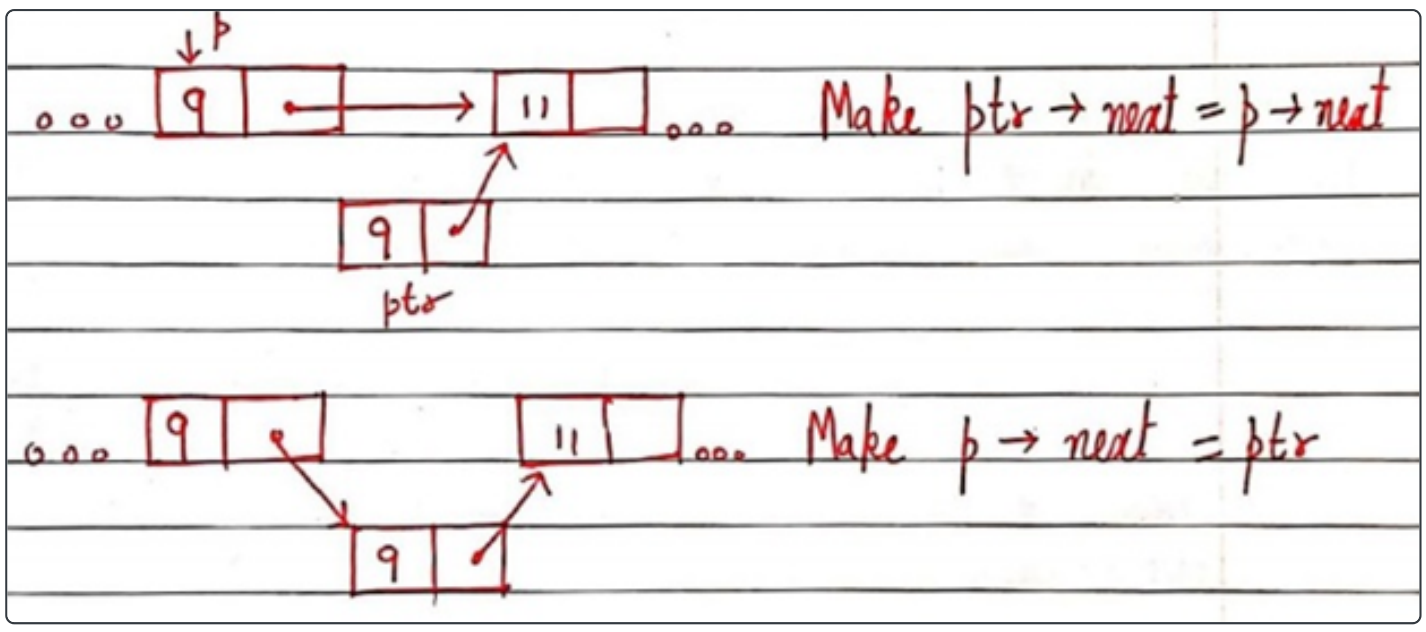
Case 2: Insert in between

Assuming index starts from 0, we can insert an element at index i>0 as follows:



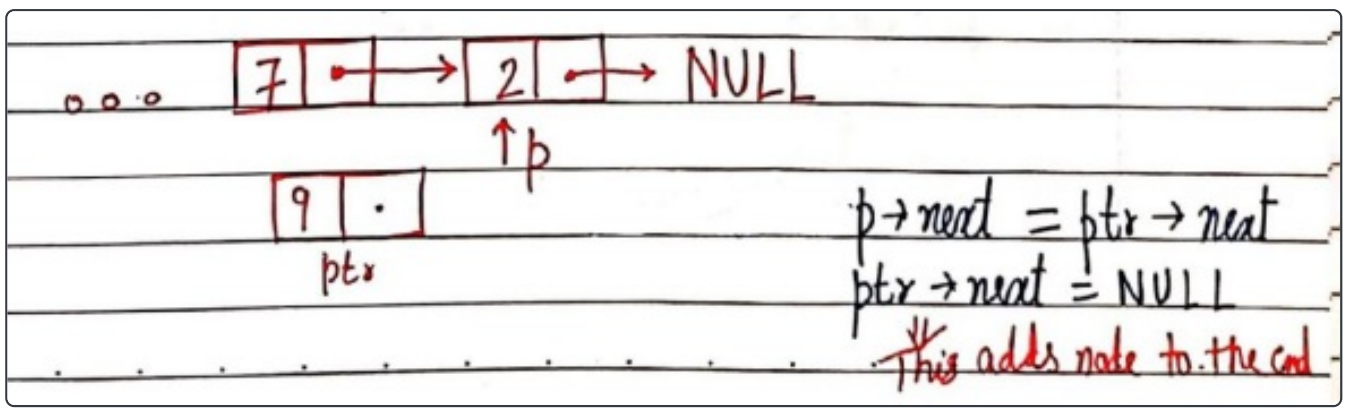
Bring a temporary pointer p before the element you want to insert in the linked list.

Since we want to insert between 9 and 11, we bring p to 9



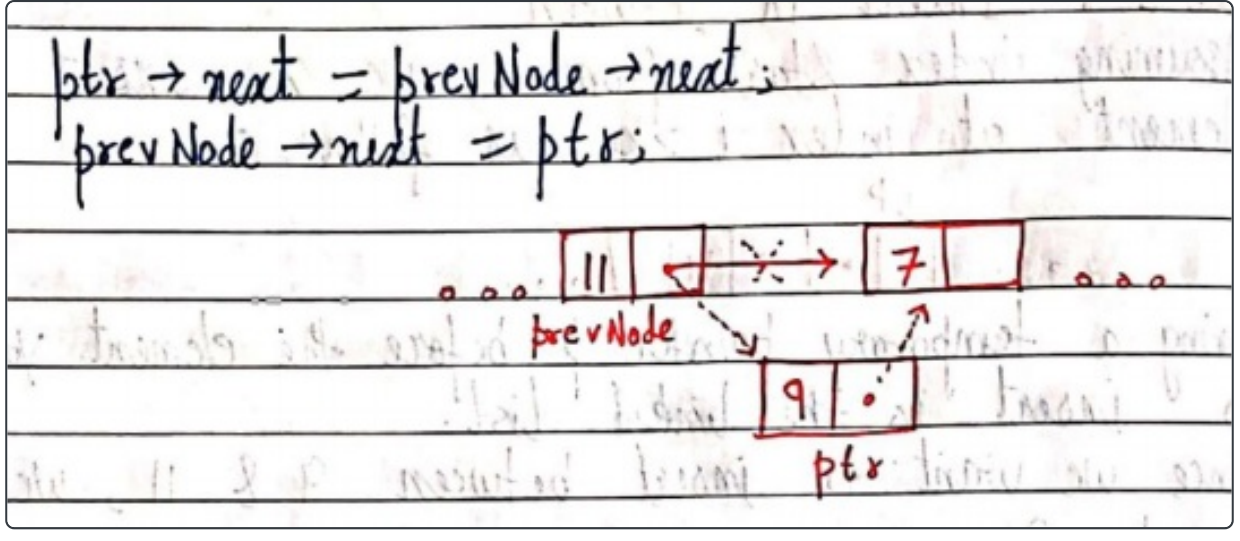
Case 3: Insert at the end

In order to insert an element at the end of the linked list, we bring a temporary pointer to the last element.



Case 4: Insert after a node

Similar to other cases, ptr can be inserted after a node as follows:



← Previous

Nex