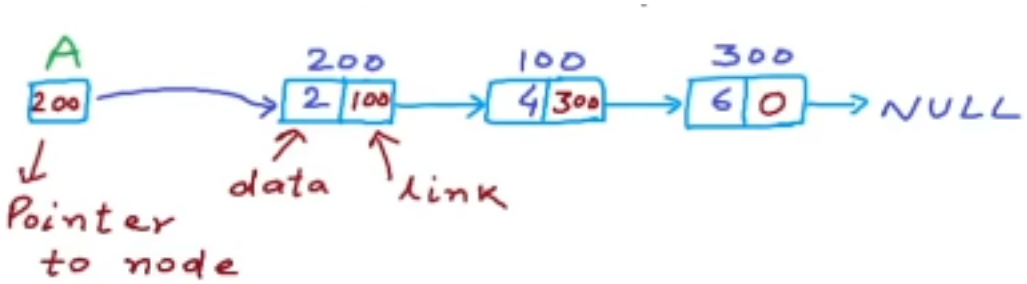
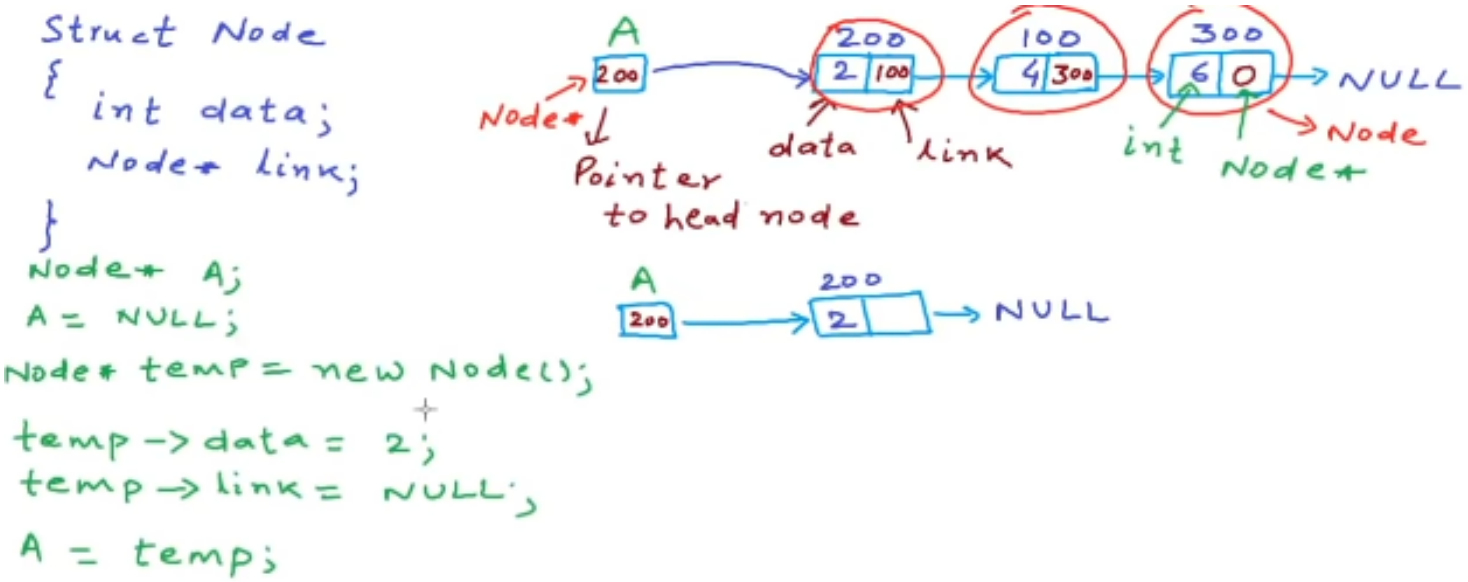
Linked List

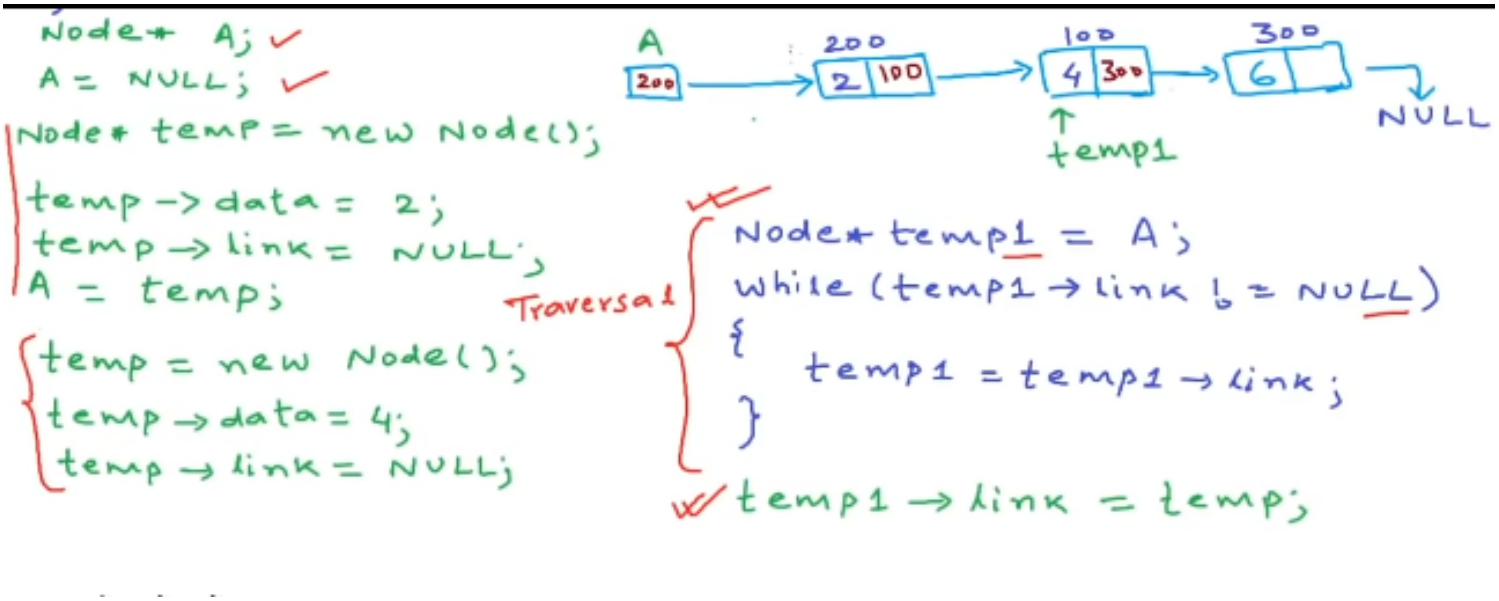


-A linked list node consist of data field and a link field.
-The data link consist of the actual data whereas the link field will consist of the address of the next node.
-Link field is a pointer variable pointing to the address of next node.
-The link field of the last node contains 0 which means it doesn't point to any memory location or it points to NULL.
-We store the address of first node in a pointer variable called Head.
-The address of first node can be used to acces all other values in the linked list.

Implementation in C++(Theory)



-We first create a Node structure containing data field and link field.
-We then create a head pointer(A).
-A initially points to NULL.
-We create a node from the structure Node which will return a void pointer, which is then typecasted and stored in pointer temp.
-Now temp points to the first node and using it we then initialize the data and link field of the first node.
-The link field of this node is then initialized to NULL making it the last node.



-We then use the temp variable again and create another node.
-We the initialize the data and link fields.
-Link field wiill be initialized to NULL.
-Now the last node in the linked list must point to this node.
-To perform this we write a general traversal code that treaverses to the last node of the existing linked list.
-We create a new pointer variable temp1 and point to the head pointer.
-Using this pointer we traverse to the end of the list using the above logic.
-After it reaches the end of the list, we then change the value of the link field containing NULL to the address of the newly created node.