IT DS 201 LAB

SUBMITTED BY ADITYA SINGH 2K19/EP/005

Program : Write a program to implement the Linked List Data structure and insert a new node at the beginning, and at a given position.

CODE

```
#include<iostream>
using namespace std;
class Node {
    public:
        int data;
        Node*next;

        Node(int d){
        data = d;
        next = NULL;
    }
};

Node *insertAthead(Node *head, int x){
    Node *temp = new Node(x);
    temp->next = head;
    return temp;
}
```

```
Node *insertAtPos(Node* head, int pos, int x){
    Node* temp = new Node(x);
    if(pos == 1){
        temp->next = head;
        return temp;
    }
    Node *curr = head;
    for(int i=1; i<pos-1 && curr!=NULL; i++){
        curr = curr->next;
    }
    if(curr==NULL){
        return head;
    }
    temp->next = curr->next;
    curr->next = temp;
    return head;
}

void printLL(Node *head){
    while(head!=NULL){
        cout<<head->data<<"->";
        head = head->next;
    }
    cout<<endl;
}
```

ALGORITHM

1. Traverse the Linked list upto position-1 nodes.
2. Once all the position-1 nodes are traversed, allocate memory and the given data to the new node.
3. Point the next pointer of the new node to the next of the current node.
4. Point the next pointer of the current node to the new node.

INPUT/OUTPUT

```
int main(){
    Node *head = NULL;
    head = insertAthead(head,3);
    head = insertAthead(head,5);
    head = insertAthead(head,4);
    head = insertAthead(head,9);
    printLL(head);
    head = insertAtPos(head,3,6);
    printLL(head);
    head = insertAtPos(head,5,2);
    printLL(head);
    head = insertAtPos(head,1,10);
    printLL(head);
    head = insertAtPos(head,7,7);
    printLL(head);
}
```

```
9->4->5->3->
9->4->6->5->3->
9->4->6->5->2->3->
10->9->4->6->5->2->3->
10->9->4->6->5->2->7->3->
[Finished in 553ms]
```

END