DSA_College\doubly_linkedList.cpp

```cpp
#include<iostream>
using namespace std;

class Node
{
    public :
    int data ;
    Node* prev ;
    Node* next ;

    Node(int data)
    {
        this->data = data ;
        this->prev = NULL ;
        this->next = NULL ;
    }

    ~Node()
    {
        int value = this->data ;
        while(this->next != NULL)
        {
            delete next ;
            this->next = NULL ;
        }
        cout<<"memory is free for node with data = "<<value<<endl;
    }
};

int getLength(Node* &head)
{
    Node* temp = head ;
    int len =0 ;
    while(temp != NULL)
    {
        len++ ;
        temp = temp->next ;
    }
    return len ;
}

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

```cpp

void insertAtHead(Node* &head, Node* &tail, int d)
{
    if(head == NULL && tail == NULL)
    {
        Node* temp = new Node(d) ;
        head =temp ;
        tail = temp ;
    }
    else{
        Node* temp = new Node(d) ;
        temp->next = head ;
        head->prev = temp ;
        head = temp ;
    }
}

void insertAtTail(Node* &tail, Node* &head, int d)
{
    if(head == NULL && tail == NULL)
    {
        Node* temp = new Node(d) ;
        head =temp ;
        tail = temp ;
    }
    else
    {
        Node* temp = new Node(d) ;
        tail->next = temp;
        temp->prev = tail ;
        tail = temp ;
    }
}

void insertAtPosition(Node* &head, Node* &tail, int d, int pos)
{
    if(pos == 1)
    {
        insertAtHead(head,tail,d) ;
        return ;
    }
    Node* temp = head ;
    int cnt = 1 ;
    while(cnt < pos-1)
    {
        temp = temp->next ;
        cnt++ ;
    }
    if(temp->next == NULL)
    {
        insertAtTail(tail,head,d) ;
        return ;
    }
    Node* newnode = new Node(d) ;
```

```
    newnode->next = temp->next  ;
    temp->next->prev = newnode ;
    temp->next = newnode ;
    newnode->prev = temp ;
}

void deleteNode(Node* &head, int pos)
{
    if(pos == 1)
    {
        Node* temp = head ;
        temp->next->prev = NULL ;
        head = head->next ;
        temp->next = NULL ;
        delete temp ;
    }
    else{
         Node* prev = NULL ;
    Node* curr = head ;
    int cnt = 1 ;
    while(cnt < pos)
    {
        prev = curr ;
        curr = curr->next ;
        cnt++ ;
    }
    curr->prev = NULL ;
    prev->next = curr->next ;
    curr->next->prev = prev ;
    curr->next = NULL;
    delete curr ;
    }
}

int main()
{
    Node* node1 = new Node(10) ;  // node1 naam ka object bnaya jisme dat ki value 10 hai and prev and next pointers shuru mai null ko point kr rhe
    Node* head= node1 ; // head naam se ek object bnaya jisme node1 object copy krdiya ya fir node1 object se initialize kr diya
    Node* tail = node1 ;
    print(head) ; // 10

    insertAtHead(head,tail,11) ; // 11 10
    print(head) ;

    insertAtTail(tail,head,4) ; // 11 10 4
    print(head) ;

    insertAtPosition(head,tail,24,2) ; // 11 24 10 4
    print(head) ;

    deleteNode(head,1) ;
    print(head) ; // 24 10 4
```

```
158
159     return 0 ;
160 }
```