**DOUBLY LINKED LIST**

#include<stdlib.h>

#include<stdbool.h>

#include <stdio.h>

struct node{

int data;

struct node\* prev;

struct node\* next;

};

struct node\* insert(struct node\* head, int x){

struct node\* newnode = (struct node\*) malloc (sizeof(struct node));

if(head==NULL) { // first node

newnode->next = NULL;

newnode->prev=NULL;

newnode->data=x;

head=newnode;

return head;

}

newnode->data = x; // subsequent nodes

newnode->prev = NULL;

newnode->next = head;

head = newnode;

return head;

}

struct node\* Delete(struct node\* head, int x){

if(head->data==x) {

struct node\* temp = head;

head = head->next;

free(temp);

return head;

}

else {

struct node\* temp = head;

while(temp->next->data!=x){

temp = temp->next;

}

struct node\* ptr=temp;

ptr = ptr->next;

temp->next = ptr->next;

free(ptr);

return head;

}

}

bool nodefound (struct node\* head, int x) {

while(head!=NULL) {

if((head->data)==x) {

return true;

break;

}

head=head->next;

}

return false;

}

void display(struct node\* head) {

while(head!=NULL) {

printf("%d\t",head->data);

head=head->next;

}

}

int main() {

struct node\* head = NULL;

int a, choice=1, n;

while(choice){

printf("Enter value of node to be inserted : ");

scanf("%d",&a);

printf("\nInserting %d...\n",a);

head=insert(head,a);

display(head);

printf("\nWant to insert more ? (Press 1 else Press 0) : ");

scanf("%d",&choice);

}

printf("\nEnter node value to be deleted : ");

scanf("%d",&n);

if(nodefound(head,n)){

head = Delete(head,n);

printf("\nDeleting %d\n",n);

printf("\nUpdated linked list : \n");

display(head);

}

else

printf("\nNode not found!!\n");

return 0;

}