

double link list.c X

```
#include<stdio.h>

#include<stdlib.h>
struct node
{
    int data;
    struct node *next;
    struct node *prev;
};
struct node *head=NULL;
void insert_beg()
{
    struct node *new_node;
    new_node=(struct node*)malloc(sizeof(struct node));
    printf("Enter the item\n");
    scanf("%d",&new_node->data);
    new_node->next=NULL;
    new_node->prev=NULL;

    if(head==NULL)
    {
        head=new_node;
    }
    else
    {
        new_node->next=head;
        head->prev=new_node;
        head=new_node;
    }
}

void insert_end()
{
    struct node *new_node,*temp;
    new_node=(struct node*)malloc(sizeof(struct node));
```

```
void insert_end()
```

```
{
    struct node *new_node, *temp;
    new_node=(struct node*)malloc(sizeof(struct node));
    printf("Enter the item\n");
    scanf("%d",&new_node->data);
    new_node->next=NULL;
    new_node->prev=NULL;
    if(head==NULL)
    {
        head=new_node;
    }
    else
    {
        temp=head;
        while(temp->next!=NULL)
            temp=temp->next;
        temp->next=new_node;
        new_node->prev=temp;
    }
}
```

```
void insert_between()
```

```
{
    int listele;
    struct node *new_node, *temp;
    printf("Enter the element in the list\n");
    scanf("%d",&listele);
    new_node=(struct node*)malloc(sizeof(struct node));
    printf("Enter the new node data\n");
    scanf("%d",&new_node->data);
    new_node->next=NULL;
    new_node->prev=NULL;
    if(head==NULL)
    {

```



double link list.c X

```
if(head==NULL)
{
    printf("Empty list\n"); return;
}
temp=head;
while(temp->data!=listele)
{
    temp=temp->next;
    if(temp==NULL)
    {
        printf("Element is not in the list");
        return;
    }
}
new_node->next=temp->next;
temp->next=new_node;
new_node->prev=temp;
new_node->next->prev=new_node;
```

void del()

```
{
    struct node *temp;
    int ele;
    if(head==NULL)
    {
        printf("Empty List \n");
        return;
    }
    printf("Enter the element to be deleted\n");
    scanf("%d",&ele);
    temp=head;
    while(temp->data!=ele)
    {
        temp=temp->next;
        if(temp==NULL)
```



```
temp=temp->next;
if(temp==NULL)
{
    printf("Element is not in the list\n");
    break;
}
}
if(temp==head)
{
    head=head->next;
}
else if(temp->next==NULL)
{
    temp=temp->prev;
    temp->next=NULL;
}

else
{
    temp->prev->next=temp->next;
    temp->next->prev=temp->prev;
}
}

void display()
{
    struct node *temp;
    temp=head;
    while(temp!=NULL)
    {
        printf("%d\t",temp->data);
        temp=temp->next;
    }
    printf("\n");
}
```



```
temp=head;
while(temp!=NULL)
{
    printf("%d\t",temp->data);
    temp=temp->next;
}
printf("\n");
```

```
int main()
{
    int choice;

    while(1)
    {
        printf(" 1. Insert at the beg \n");
        printf(" 2. Insert at the end \n");
        printf(" 3. Insert after a given node\n");
        printf(" 4. Delete \n");
        printf(" 5. Display\n");
        printf(" 6. Exit\n");
        printf("Enter your choice\n");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1: insert_beg(); break;
            case 2: insert_end();break;
            case 3: insert_between();break;
            case 4: del(); break;
            case 5: display(); break;
            case 6: exit(0);
        }
    }
}
```