```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

struct node{
    int info;
    struct node *next;
    struct node *prev;


};
struct node *first = NULL;
struct node *last = NULL;

void insert(int );
int del (int);
void print();
struct node *search(int);


void main(){
    int n1,n2,ch;
    struct node *loc;
 while(1){
        printf("SELECT YOUR CHOICE
\n1.INSERT\n2.DELETE\n3.SEARCH\n4.DISPLAY\n6.EXIT\n\n");
        scanf("%d",&ch);
        switch (ch){
            case 1 :
            printf("Enter the element -\n");
            scanf("%d",&n1);
            insert(n1);
            break;
case 2:
printf("Enter the element to be del");
scanf("%d",&n1);
n2=del(n1);
if(n2==-1)
printf("%d is not found",n1);
```

```c
else
printf("%d deleted",n2);
break;

case 3:
printf("Enter the element-");
scanf("%d",&n1);
loc = search(n1);

if(loc==NULL){
    printf("not found");
}
if(loc==last){
    printf("it is the last element");
}
else
printf("element %d should be present before the element %d",n1,(loc-
>next)->info);
break;


    case 4 :
    print();
    break;

    default:
    exit(0);
    }

 }
}
void insert(int v){
    struct node *ptr = (struct node*)malloc(sizeof(struct node));
    ptr->info = v;

    if(first ==NULL){
        first = last = ptr;
```

```c
        ptr->next=NULL;
        ptr->prev=NULL;
    }
    else {
        last->next=ptr;
        ptr->next=NULL;
        ptr->prev=last;
        last = ptr;
    }
}

int del(int v) {
    struct node *loc, *temp;
    int i;
    i = v;
    loc = search(i);

    if (loc == NULL) {
        return -1;
    }

    if (loc == first) {
        if (first == last) {
            first = last = NULL;
        } else {
            first->next->prev = NULL;
            first = first->next;
        }
        return (v);
    }

    for (temp = first; temp->next != loc; temp = temp->next);

    if (loc == last) {
        last = temp;
        temp->next = NULL;
    } else {
```

```c
        temp->next = loc->next;
        loc->next->prev = temp;
    }

    return (loc->info);
}

struct node *search(int v) {
    struct node *ptr;

    if (first == NULL) {
        return NULL;
    }

    for (ptr = first; ptr != NULL; ptr = ptr->next) {
        if (ptr->info == v) {
            return ptr;
        }
    }

    return NULL;
}




void print(){
    struct node *ptr;
    printf("'The elements are -");
    if(first==last){
        printf("\t%d",first->info);
        return ;
    }
 for(ptr=first;ptr!=last;ptr=ptr->next)
    printf("\t%d",ptr->info);
    printf("\t%d",last->info);
}
```