

```
1 #include<stdio.h>
2 #include<conio.h>
3 #include<stdlib.h>
4 struct node{
5     int data;
6     struct node *next;
7     struct node *prev;
8 };
9 struct node *head;
10 void insertbeginning(int item)
11 {
12     struct node *ptr = (struct node *)malloc(sizeof(struct node));
13     if(head==NULL)
14     {
15         ptr->next = NULL;
16         ptr->prev=NULL;
17         ptr->data=item;
18         head=ptr;
19     }
20     else
21     {
22         ptr->data=item;
23         ptr->prev=NULL;
24         ptr->next = head;
25         head->prev=ptr;
26         head=ptr;
27     }
28 }
29
30 void delete_specified( )
31 {
32     struct node *ptr, *temp;
33     int val;
34     printf("Enter the value");
35     scanf("%d",&val);
36     temp = head;
37     while(temp -> data != val)
38         temp = temp -> next;
39     if(temp -> next == NULL)
```

```
8 void delete_specified( )
9 {
10     struct node *ptr, *temp;
11     int val;
12     printf("Enter the value");
13     scanf("%d",&val);
14     temp = head;
15     while(temp -> data != val)
16         temp = temp -> next;
17     if(temp -> next == NULL)
18     {
19         printf("\nCan't delete\n");
20     }
21     else if(temp -> next -> next == NULL)
22     {
23         temp -> next = NULL;
24         printf("Node Deleted\n");
25     }
26
27     else{
28         ptr = temp -> next;
29         temp -> next = ptr -> next;
30         ptr -> next -> prev = temp;
31         free(ptr);
32         printf("Node Deleted\n");
33     }
34 }
35
36 void display()
37 {
38     struct node *ptr;
39     ptr=head;
40     if(ptr==NULL)
41     {
42         printf("empty ");
43     }
44     else
45     {
46         while(ptr!=NULL)
47         {
48             printf("%d ",ptr->data);
49             ptr=ptr -> next;
50         }
51         printf("\n");
52     }
```



```
47
48 void display()
49 {
50     struct node *ptr;
51     ptr=head;
52     if(ptr==NULL)
53     {
54         printf("empty ");
55     }
56     else
57     {
58         while(ptr!=NULL)
59         {
60             printf("%d ",ptr->data);
61             ptr=ptr -> next;
62         }
63         printf("\n");
64     }
65 }
66
67 void main()
68 {
69     int op=0;int a;
70     printf("\n 1.Insert to the left\n2.Delete\n3.Display\n ");
71
72     while(op!=4)
73     {
74         printf("\nenter your choice : ");
75         scanf("%d",&op);
76         switch(op)
77         {
78             case 1:printf("enter value to be inserted ");
79                     scanf("%d",&a);
80                     insertbeginning(a);
81                     break;
82             case 2:delete_specified();
83                     break;
```



```

5 }
5 else
7 {
3     while(ptr!=NULL)
9 {
9     printf("%d ",ptr->data);
1     ptr=ptr -> next;
2     }
3     printf("\n");
4 }
5 }
5
7 void main()
8 {
9     int op=0;int a;
9     printf("\n 1.Insert to the left\n2.Delete\n3.Display\n ");
1
2     while(op!=4)
3     {
4         printf("\nenter your choice : ");
5         scanf("%d",&op);
6         switch(op)
7         {
8             case 1:printf("enter value to be inserted ");
9                     scanf("%d",&a);
10                    insertbeginning(a);
11                    break;
12             case 2:delete_specified();
13                    break;
14             case 3: display();
15                    break;
16
17         }
18     }
19 }

```



1.Insert to the left
2.Delete
3.Display

enter your choice : 1
enter value to be inserted 10

enter your choice : 1
enter value to be inserted 11

enter your choice : 1
enter value to be inserted 12

enter your choice : 1
enter value to be inserted 13

enter your choice : 1
enter value to be inserted 14

enter your choice : 3
14 13 12 11 10

enter your choice : 2
Enter the value 11
Node Deleted

enter your choice : 3
14 13 12 11

enter your choice :
2
Enter the value 14
Node Deleted

