

main.c

```
1  #include<stdio.h>
2
3  #include<stdlib.h>
4  struct node
5  {
6      int data;
7      struct node *next;
8      struct node *prev;
9  };
10 struct node *head=NULL;
11 void insert_beg()
12 {
13     struct node *new_node;
14     new_node=(struct node*)malloc(sizeof(struct node));
15     printf("Enter the item\n");
16     scanf("%d",&new_node->data);
17     new_node->next=NULL;
18     new_node->prev=NULL;
19
20     if(head==NULL)
21     {
22         head=new_node;
23     }
24     else
25     {
26         new_node->next=head;
27         head->prev=new_node;
28         head=new_node;
29     }
```

```

29 /
30
31 }
32
33 void insert_end()
34 {
35     struct node *new_node,*temp;
36     new_node=(struct node*)malloc(sizeof(struct node));
37     printf("Enter the item\n");
38     scanf("%d",&new_node->data);
39     new_node->next=NULL;
40     new_node->prev=NULL;
41     if(head==NULL)
42     {
43         head=new_node;
44     }
45     else
46     {
47         temp=head;
48         while(temp->next!=NULL)
49             temp=temp->next;
50         temp->next=new_node;
51         new_node->prev=temp;
52     }
53 }
54
55 }
56 void insert_between()
57 {
58     int listele;
59     struct node *new_node,*temp;

```

main.c

```
61 scanf("%d",&listele);
62 new_node=(struct node*)malloc(sizeof(struct node));
63 printf("Enter the new node data\n");
64 scanf("%d",&new_node->data);
65 new_node->next=NULL;
66 new_node->prev=NULL;
67 if(head==NULL)
68 {
69     printf("Empty list\n"); return;
70 }
71 temp=head;
72 while(temp->data!=listele)
73 {
74     temp=temp->next;
75     if(temp==NULL)
76     {
77         printf("Element is not in the list");
78         return;
79     }
80 }
81 new_node->next=temp->next;
82 temp->next=new_node;
83 new_node->prev=temp;
84 new_node->next->prev=new_node;
85 }
86 void del()
87 {
88     struct node *temp;
89     int ele;
90     if(head==NULL)
91     {
```

main.c

```
89 int ele;
90 if(head==NULL)
91 {
92     printf("Empty List \n");
93     return;
94 }
95 printf("Enter the element to be deleted\n");
96 scanf("%d",&ele);
97 temp=head;
98 while(temp->data!=ele)
99 {
100     temp=temp->next;
101     if(temp==NULL)
102     {
103         printf("Element is not in the list\n");
104         break;
105     }
106 }
107 if(temp==head)
108 {
109     head=head->next;
110 }
111 else if(temp->next==NULL)
112 {
113     temp=temp->prev;
114     temp->next=NULL;
115 }
116
117 else
118 {
```

```

121     }
122 }
123 void display()
124 {
125     struct node *temp;
126     temp=head;
127     while(temp!=NULL)
128     {
129         printf("%d\t",temp->data);
130         temp=temp->next;
131     }
132     printf("\n");
133 }
134
135
136 int main()
137 {
138     int choice;
139
140     while(1)
141     {
142         printf(" 1. Insert at the beg \n");
143         printf(" 2. Insert at the end \n");
144         printf(" 3. Insert after a given node\n");
145         printf(" 4. Delete \n");
146         printf(" 5. Display\n");
147         printf(" 6. Exit\n");
148         printf("Enter your choice\n");
149         scanf("%d",&choice);
150         switch(choice)
151         {

```

```

1      }
2      printf("\n");
3  }
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36 int main()
37 {
38     int choice;
39
40     while(1)
41     {
42         printf(" 1. Insert at the beg \n");
43         printf(" 2. Insert at the end \n");
44         printf(" 3. Insert after a given node\n");
45         printf(" 4. Delete \n");
46         printf(" 5. Display\n");
47         printf(" 6. Exit\n");
48         printf("Enter your choice\n");
49         scanf("%d",&choice);
50         switch(choice)
51         {
52             case 1: insert_beg(); break;
53             case 2: insert_end();break;
54             case 3: insert_between();break;
55             case 4: del(); break;
56             case 5: display(); break;
57             case 6: exit(0);
58         }
59     }
60 }

```


- 1. Insert at the beg
- 2. Insert at the end
- 3. Insert after a given node
- 4. Delete
- 5. Display
- 6. Exit

Enter your choice

1

Enter the item

2

- 1. Insert at the beg
- 2. Insert at the end
- 3. Insert after a given node
- 4. Delete
- 5. Display
- 6. Exit

Enter your choice

2

Enter the item

1

- 1. Insert at the beg
- 2. Insert at the end
- 3. Insert after a given node

