

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  void create();
4  void display();
5  void delfun(int);
6  void insert_front();
7  struct node
8  {
9      int data;
10     struct node *next;
11 };
12 struct node *head=NULL;
13 int main(int argc, char **argv)
14 {
15     int choice,ele;
16     int i=0;
17     do
18     {
19         printf("\n1. Create \n2. Display \n3. Delete \n4. Insertfront\n");
20         printf("\nEnter your choice : ");
21         scanf("%d",&choice);
22         switch(choice)
23         {
24             case 1: create(); break;
25             case 2: display();break;
26             case 3: printf("Enter the element to be deleted\n");
27                     scanf("%d",&ele);
28                     delfun(ele); break;
29             case 4: insert_front();
30                     break;
31         }

```

```

32
33     }while(i==0);
34 }
35
36 void create()
37 {
38     struct node *newnode,*temp;
39     int item;
40     newnode =(struct node *) malloc (sizeof(struct node));
41     printf("Enter the data : ");
42     scanf("%d",&item);
43     newnode->data=item;
44     if (head==NULL)
45     {
46         newnode->next=NULL;
47         head=newnode;
48         printf("Node created\n");
49     }
50     else
51     {
52         temp=head;
53         while(temp->next!=NULL)
54         {
55             temp=temp->next;
56         }
57         temp->next=newnode;
58         newnode->next=NULL;
59         printf("Node created\n");
60     }
61 }
62
63 void display()
64 {

```


main.c

```
62
63 void display()
64 {
65     struct node *ptr=NULL;
66     ptr=head;
67
68     if(ptr==NULL)
69     {
70         printf("Nothing to print\n");
71     }
72     else
73     {
74         while(ptr!=NULL)
75         {
76
77             printf("%d ",ptr->data);
78             ptr=ptr->next;
79             }printf("\n");
80     }printf("-----\n");
81
82 }
83
84 void delfun(int ele)
85 {
86     struct node *temp,*del=NULL;
87
88     if (head == NULL)
89     {
90         printf("Empty List. Can't delete\n");return;
91     }
92     temp=head;
93     if(head->data==ele)
```

```

91     }
92     temp=head;
93     if(head->data==ele)
94     {
95         head=head->next;
96         return;
97     }
98     while (temp->next!=NULL)
99     {
100         if(temp->next->data==ele)
101         {
102             del=temp->next;
103             if(del->next==NULL)
104                 temp->next=NULL;
105             else
106                 temp->next=del->next;
107         }
108
109         else
110             temp=temp->next;
111     }
112     if(del==NULL)
113     {
114         printf("Element not found in the list\n");return;
115     }
116
117
118 }
119
120 void insert_front()
121 {
122     struct node *newnode;

```


main.c

```
117
118 }
119
120 void insert_front()
121 {
122     struct node *newnode;
123     int ele;
124     printf("Enter the element : ");
125     scanf("%d",&ele);
126
127     newnode=(struct node*)malloc(sizeof(struct node));
128
129     newnode->data =ele;
130     newnode->next=head;
131     head=newnode;
132
133
134 }
```