

```

1  //////////////////////////////////CIRCULAR LINKED LIST
OPERATIONS////////////////////////////////////
2  #include<stdio.h>
3  #include<stdlib.h>
4
5  struct node
6  {
7      int data;
8      struct node *next;
9  };
10
11 struct node *head=NULL;
12
13 void InsertInB(int Value) // Insert in the beginning of circular LL
14 {
15     struct node *p=(struct node *)malloc(sizeof(struct node));
16     p->data=Value;
17     if(head==NULL)
18     {
19         p->next=p;
20         head=p;
21     }
22     else
23     {
24         struct node *q=head;
25         while(q->next!=head)
26         {
27             q=q->next;
28         }
29         q->next=p;
30         p->next=head;
31         head=p;
32     }
33 }
34
35
36 void InsertInE(int Value) // Insert in the End of circular LL.
37 {
38     struct node *p=(struct node *)malloc(sizeof(struct node));
39     p->data=Value;
40     if(head==NULL)
41     {
42         p->next=p;
43         head=p;
44     }
45     else
46     {
47         struct node *q=head;
48         p->next=q;
49         do
50         {
51             q=q->next;
52         }while(q->next!=head);
53
54         q->next=p;
55     }
56 }
57
58 void display()
59 {
60     struct node *display=head;
61     do
62     {
63         printf("%d",display->data);
64         printf("---->");
65         display=display->next;

```

```
66         }while(display!=head);
67     }
68
69     void main()
70     {
71
72         InsertInB(30);
73         InsertInB(20);
74         InsertInB(10);
75         display();
76         InsertInE(45);
77         InsertInE(34);
78         printf("\n\n");
79         display();
80     }
```