```c
#include<stdio.h>
#define MAX 10
int deque[MAX];
int left = -1, right = -1;
void input_deque(void);
void output_deque(void);
void insert_left(void);
void insert_right(void);
void delete_left(void);
void delete_right(void);
void display(void);
       int main() {
       int option;
       printf("\n MAIN MENU");
       printf("\n 1. Input Restricted Deque");
       printf("\n 2. Output Restricted Deque");
       printf("\n Enter your option: ");
       scanf("%d",&option);
       switch(option)
       {
       case 1:
       input_deque();
       break;
       case 2:
       output_deque();
       break;
       }
       return 0;
       }
```

```
30
31        // Input Deque
32        void input_deque() {
33        int option;
34        do {
35        printf("\n Input Restricted Deque");
36        printf("\n 1. Insert at right");
37        printf("\n 2. Insert at left");
38        printf("\n 3. Delete from left");
39        printf("\n 4. Display");
40        printf("\n 5. Quit");
41        printf("\n Enter your option: ");
42        scanf("%d",&option);
43        switch (option) {
44        case 1:
45        insert_right();
46        break;
47        case 2:
48        insert_left();
49        break;
50        case 3:
51        delete_left();
52        break;
53        case 4:
54        display();
55        break;
56        }
57            } while(option!=5);
58        }
```

```
59
60        // Output Deque
61        void output_deque() {
62        int option;
63        do {
64        printf("\n Output Restricted Deque");
65        printf("\n 1. Insert at right");
66        printf("\n 2. Insert at left");
67        printf("\n 3. Delete from left");
68        printf("\n 4. Display");
69        printf("\n 5. Quit");
70        printf("\n Enter your option: ");
71        scanf("%d",&option);
72        switch (option) {
73        case 1:
74        insert_right();
75        break;
76        case 2:
77        insert_left();
78        break;
79        case 3:
80        delete_left();
81        break;
82        case 4:
83        display();
84        break;
85        }
86             } while(option!=5);
87        }
88
89        // Insert Right
90        void insert_right() {
91        int value;
92        printf("\n Enter the value to be added: ");
93        scanf("%d",&value);
94        if ((left == 0 && right == MAX-1) || (left == right + 1))
95        {
96             printf("\n Overflow");
97             return;
98        }
99        if (left == -1) // Considering queue is initially empty
100       {
101       left = 0;
102       right = 0;
103       }
104       else
105       {
106            if(right == MAX-1) //  right is at last position of queue
107            right = 0;
108            else
109            right = right + 1;
110       }
111       deque[right] = value;
112       }
```

```
// Insert Left
void insert_left() {
int value;
printf("\n Enter the value to be added: ");
scanf("%d",&value);
if ((left == 0 && right == MAX - 1) && (left = right + 1))
{
      printf("\n Overflow");
      return;
}
if (left == -1) // Considering the queue is initially empty
{
left = 0;
right = 0;
}
else
{
  if (left == 0)
  left = MAX - 1;
  else
  left = left - 1;
}
deque[left] = value;
}
```

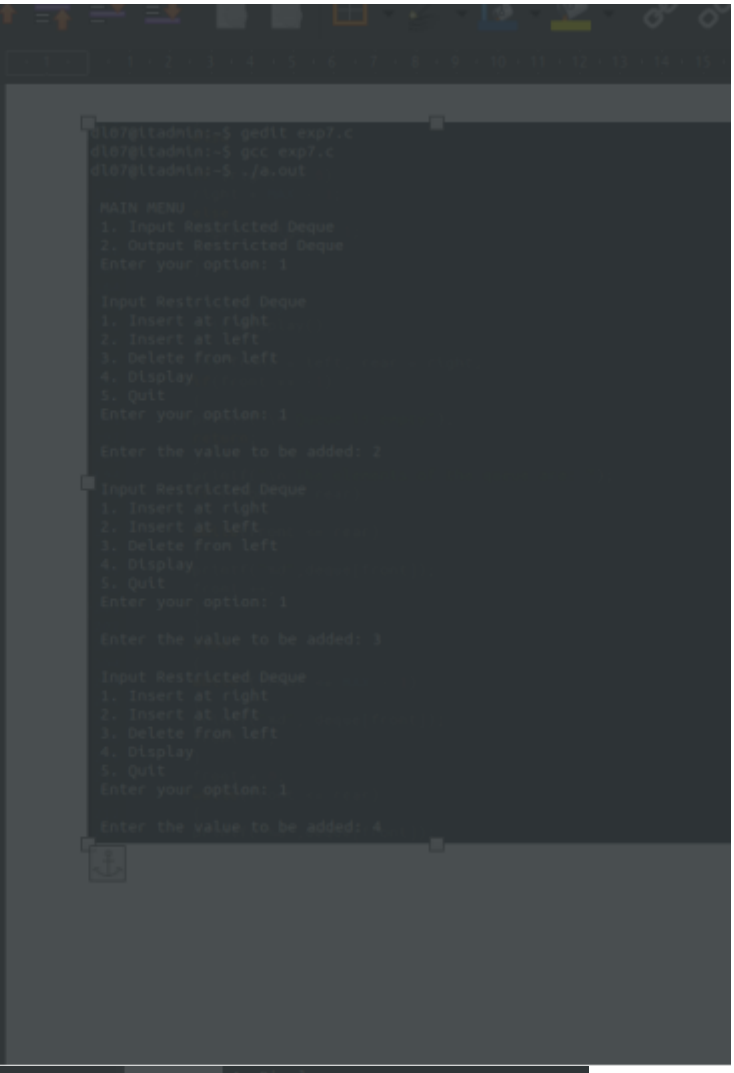
```
// Delete Left
void delete_left()
{
      if (left == -1)
      {
         printf("\n Underflow");
         return;
      }
      printf(" \n The deleted element is : %d",deque[left]);
      if (left == right) // Queue consists of only one element
      {
      left = -1;
      right = -1;
      }
      else
      {
      if (left == MAX - 1)
      left = 0;
      else
      left = left + 1;
      }
}
```

```
// Delete Right
void delete_right()
{
if (left == -1)
{
printf("\n Underflow");
return;
}
printf("\n The element deleted is : %d",deque[right]);
if (left == right) // Queue consist of only one element
{
left = -1;
right = -1;
}
else
{
if (right == 0)
right = MAX - 1;
else
right = right - 1;
}
}
```

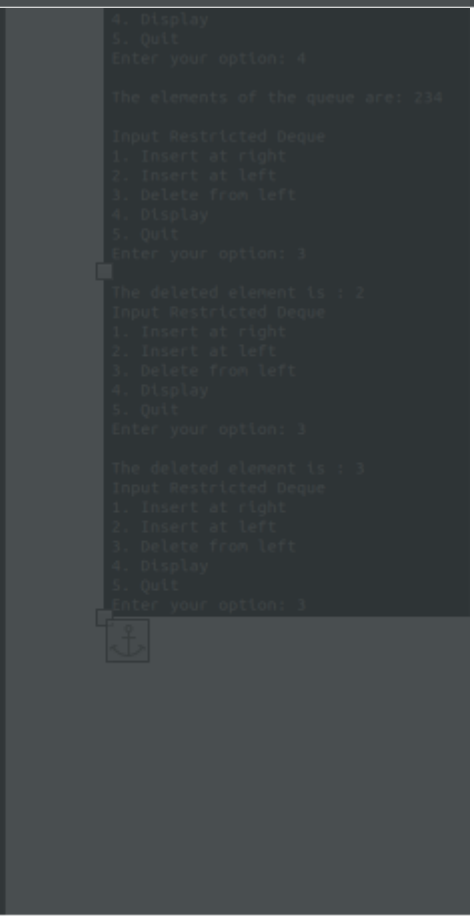
```
184
185         // Display
186         void display()
187         {
188         int front = left, rear = right;
189         if(front == -1)
190         {
191         printf("\n Queue is empty");
192         return;
193         }
194         printf("\n The elements of the queue are: ");
195         if (front <= rear)
196         {
197         while(front <= rear)
198         {
199         printf("%d",deque[front]);
200         front ++;
201         }
202         }
203         else
204         {
205         while (front <= MAX - 1)
206         {
207         printf("%d", deque[front]);
208         front ++;
209         }
210         front = 0;
211         while(front <= rear)
212         {
213         printf("%d",deque[front]);
214         front++;
215         }
216         }
217          printf("\n");
218         }
```

```
dl07@itadmin:~$ gedit exp7.c
dl07@itadmin:~$ gcc exp7.c
dl07@itadmin:~$ ./a.out

 MAIN MENU
 1. Input Restricted Deque
 2. Output Restricted Deque
 Enter your option: 1

 Input Restricted Deque
 1. Insert at right
 2. Insert at left
 3. Delete from left
 4. Display
 5. Quit
 Enter your option: 1

 Enter the value to be added: 2

 Input Restricted Deque
 1. Insert at right
 2. Insert at left
 3. Delete from left
 4. Display
 5. Quit
 Enter your option: 1

 Enter the value to be added: 3

 Input Restricted Deque
 1. Insert at right
 2. Insert at left
 3. Delete from left
 4. Display
 5. Quit
 Enter your option: 1

 Enter the value to be added: 4
```

```
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 4

The elements of the queue are: 234

Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3

The deleted element is : 2
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3

The deleted element is : 3
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3
```



```
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3

The deleted element is : 4
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 4

Queue is empty
Input Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 5
```



```
dl07@itadmin:~$ gcc exp7.c
dl07@itadmin:~$ ./a.out
MAIN MENU
1. Input Restricted Deque
2. Output Restricted Deque
Enter your option: 2

Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 1

Enter the value to be added: 1

Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 1

Enter the value to be added: 2

Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 1

Enter the value to be added: 3
Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 4

The elements of the queue are: 123

Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3

The deleted element is : 1
Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option:
3

The deleted element is : 2
Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 3

The deleted element is : 3
```

```
Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 4

Queue is empty
Output Restricted Deque
1. Insert at right
2. Insert at left
3. Delete from left
4. Display
5. Quit
Enter your option: 5
```