LINEAR QUEUE

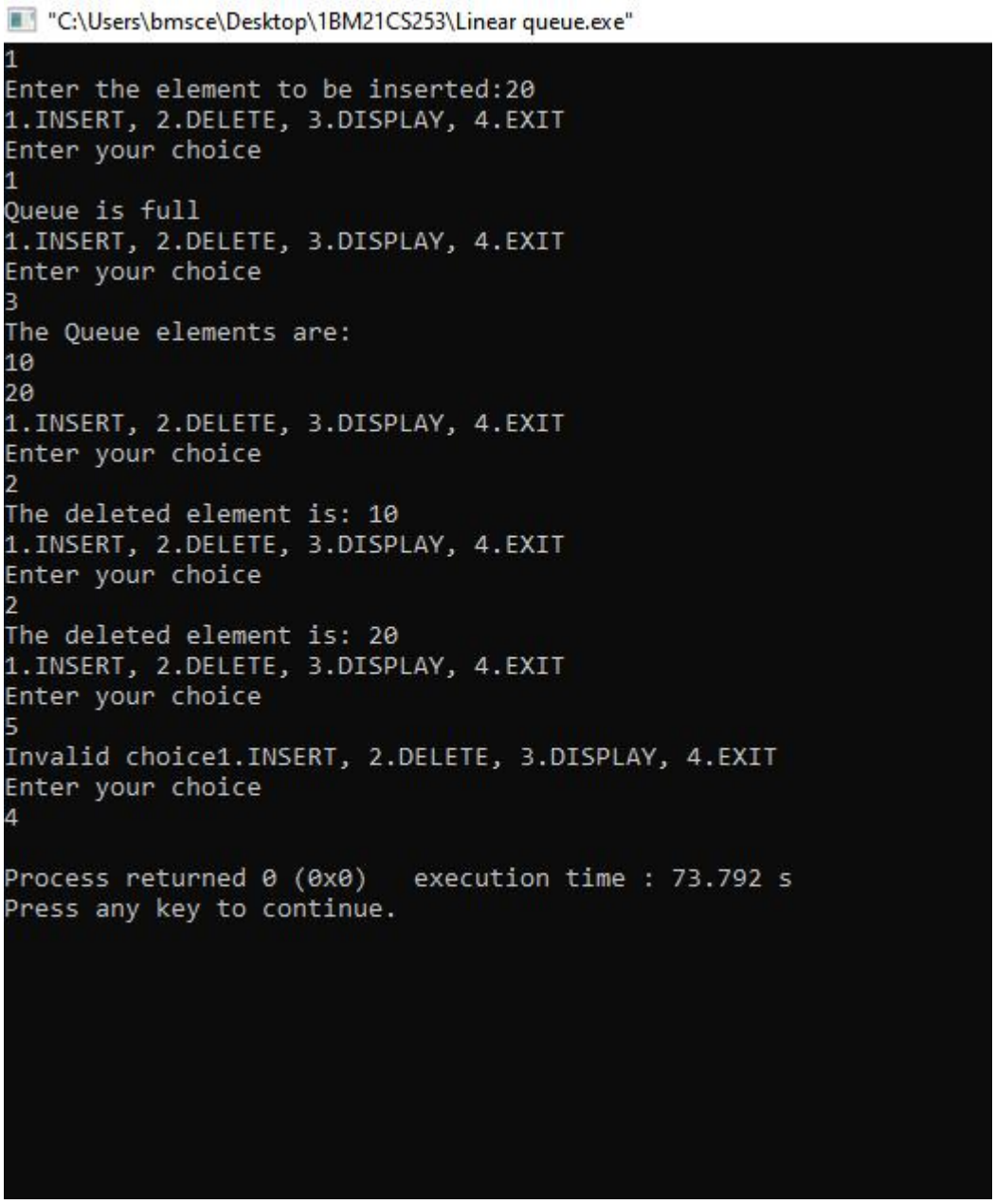
Start here | Linear queue.c | Circular queue.c

```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#define size 2
int Queue[size], front=-1,rear=-1;
void insert();
void delete();
void display();
void main()
{
int choice;
while(1){
printf("1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT\n");
printf("Enter your choice\n");
scanf("%d",&choice);
switch(choice)
{
 case 1: insert();
  break;
 case 2: delete();
  break;
 case 3: display();
         break;
 case 4: exit(0);
         break;
 default: printf("Invalid choice");
}
}
getch();
}

void insert()
{ int item;
if(rear==size-1)
printf("Queue is full\n");
else
{
printf("Enter the element to be inserted:");
scanf("%d",&item);
if((front==-1) && (rear==-1))
{
front=0;
rear=0;
Queue[rear]=item;
   }
else
{
rear++;
Queue[rear]=item;
}
```

Start here | Linear queue.c | Circular queue.c

```
36      else
37      {
38      printf("Enter the element to be inserted:");
39      scanf("%d",&item);
40      if((front==-1) && (rear==-1))
41      {
42      front=0;
43      rear=0;
44      Queue[rear]=item;
45         }
46      else
47      {
48      rear++;
49      Queue[rear]=item;
50      }
51      }
52      }
53
54      void delete()
55      { int del;
56      if((front==-1) && (rear==-1))
57      printf("Queue is empty\n");
58      int printf(const char*, ...)
59      {
60      del=Queue[front];
61      printf("The deleted element is: %d\n",del);
62      if(front==rear)
63      {
64      front=-1;
65      rear=-1;
66         }
67      else
68      front++;
69      }
70      }
71
72      void display()
73      { int i;
74      if((front==-1) && (rear==-1))
75      printf("Queue is empty\n");
76      else
77      printf("The Queue elements are:\n");
78      for(i=front; i<=rear; i++)
79      {
80      printf("%d\n",Queue[i]);
81      }
82
83      }
84
85
```


"C:\Users\bmsce\Desktop\1BM21CS253\Linear queue.exe"
1
Enter the element to be inserted:20
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
1
Queue is full
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
3
The Queue elements are:
10
20
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
2
The deleted element is: 10
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
2
The deleted element is: 20
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
5
Invalid choice1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
4
Process returned 0 (0x0) execution time : 73.792 s
Press any key to continue.

CIRCULAR QUEUE

Start here | Linear queue.c | *Circular queue.c

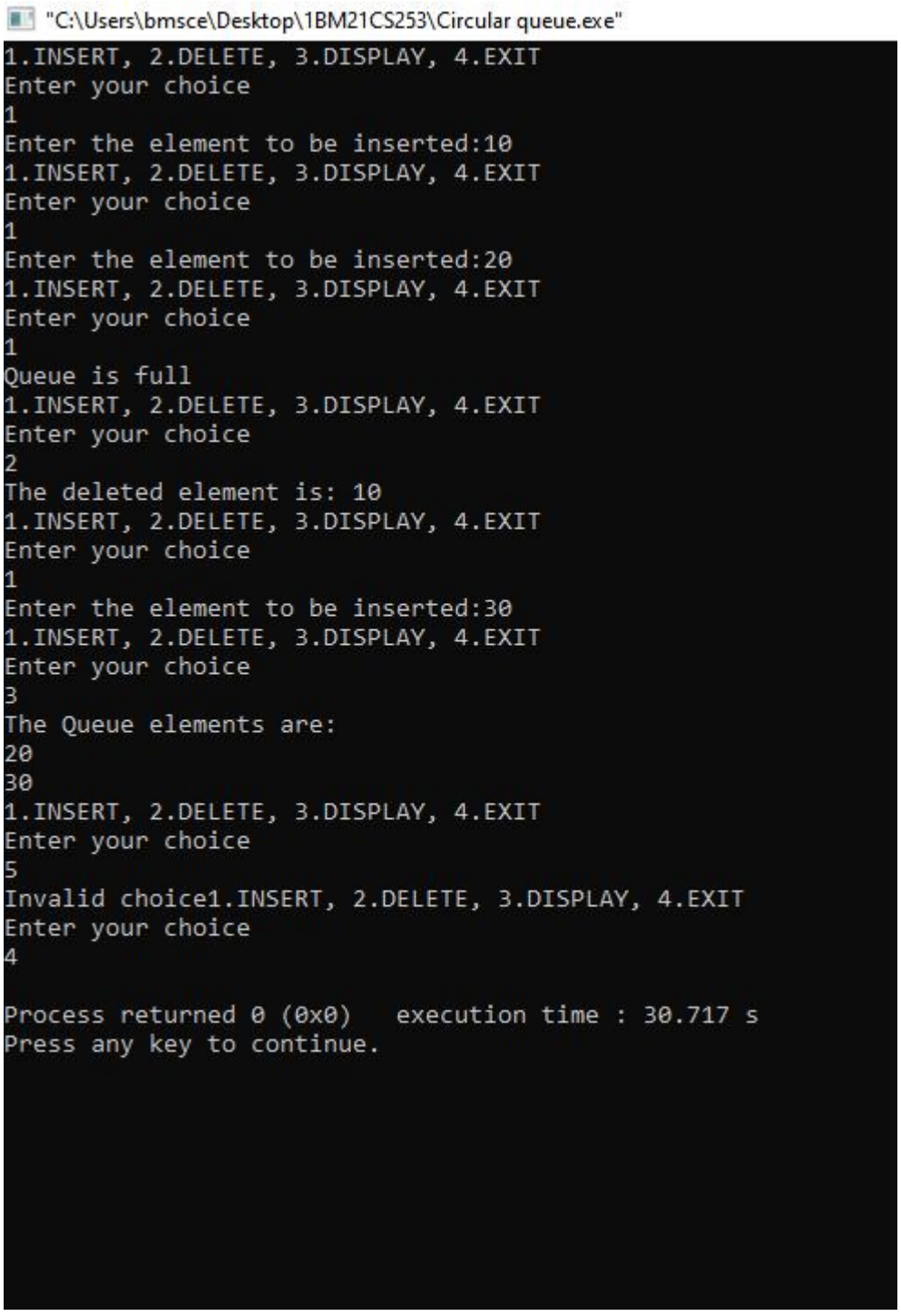
```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#define size 2
int Queue[size], front=-1,rear=-1;
void insert();
void delete();
void display();
void main()
{
int choice;
while(1){
printf("1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT\n");
printf("Enter your choice\n");
scanf("%d",&choice);
switch(choice)
{
 case 1: insert();
  break;
 case 2: delete();
  break;
 case 3: display();
         break;
 case 4: exit(0);
         break;
 default: printf("Invalid choice");
}
}
getch();
}

void insert()
{ int item;
if((front==0) && (rear==size-1) || (front==rear+1))
printf("Queue is full\n");
else
{
printf("Enter the element to be inserted:");
scanf("%d",&item);
if((front==-1) && (rear==-1))
{
front=0;
rear=0;
Queue[rear]=item;
   }
else
{
rear=(rear+1)%size;
Queue[rear]=item;
}
```

Start here | Linear queue.c | *Circular queue.c

```
47  {
48  rear=(rear+1)%size;
49  Queue[rear]=item;
50  }
51  }
52  }
53
54  void delete()
55  { int del;
56  if((front==-1) && (rear==-1))
57  printf("Queue is empty\n");
58  else
59  {
60  del=Queue[front];
61  printf("The deleted element is: %d\n",del);
62  if(front==rear)
63  {
64  front=-1;
65  rear=-1;
66      }
67  else
68  front=(front+1)%size;
69  }
70  }
71
72  void display()
73  { int i;
74  if((front==-1) && (rear==-1))
75  printf("Queue is empty\n");
76  else
77  printf("The Queue elements are:\n");
78  if(front<=rear)
79  for(i=front; i<=rear; i++)
80  {
81  printf("%d\n",Queue[i]);
82  }
83  else
84      {
85  for(i=front; i<=(size-1); i++)
86  {
87  printf("%d\n",Queue[i]);
88  }
89  for(i=0; i<=rear; i++)
90  {
91  printf("%d\n",Queue[i]);
92  }
93  }
94
95  }
96
```

OUTPUT


"C:\Users\bmsce\Desktop\1BM21CS253\Circular queue.exe"
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
1
Enter the element to be inserted:10
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
1
Enter the element to be inserted:20
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
1
Queue is full
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
2
The deleted element is: 10
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
1
Enter the element to be inserted:30
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
3
The Queue elements are:
20
30
1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
5
Invalid choice1.INSERT, 2.DELETE, 3.DISPLAY, 4.EXIT
Enter your choice
4
Process returned 0 (0x0) execution time : 30.717 s
Press any key to continue.