```
//Aim 8(b)-Implementing Circular Queue using Array
#include <stdio.h>
#define max 6
int queue[max];  // array declaration
int front=-1;
int rear=-1;
// function to insert an element in a circular queue
void enqueue(int element)
{
   if(front==-1 && rear==-1)   // condition to check queue is empty
   {
            front=0;
            rear=0;
            queue[rear]=element;
   }
   else if((rear+1)%max==front)  // condition to check queue is full
   {
            printf("Queue is overflow..");
   }
   else
   {
            rear=(rear+1)%max;      // rear is incremented
            queue[rear]=element;     // assigning a value to the queue at the rear position.
   }
}
// function to delete the element from the queue
int dequeue()
{
   if((front==-1) && (rear==-1))  // condition to check queue is empty
   {
            printf("\nQueue is underflow..");
   }
 else if(front==rear)
{
  printf("\nThe dequeued/deleted element is %d", queue[front]);
  front=-1;
  rear=-1;
}
else
{
   printf("\nThe dequeued/deleted element is %d", queue[front]);
  front=(front+1)%max;
}
}
// function to display the elements of a queue
void display()
{
   int i=front;
```

```
   if(front==-1 && rear==-1)
   {
         printf("\n Queue is empty..");
   }
   else
   {
         printf("\nElements in a Queue are :");
         while(i<=rear)
         {
            printf("%d,", queue[i]);
            i=(i+1)%max;
         }
   }
}
int main()
{
   int choice=1,x,n,i;   // variables declaration
   clrscr();
   printf("Maximum elements will be 6\n");
   while(choice<4 && choice!=0)   // while loop
   {
   printf("\nPress 1: Insert an element");
   printf("\nPress 2: Delete an element");
   printf("\nPress 3: Display the element");
   printf("\nPress 4: To Exit");
   printf("\nEnter your choice");
   scanf("%d", &choice);
   switch(choice)
    {
         case 1:
         printf("Enter total number of elements in a queue");
         scanf("%d",&n);
         for(i=1;i<=n;i++)
         {
         printf("\nEnter the element %d of Queue: ",i);
         scanf("%d", &x);
         enqueue(x);
         }
         break;
         case 2:
         dequeue();
         break;
         case 3:
         display();
    }
   }
 return 0;
}
```

OUTPUT: 8(b)

```
Maximum elements will be 6

Press 1: Insert an element
Press 2: Delete an element
Press 3: Display the element
Press 4: To Exit
Enter your choice1
Enter total number of elements in a queue4

Enter the element 1 of Queue: 10

Enter the element 2 of Queue: 20

Enter the element 3 of Queue: 30

Enter the element 4 of Queue: 40

Press 1: Insert an element
Press 2: Delete an element
Press 3: Display the element
Press 4: To Exit
Enter your choice_
```

```
DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip  0, Program:
Enter the element 2 of Queue: 20

Enter the element 3 of Queue: 30

Enter the element 4 of Queue: 40

Press 1: Insert an element
Press 2: Delete an element
Press 3: Display the element
Press 4: To Exit
Enter your choice3

Elements in a Queue are :10,20,30,40,
Press 1: Insert an element
Press 2: Delete an element
Press 3: Display the element
Press 4: To Exit
Enter your choice2

The dequeued/deleted element is 10
Press 1: Insert an element
Press 2: Delete an element
Press 3: Display the element
Press 4: To Exit
Enter your choice_
```