Program - 10

<u>Title</u>: Write a Program to Implement Circular Queue Operations by using Array

```
Code:
```

```
#include<stdio.h>
#define MAX_SIZE 10
int cq[MAX\_SIZE], front = -1, rear = -1;
void insert(int item)
  if ((front == 0 && rear == MAX_SIZE - 1) || (front == rear + 1))
    printf("Queue Overflow \n");
    return;
  else if (front == -1 && rear == -1)
    front = rear = 0;
  else if (rear == MAX_SIZE - 1 && front != 0)
    rear = 0;
  }
  else
    rear++;
  cq[rear] = item;
void delete()
  if (front == -1 && rear == -1)
    printf("Queue Underflow\n");
    return;
  int item = cq[front];
  if (front == rear)
    front = -1;
    rear = -1;
  else if (front == MAX_SIZE - 1)
    front = 0;
  }
  else
    front++;
  printf("Deleted element is: %d\n", item);
void display()
```

```
if (front == -1 && rear == -1)
    printf("Queue is empty\n");
    return;
  printf("Queue elements are:\n");
  if (rear >= front)
    for (int i = front; i <= rear; i++)
       printf("%d ", cq[i]);
    printf("\n");
  }
  else
  {
    for (int i = front; i < MAX_SIZE; i++)
       printf("%d ", cq[i]);
    for (int i = 0; i <= rear; i++)
       printf("%d ", cq[i]);
    printf("\n");
}
int main()
  int choice, item;
  do
    printf("Circular Queue Operations:\n");
    printf("1. Insert\n2. Delete\n3. Display\n4. Exit\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);
     switch (choice)
     {
         printf("Enter the element to insert: ");
         scanf("%d", &item);
         insert(item);
         break;
       case 2:
         delete();
         break;
       case 3:
         display();
         break;
       case 4:
         printf("Program Exited\n");
         break;
       default:
         printf("Invalid choice, Please try again\n");
```

} while (choice != 4);	
return 0;	
}	
Output:	
Insertion	Deletion
o	Deletion
Circular Queue Operations: 1. Insert	Circular Queue Operations:
1. Insert 2. Delete	
3. Display	1. Insert
4. Exit	2. Delete
Enter your choice: 1	3. Display
Enter the element to insert: 34	4. Exit
Circular Queue Operations:	
1. Insert	Enter your choice: 2
2. Delete	Deleted element is: 34
3. Display 4. Exit	Circular Queue Operations:
4. Exit Enter your choice: 1	
Enter the element to insert: 20	1. Insert
Circular Queue Operations:	2. Delete
1. Insert	3. Display
2. Delete	
3. Display	4. Exit
4. Exit	Enter your choice: 3
Enter your choice: 3	Queue elements are:
Queue elements are:	
34 20	20
Date :/	
	Teacher Sign