6. Develop a menu driven Program in C for the following operations on Circular QUEUE of characters(Array ImplementationofQueue with maximum size MAX)

f. Insert an Element on to CircularQUEUE

g. Delete an Elementf rom CircularQUEUE

h. Demonstrate Overflow and Underflow situations on CircularQUEUE

i. Display the status of CircularQUEUE

j. Exit

Support the program with appropriate functions for each of the above operations.

```
#include<stdio.h>
#include<stdlib.h>
#define MAX 4
Int ch, front = 0, rear = -1, count=0;
char q[MAX], item;
void insert()
      {
 if(count == MAX)
 printf("\nQueue is Full");
  else
  {
   rear = (rear + 1) % MAX;
   q[rear]=item;
   count++;
  }
}
void del()
{
 if(count == 0)
 printf("\nQueue is Empty");
 else
 {
```

```
   if(front > rear && rear==MAX-1)
   {
    front=0;
    rear=-1;
   count=0;
   }
  else
  {
   item=q[front];
   printf("\nDeleted item is: %c",item);
   front = (front + 1) % MAX;
   count--;
  }
 }
}

 void display()
{
  int i, f=front, r=rear;
  if(count == 0) printf("\nQueue is Empty");
 else
 {
   int j=f;
   printf("\nContents of Queue is:\n");
   for(int i=1; i<=count; i++)
  {
      printf("%c\t", q[j]);
       j = (j + 1) % MAX;
   }
  }
}
```

```
void main()
{
 do
 {
   printf("\n1. Insert\n2. Delete\n3. Display\n4. Exit");
   printf("\nEnter the choice: ");
   scanf("%d", &ch);
  getchar();
  switch(ch)
  {
   case 1: printf("\nEnter the character / item to be inserted: ");
            scanf("%c", &item);
            insert();
            break;
   case 2: del();
            break;
   case 3: display();
             break;
   case 4: exit(0);
         break;
  }
 }while(ch!=4);
}
```