LAB PROGRAM 3B

WAP to simulate the working of a circular queue of integers using an array. Provide the following operations: Insert, Delete & Display The program should print appropriate messages for queue empty and queue overflow conditions

```
#include<stdio.h>

#define size 5

int queue [size],front=-1,rear=-1;

void insert (int ele)

{

   if ((front== 0 && rear== size -1) || (rear == (front-1)%(size-1)))

   {

      printf("queue overflow\n");

      return;

   }

   else if (front ==-1)

   {

      front=rear=0;

      queue[rear]=ele;

   }

   else if ((rear == size-1 )&& (front=!0))

   {

      rear=0;

      queue [rear] =ele;

   }

   else

   {

      rear++;

      queue[rear]= ele;

   }

   printf(" %d has been inserted\n", ele);

}

void delete()
```

```
{
    if (front == -1)
    {
        printf("queue underflow\n");
        return;
    }
    int temp = queue[front];
    printf("%d has been deleted", temp);
    if (front== rear)
    {
        front= rear= -1;

    }
    else if (front== size-1)
    {
        front =0;
    }
    else
    {
        front ++;
    }
}
void display()
{
    if (front==-1)
    {
        printf("queue is empty");
        return;
    }
    printf("elements of the queue are:");
    if(rear>=front)
```

```
    {
        for(int i= front; i<= rear;i++)
        {
            printf("%d ", queue[i]);
        }
    }
    else
    {
        for(int i= front; i< size; i++)
        {
            printf("%d ", queue[i]);
        }
        for(int i=0; i<= rear;i++)
        {
            printf("%d ", queue[i]);
        }
    }
    printf("\n");
}
int main()
{
    int choice,ele;
    while (1)
    {
        printf("\n circular queue operators:\n");
        printf("1.insert\n 2.delete\n 3.display\n 4.exit\n");
        printf("enter your choice");
        scanf("%d", &choice);
        switch(choice)
        {
            case 1: printf("enter element to be inserted ");
```

```
            scanf("%d",&ele);
            insert (ele);
            break;
        case 2: delete ();
            break;
        case 3: display();
            break;
        case 4: return 0;
        default: printf("invalid choice");
      }
   }
}
```