DHRUVA M
1BM19CS049

```
void enque(int ele)
{
  if (((front == 0 && rear = MAX-1)) || (front = rear+1))
    {
      printf("Que is full\n");
      return;
    }
  else
  {
    rear = (rear+1) % MAX;
    queue[rear] = ele;
    if (front == -1)
      front = 0;
  }
}
int Deque()
{
  int item;
  if ((front == -1) && (rear == -1))
  {
    return(-999);
  }
```

DHRUVA M
1BM19CS049

```
else
{
    item = queue[front];
    if (front == rear)
    {
        front = -1;
        rear = -1;
    }
    else
    {
        front = (front+1)%MAX;
    }
    return item;
}
}
void display()
{
int i;
if ((front == -1 && rear == -1)) || (front = rear))
{
    printf("Queue is empty\n");
    return;
}
else
{
    printf("The queue contents are\n");
```

```
        for (i = front ; i <= rear ; i++)
            printf ("%d \t", queue[i]) ;
    } printf ("\n") ;
}
```