19-10-2020

VALLISHA. M
1BM19CS177

CIRCULAR QUEUE

```
size = 4
front = rear = -1
queue [size]

void enqueue (int input)
{
    if ((front = = 0 && rear = = size -1)
         || (rear = = front -1))
         printf " QUEUE IS FULL"
    else
    {
         if ( front = = -1 && rear = = -1)
              front = rear = 0
         else if ( rear = = size-1)
              rear = 6
         else rear ~~++~~= rear +1
         queue [rear] = input
    }
}

void dequeue ()
{
    if ( front = = -1 && rear = = -1)
    {
         print " QUEUE is empty"
         return
    }
```

```
   output= queue[front];
   if (front ==rear)
           front = rear =-1
   else
          front = (front+1) % size
   print ~~OUTPUT~~ output
}
void   display ()
{
     if(front ==-1 && rear==-1)
     {
          print "QUEUE is empty"
          return
     }
     i =0
     ~~print~~
     if (front <= rear)
     {
            for (i= front; i<=rear; i++)
                  print (queue[i])
     }
      else
      {        for(i = front; i<=rear; i=(i+1)%size)
                  print(queue[i])
      }
}
```

```
int main()
{
    print ("Enter 1 to enqueue")
    print ("Enter 2 to dequeue")
    print ("Enter 3 to display")
    Input choice; // scanf
    switch (choice)
    {
          case 1:
            print ("Enter numbers to enqueue:")
            Input (input)
            enqueue (input)
            break;
          case 2:
              dequeue ();break;
          case 3:
                display(); break;
          default:
                print "wrong input";
                break;
    }
    return 0;
}
```