Exercise on Queue

Suppose that there is an array-based queue, on array Q[x] of size N. Operations enqueue, dequene, and size are shown in Ch.5-2. Now add the following operations of stacks to the queue.

(1) push(o), adding object o at the front position of the queue.

(2) pop(), removing from the front position of the queue.

(3) Show your algorithms for (1) and (2), both running in O(1) time.

Describe your algorithm by pseudo code of Ch.4,p.6-7, and add in-line comments to important statement. Programming code is not necessary, and just showing programming code will have point reduction.

```
Algorithm isFull()
    return (size = N)                    // the queue is full when
size reaches capacity
```

```
Algorithm isEmpty()
    return (size = 0)                    // the queue is empty when
there are no elements
```

```
Algorithm push(o)
    if isFull()
        throw FullQueueException         // cannot insert into a
full queue

    front ← (front - 1 + N) mod N        // move front backward

    Q[front] ← o                         // insert input o at front

    size ← size + 1                      // increment size
```

```
Algorithm pop()
    if isEmpty()
        throw EmptyQueueException        // cannot remove from an
empty queue

    o ← Q[front]                         // retrieve front element

    front ← (front + 1) mod N            // move front forward

    size ← size - 1                      // decrement size

    return o
```