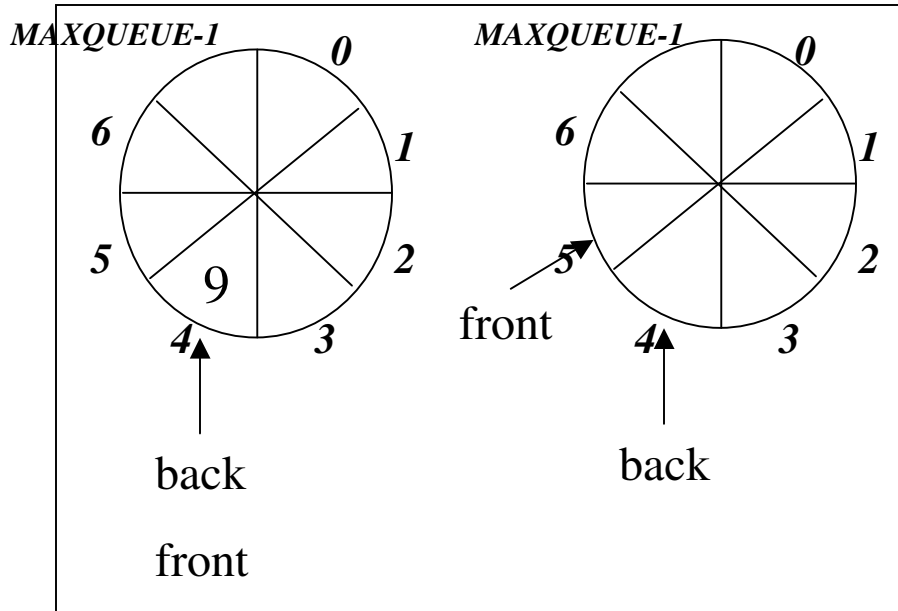
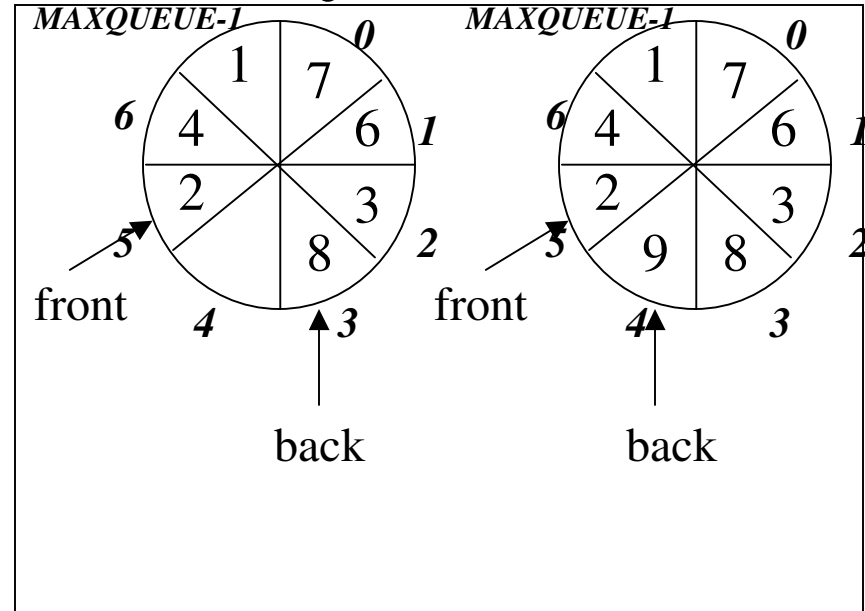


# Circular Array



Queue with single item  $\xrightarrow{\text{delete item}}$  Empty Queue

*front* passes *back* when the queue becomes empty

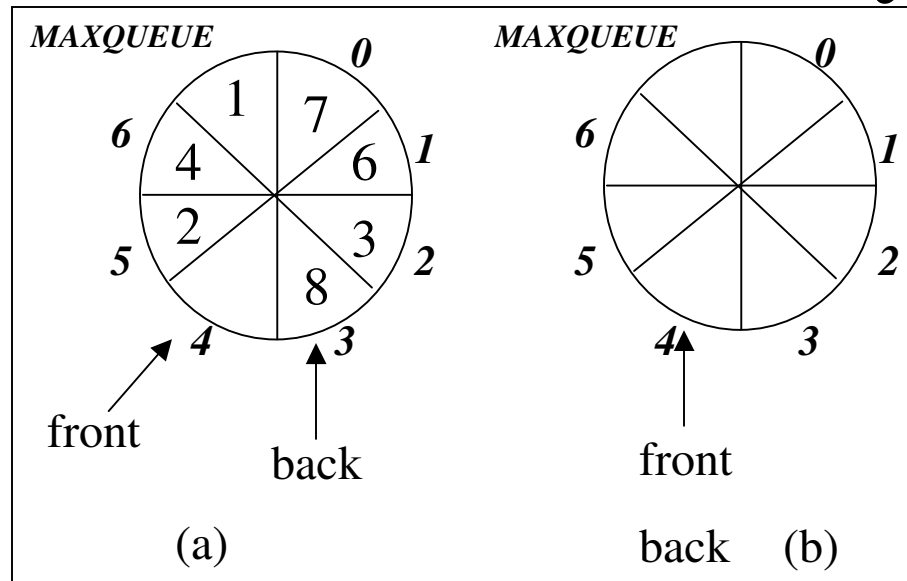


Queue with single empty slot  $\xrightarrow{\text{insert item}}$  Full Queue

*back* catches up with *front* when the queue becomes full

In both cases, in order to distinguish between an empty and a full queue it is necessary to keep a count of the number of items in the queue

# Circular Array



A variation on the circular queue implementation which does not require a count of the items in the queue is where  $MAXQUEUE+1$  locations are declared for the array, but the queue only uses  $MAXQUEUE$  locations, leaving a free space in the queue at all times. *front* is made the index of the location before the front of the queue. The queue is full ( (a) in diagram above) if:

$$front == (back+1) \% (MAXQUEUE-1)$$

and empty ( (b) in diagram above) if:

$$front == back$$