Name: Hasan Yahya, Class: BSE-6C, Roll No: 22L-7971

- **Schema:**

OrderedQueue

$$queue: seq\ \mathbb{Z}$$
$$max\_size: \mathbb{N}$$
$$size: \mathbb{N}$$

$$\forall\ i, j: \mathbb{N} \mid i < j \land j \leqslant \#queue \bullet queue(i) \leqslant queue(j)$$

There is a sequence of $\mathbb{Z}$ which can be an Element of numbers. It has a maximum and current size. Moreover, (∀ i, j: ℕ | i < j ∧ j ≤ #queue • queue(i) ≤ queue(j)) defines that the queue will be maintained in an ascending order.

- **Initialization:**

OrderedQueueInit

$$OrderedQueue'$$

$$queue' = \langle\ \rangle$$
$$max\_size' = 20$$
$$size = 0$$

Here at initialization the sequence of queue must be empty. I set the maximum size to 20 and the current size will always be zero at start.

- **Enqueue:**

Enqueue

$$\Delta OrderedQueue$$
$$x?: \mathbb{Z}$$

$$size < max\_size$$
$$queue' = insertAscending(queue, x?)$$
$$size' = size + 1$$
$$max\_size' = max\_size$$

The set of elements in queue' is equal to the set of elements in queue plus number? Input maxSize will remain same in Enqueue operation. The insertAscending will insert the new element in Ascending order. However, it's defination is below ie, it filters the queue into 2 parts (high and low) and then contatinates the new element in between the 2 filtered sub sequences. Moreover, queue cannot be full.

$insertAscending(s: seq \mathbb{Z}; x: \mathbb{Z}): seq \mathbb{Z}$

$= \langle y: \mathbb{Z} \mid y \in s \land y \leq x \rangle \frown \langle x \rangle \frown \langle z: \mathbb{Z} \mid z \in s \land z > x \rangle$

So, enqueue becomes,

Enqueue

$\Delta OrderedQueue$
$x?: \mathbb{Z}$

$size < max\_size$
$queue' = \langle y: \mathbb{Z} \mid y \in s \land y \leq x \rangle \frown \langle x \rangle \frown \langle z: \mathbb{Z} \mid z \in s \land z > x \rangle$
$size' = size + 1$
$max\_size' = max\_size$

- **Enqueue Error:**

If size is equal to maximum size, return a message without changes to OrderedQueue.

EnqueueError

$\Xi OrderedQueue$
$message!: Error$

$size = max\_size$
$message! =$ "*Buffer is Full*"

- **Dequeue:**

Dequeue

$\Delta OrderedQueue$
$x!: \mathbb{Z}$

$size > 0$
$queue \neq \langle \rangle$
$x! = head(OrderedQueue)$
$queue' = queue(2..size)$
$size' = size - 1$

Removes an element from the start of list. The it puts the new queue as the old queue but from index 2 to size. Ie, if we have <1, 2, 3, 4> then queue (2..size) makes it <2, 3, 4>. Moreover, the queue cannot be empty.

- **Dequeue Error:**

If the queue is already empty, give an error message.

DequeueError

$\Xi OrderedQueue$ $message!: Error$
$size = 0$ $message! = \text{“}Buffer\ is\ Empty\text{”}$

- **isEmpty:**

isEmpty

$\Xi OrderedQueue$ $empty!: Boolean$
$empty! \Leftrightarrow (size = 0)$ $max\_size' = max\_size$ $size' = size$ $queue' = queue$

Return a Boolean of true if and only If (⇔) the queue size is zero.

- **isFull:**

isFull

$\Xi OrderedQueue$ $full!: Boolean$
$full! \Leftrightarrow (size = max\_size)$ $max\_size' = max\_size$ $size' = size$ $queue' = queue$

Return a Boolean value of true if and only if (⇔) the queue size equals maximum size, which in this case is 20.