Implementation of Queue using VDM - SL

Queue
queue : Element*
Enqueue (Element)
Dequeue () : Element
GetFront () : Element
IsEmpty () : Boolean
IsFull () : Boolean

(1) types

Element : TOKEN

(2) values

LIMIT : N = 20

1, 2, 1, 3

(3) State Queue of

queue : Element*

inv mk-Queue(q) $\triangleq$ len q $\leq$ LIMIT

init mk-Queue(q) $\triangleq$ q = []

end

(4) Enqueue(itemIn : Element)

ext wr queue : Element*

pre len queue < LIMIT

post queue = $\overline{queue}$ ^ [itemIn]

(5) DeQueue() itemRemoved : Element

ext wr queue : Element*

pre ~~len~~ queue $\neq$ []

post queue = tl $\overline{queue}$ $\wedge$

itemRemoved = hd $\overline{queue}$

(6) getFront() frontElement : Element

ext rd queue : Element*

pre TRUE

post ~~itemRemoved~~ frontElement = hd queue

(7) IsEmpty() query : B

ext rd queue : Element*

pre TRUE

post queue $\Leftrightarrow$ queue = []

(8) IsFull() query : B

ext rd queue : Element*

pre TRUE

post query $\Leftrightarrow$ len query = LIMIT