Example — Queue ADT

This example is the first-in-first-out (FIFO) queue. For simplicity we formulate this ADT monomorphically with queue items as numbers. We take the Number and Boolean ADTs as pre-defined, with operations as usual.

SIGNATURE

NEW: → Queue
IsNEW: Queue → Boolean
FRONT: Queue → Number
ADD: Queue × Number → Queue
DEL: Queue → Queue

INFORMAL DESCRIPTION

Items are added to a Queue at the “back” and accessed/removed from the “front” — this is the FIFO behavior. NEW is a constant — the empty Queue. The ADD operation appends the second argument at the back of the first argument and returns the resulting Queue. The DEL operation removes the “front” item of the Queue, and FRONT returns the “front” item.

EQUATIONS (for all q   Queue and n   Number)

IsNEW(NEW) = True
IsNEW(ADD(q,n)) = False
FRONT(ADD(q,n)) = if IsNEW(q) then n else FRONT(q)
DEL(NEW) = NEW
DEL(ADD(q,n)) = **if** IsNEW(Q)
then NEW
else ADD(DEL(q),n)

INITIAL ALGEBRA (term equivalence classes)

[NEW] = {NEW, DEL(ADD(NEW,n)),
DEL(DEL(ADD(ADD(NEW,n1),n2))), … }
[ADD(NEW,n1)] = { ADD(NEW,n1),
DEL(ADD(ADD(NEW,n2),n1)), … }
etc.