```
------------------ MODULE NProcBoundedBuffer ------------------
EXTENDS Integers, Sequences, ISequences

CONSTANT N, Msg, Input
ASSUME /\ N \in Nat \ {0}
       /\ Input \in ISeq(Msg)

a (+) b == (a + b) % 2*N
a (-) b == (a - b) % 2*N

(***********************************************************
--algorithm NProcBBuf {
    variables in = Input, out = << >>,
              buf \in [0..(N-1) -> Msg], p = 0, c = 0;
    fair process (Buffer \in 0..(N-1))
    { b1:- while (TRUE)
          { await p % N = self ;
            buf[self % N] := IHead(in);
            in := ITail(in) ;
            p := p (+) 1 ;
      b2:     await c % N = self ;
            out := Append(out, buf[self % N]) ;
            c := c (+) 1
          }
    }
}
***********************************************************)
===============================================================
```