MODULE *GCD*

Specification for the algorithm to find the greatest common divisor between two integer numbers greater than zero, aka *Euclid* algorithm.

EXTENDS *Integers*

CONSTANTS M, N

VARIABLES x, y

$PositiveInteger(n) \triangleq n \in Nat \land n \neq 0$

$$TypeInvariant \triangleq \land PositiveInteger(x) \\ \land PositiveInteger(y)$$

Initial state

$$Init \triangleq (x = M) \land (y = N)$$

Next state of computation

$$Next \triangleq ((x < y) \land (x' = x) \land (y' = y - x)) \\ \lor ((y < x) \land (y' = y) \land (x' = x - y))$$

$$Spec \triangleq Init \land \Box[Next]_{\langle x, y \rangle}$$

THEOREM $Spec \Rightarrow TypeInvariant$