MODULE *Euclid*

EXTENDS *Integers*, *GCD*, *TLC*
CONSTANTS N
ASSUME $\land N \in Nat \setminus \{0\}$

```
**************************************************************************
--fair algorithm Euclid{
 variables x ∈ 1 .. N, y ∈ 1 .. N, x0 = x, y0 = y ;
 { abc: while ( x ≠ y ) { d: if ( x < y ) { y := y − x }
                                 else     { x   := x − y }
                          } ;
   assert (x = y) ∧ (x = GCD(x0, y0))
 }
}
**************************************************************************
```

BEGIN TRANSLATION
VARIABLES x, y, $x0$, $y0$, pc

$vars \triangleq \langle x, y, x0, y0, pc \rangle$

$Init \triangleq$ Global variables
$\land x \in 1 \,..\, N$
$\land y \in 1 \,..\, N$
$\land x0 = x$
$\land y0 = y$
$\land pc =$ "abc"

$abc \triangleq \land pc =$ "abc"
$\land$ IF $x \neq y$
THEN $\land pc' =$ "d"
ELSE $\land Assert((x = y) \land (x = GCD(x0, y0)),$
"Failure of assertion at line 12, column 4.")
$\land pc' =$ "Done"
$\land$ UNCHANGED $\langle x, y, x0, y0 \rangle$

$d \triangleq \land pc =$ "d"
$\land$ IF $x < y$
THEN $\land y' = y - x$
$\land x' = x$
ELSE $\land x' = x - y$
$\land y' = y$
$\land pc' =$ "abc"
$\land$ UNCHANGED $\langle x0, y0 \rangle$

$Next \triangleq abc \lor d$
$\lor$ Disjunct to prevent deadlock on termination

$$(pc = \text{“Done”} \land \text{UNCHANGED } vars)$$

$$Spec \triangleq \land Init \land \Box[Next]_{vars} \\ \land \mathrm{WF}_{vars}(Next)$$

$$Termination \triangleq \Diamond(pc = \text{“Done”})$$

END TRANSLATION

\ * Modification History
\ * Last modified *Tue Jun* 03 15:13:03 *CST* 2014 by *yaojingguo*
\ * Last modified *Tue Jun* 03 11:00:10 *CST* 2014 by *yaojingguo*
\ * Last modified Sat May 31 17:16:19 *CST* 2014 by *jing*
\ * Created Sat May 31 16:03:52 *CST* 2014 by *jing*