PARALLEL GCD (60 minutes)
========================

Algorithm description

Various parallel GCD algorithms exist. In this challenge, we consider a simple Euclid-like algorithm with two parallel threads. One thread subtracts in one direction, the other thread subtracts in the other direction, and eventually this procedure converges on GCD.

Implementation

In pseudocode, the algorithm is described as follows:

```
(
  WHILE a != b DO
      IF a>b THEN a:=a-b ELSE SKIP FI
  OD
||
  WHILE a != b DO
       IF b>a THEN b:=b-a ELSE SKIP FI
  OD
);
OUTPUT a
```

Verification tasks

Specify and verify the following behaviour of this parallel GCD algorithm:

Input: two positive integers a and b
Output: a positive number that is the greatest common divisor of a and b

Feel free to add synchronisation where appropriate, but try to avoid blocking of the parallel threads.

Sequentialization

If your tool does not support reasoning about parallel threads, you may verify the following pseudocode algorithm:

```
WHILE a != b DO
   CHOOSE(
       IF a > b THEN a := a - b ELSE SKIP FI,
       IF b > a THEN b := b - a ELSE SKIP FI
   )
OD;
OUTPUT a
```