```racket
#lang racket

(define (GCD x y)
  (if (= x 0)
      y
      (if (= y 0)
          x
          (GCD y (modulo x y)))))

(GCD 41 4)

(GCD 5740 70)

;; solution using cond

(define (GCD-2 x y)
  (cond ((= x 0) y)
        ((= y 0) x)
        (else (GCD-2 y (modulo x y)))))

(GCD-2 5740 70)


;; solution using let

(define (GCD-3 n1 n2)
  (cond ((= n1 0) n2)
        ((= n2 0) n1)
        (else (let ((remainder (modulo n1 n2)))
                (GCD-3 n2 remainder)))))

(GCD-3 48 6)
```