Euclidean Algorithm

--for GCD(a, b)
1. \( x := a, \ y := b. \)
2. If \( y = 0, \) return \( x \) & terminate;
3. else simultaneously:
   \( (x, y) := (y, \text{rem}(x, y)) \)
4. Go to step 2.

Euclid Algorithm State Machine

States ::= \( \mathbb{N} \times \mathbb{N} \)
start ::= \((a,b)\)
state transitions defined by

\((x, y) \rightarrow (y, \text{rem}(x, y)) \) for \( y \neq 0 \)

GCD correctness

Example: GCD(662,414)
\[= \text{GCD}(414, 248) \text{ since } \text{rem}(662,414) = 248\]
\[= \text{GCD}(248, 166) \text{ since } \text{rem}(414,248) = 166\]
\[= \text{GCD}(166, 82) \text{ since } \text{rem}(248,166) = 82\]
\[= \text{GCD}(82, 2) \text{ since } \text{rem}(166,82) = 2\]
\[= \text{GCD}(2, 0) \text{ since } \text{rem}(82,2) = 0\]
return value: 2

preserved invariant \( P(x,y) : \)
\[ [\text{gcd}(a,b) = \text{gcd}(x,y)] \]

GCD correctness

transitions: \((x, y) \rightarrow (y, \text{rem}(x, y)) \)
\( P \) is preserved because:
\[ \text{gcd}(x,y) = \text{gcd}(y, \text{rem}(x,y)) \]
for \( y \neq 0 \)
Proof: \( x = qy + \text{rem}. \)
any divisor of 2 of these 3 terms divides all 3.
GCD correctness

P is true at start: 
\[ x = a \quad y = b, \text{ so } P(\text{start}) \equiv [\gcd(a,b) = \gcd(a,b)] \]

Conclusion: on termination
\[ x = \gcd(a,b) \]
Proof: at termination, \( y = 0 \), so
\[ x = \gcd(x,0) = \gcd(x,y) = \gcd(a,b) \]

preserved invariant

GCD Termination

\( y \) decreases at each step 
\( y \in \mathbb{N} \) (another invariant) 
Well Ordering implies reaches minimum & stops

Derived Variables

A derived variable, \( v \), is a function assigning a “value” to each state:
\[ v: \text{States} \rightarrow \text{Values} \]
If \( \text{Vals} = \mathbb{N} \), say \( v \) is “\( \mathbb{N} \)-valued” or “nonnegative-integer-valued”

Robot on the grid example:
States = \( \mathbb{N}^2 \). Define the sum-value, \( \sigma \), of a state:
\[ \sigma(x,y) ::= x+y \]
an \( \mathbb{N} \)-valued derived variable

Called derived to distinguish from actual variables that appear in a program. 
For robot \( \text{Actual: } x, y \) 
\( \text{Derived: } \sigma \)
Another derived variable:
\[ \pi := \sigma \pmod{2} \]
\( \pi \) is \( \{0,1\} \)-valued

For GCD, have (actual) variables \( x, y \).
Proof of GCD termination: \( y \) is strictly decreasing & natural number-valued

Termination followed by Well Ordering Principle:
\( y \) must take a least value.
then the algorithm is stuck

\( \sigma \): up & down all over the place
neither increasing
nor decreasing
\( \pi \): is constant
both weakly increasing
& weakly decreasing
Partial-order valued variables

Defs of increasing/decreasing variables extend to variables with partially ordered values.

Team Problems

Problems

1–3