Proving algorithms correct

i.e. proving that a particular implementation matches a specification.

Often, 2 parts: (1) does it terminate for all inputs? (2) does it yield correct answer when it does terminate?

Theorem. GCDR correctly computes gcd. i.e. for all natural numbers a, b,

$$GCDR(a,b) = gcd(a,b).$$

algorithm   math specification.

First: show GCDR always terminates for any input.

Informally: 2nd argument b gets strictly smaller w/ every recursive call, and stops when $b = 0$.

Formally, proof by (strong) induction on b.

Base case: when $b = 0$, it terminates since it immediately returns a.

Induction step:

Suppose for some $b > 0$, $GCDR(a, b')$ terminates for any a and any $b' < b$. Then

$$GCDR(a,b) = GCDR(b, a \bmod b)$$

which terminates according to the IH since $a \bmod b < b$.

Now, show that $GCDR(a,b) = gcd(a,b)$. Need some math facts:

Lemma. If $b > 0$, (and $a > b$) then $gcd(a,b) = gcd(a-b, b) = gcd(a \bmod b, b)$.

Lemma. $gcd(a,b) = gcd(b,a)$.

Proof: by (Strong) induction on b

Base case: if $b = 0$, then $GCDR(a,0) = a = gcd(a,0)$. ✓

Induction step: Suppose that $GCDR(a, b') = gcd(a, b')$ for any a and any $b' < b$.

Then

$GCDR(a,b)$

$=$ $GCDR(b, a \bmod b)$ { defn of GCDR }

$=$ $gcd(b, a \bmod b)$ { IH, since $a \bmod b < b$ }

$=$ $gcd(a \bmod b, b)$ { Lemma – gcd is commutative }

$=$ $gcd(a, b)$ { Lemma }

How do we prove $GCDI(a,b) = gcd(a,b)$?

① Termination?

Informally – Stops when a or b are 0.

At each step, one stays same + one gets smaller –

if $a \le b$, then $b \leftarrow b \bmod a$ and $b \bmod a < a \le b$.

or if $b < a$, similarly a gets smaller.

(Can make it formal w/ generalized form of induction).

② Correctness?

Choose a loop invariant – something that remains true with every loop.

Suppose we call $GCDI(m, n)$. Then loop invariant is that

$gcd(a,b) = gcd(m,n)$.

– Is this true before loop starts? Yes, $a = m$ and $b = n$

– If it is true @ start of loop execution, is it still true @ end?

Yes, because of lemma.

Then loop invariant being true @ end implies we return the correct result.