CSE 311

Foundations of Computing I

Fall 2014

Useful GCD Fact

If *a* and *b* are positive integers, then

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

Proof:

By definition of mod, $a = qb + (a \bmod b)$ for some integer $q = a \text{ div } b$.

Let $d=\gcd(a,b)$. Then $d|a$ and $d|b$ so $a=kd$ and $b=jd$ for some integers k and j.
Therefore $(a \bmod b) = a - qb = kd - qjd = d(k - qj)$.
So, $d \mid (a \bmod b)$ and since $d \mid b$ we must have $d \leq \gcd(b, a \bmod b)$.

Now, let $e=\gcd(b, a \bmod b)$. Then $e \mid b$ and $e \mid (a \bmod b)$. It follows that $b=me$ and $(a \bmod b) = ne$ for some integers m and n. Therefore

$$a = qb + (a \bmod b) = qme + ne = e(qm+n)$$

So, $e \mid a$ and since $e \mid b$ we must have $e \leq \gcd(a, b)$.

Therefore $\gcd(a, b)=\gcd(b, a \bmod b)$.

Euclid's Algorithm

Repeatedly use the fact to reduce numbers until you get

$$\begin{aligned}\gcd(660,126) &= \gcd(126, 660 \bmod 126) &&= \gcd(126, 30)\\ &= \gcd(30, 126 \bmod 30) &&= \gcd(30, 6)\\ &= \gcd(6, 30 \bmod 6) &&= \gcd(6, 0)\\ &= 6\end{aligned}$$

$$660 = 5 \bullet 126 + 30$$

$$126 = 4 \bullet 30 + 6$$

$$30 = 5 \bullet 6$$

Euclid’s Algorithm

GCD(x, y) = GCD(y, x mod y)

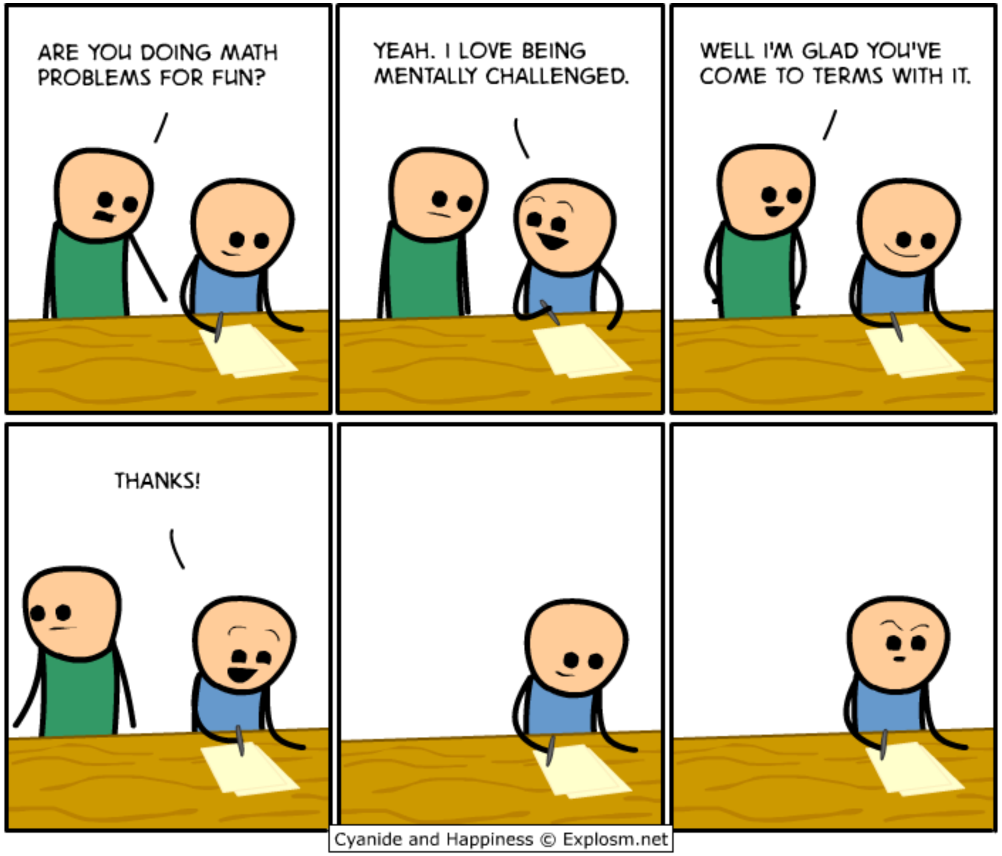
```
int GCD(int a, int b){ /* a >= b, b > 0 */
    int tmp;
    while (y > 0) {
        tmp = a % b;
        a = b;
        b = tmp;
    }
    return a;
}
```

Example: GCD(660, 126)

CSE 311: Foundations of Computing

Fall 2014

Lecture 13: Modular Inverses, Induction



Bézout's theorem

If a and b are positive integers, then there exist integers $\boldsymbol{s}$ and $\boldsymbol{t}$ such that

$$\gcd(a,b) = \boldsymbol{s}a + \boldsymbol{t}b.$$

Extended Euclidean algorithm

- **Can use Euclid's Algorithm to find s, t such that**

$$\gcd(a,b) = sa + tb$$

- e.g. gcd(35,27):

$35 = \mathit{1} \bullet 27 + 8$	$35 - \mathit{1} \bullet 27 = 8$
$27 = \mathit{3} \bullet 8 + 3$	$27 - \mathit{3} \bullet 8 = 3$
$8 = \mathit{2} \bullet 3 + 2$	$8 - \mathit{2} \bullet 3 = 2$
$3 = \mathit{1} \bullet 2 + 1$	$3 - \mathit{1} \bullet 2 = 1$
$2 = \mathit{2} \bullet 1 + 0$	

- **Substitute back from the bottom**

$1 = 3 - \mathit{1} \bullet 2 \quad = 3 - \mathit{1}\,(8 - \mathit{2} \bullet 3) \quad = (-\mathit{1}) \bullet 8 + \mathit{3} \bullet 3$

$\qquad = (-\mathit{1}) \bullet 8 + \mathit{3}\,(27 - \mathit{3} \bullet 8) \quad = \mathit{3} \bullet 27 + (-\mathit{10}) \bullet 8$

$\qquad =$

multiplicative inverse $\bmod m$

Suppose $\mathrm{GCD}(a,m)=1$

By Bézout's Theorem, there exist integers s and t such that $sa+tm=1$.

$s \bmod m$ is the multiplicative inverse of a:

$$1=(sa+tm) \bmod m=sa \bmod m$$

Solving Modular Equations

Solving $ax \equiv b \pmod{m}$ **for unknown** x **when** $\gcd(a, m) = 1$**.**

1. **Find** s **such that** $sa + tm = 1$
2. **Compute** $a^{-1} = s \bmod m$**, the multiplicative inverse of** a **modulo** m
3. **Set** $x = (a^{-1} \cdot b) \bmod m$

Example

Solve: $7x \equiv 1 \pmod{26}$

gcd(26, 7) = gcd(7, 5) = gcd(5, 2) = gcd(2, 1) = 1

26 = 7*3 + 5	5 = 26 – 7*3
7 = 5*1 + 2	2 = 7 – 5*1
5 = 2*2 + 1	1 = 5 – 2*2

1 = 5 – (7 – 5*1)*2
= (–7)*2 + 5*3
= (–7)*2 + (26 – 7*3)*3
= 7*(-11) + 26*3

So, $x = 15 + 26k$ for $k \in \mathrm{N}$.

Mathematical Induction

Method for proving statements about all natural numbers

- **A new logical inference rule!**
  - It only applies over the natural numbers
  - The idea is to **use** the special structure of the naturals to prove things more easily
- **Particularly useful for reasoning about programs!**

```
for(int i=0; i < n; n++) { … }
```

  - Show P(i) holds after i times through the loop

```
public int f(int x) {
    if (x == 0) { return 0; }
    else { return f(x – 1); }
}
```

  - f(x) = x for all values of x ≥ 0 naturally shown by induction.

Prove for all $k > 0$, n^k even $\rightarrow$ n even

Let $k > 0$ be arbitrary. We go by contrapositive. Suppose that n is odd. We know that if a, b are odd, then ab is also odd.

So,

$$(\ldots\bullet\ ((n\bullet n)\ \bullet n)\ \bullet\ldots\bullet n) = n^k$$

(k times)

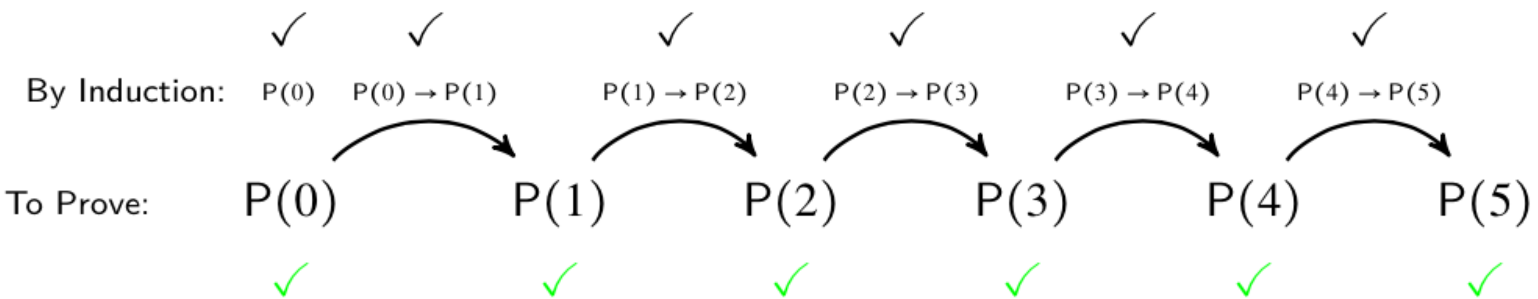
Those "…"s are a problem! We're trying to say "we can use the same argument over and over"… We should use induction instead.

Induction Is A Rule of Inference

Domain: Natural Numbers

$$
\begin{array}{l}
P(0) \\
\forall\, k\, (P(k) \;\rightarrow\; P(k+1)) \\
\hline
\therefore\; \forall\, n\, P(n)
\end{array}
$$

How does this technique prove P(5)?



First, we prove P(0).
Since P(n) → P(n+1) for all n, we have P(0) → P(1).
Since P(0) is true and P(0) → P(1), by Modus Ponens, P(1) is true.
Since P(n) → P(n+1) for all n, we have P(1) → P(2).
Since P(1) is true and P(1) → P(2), by Modus Ponens, P(2) is true.

Using The Induction Rule In A Formal Proof

$$\frac{\begin{array}{l} P(0) \\ \forall\, k\, (P(k) \;\rightarrow\; P(k+1)) \end{array}}{\therefore\; \forall\, n\, P(n)}$$

1. Prove P(0)
2. Let k be an arbitrary integer ≥ 0
    3. Assume that P(k) is true
    4. ...
    5. Prove P(k+1) is true
6. P(k) → P(k+1) — Direct Proof Rule
7. ∀ k (P(k) → P(k+1)) — Intro ∀ from 2-6
8. ∀ n P(n) — Induction Rule 1&7

What can we say about $1 + 2 + 4 + 8 + \ldots + 2^n$

- $1 = 1$
- $1 + 2 = 3$
- $1 + 2 + 4 = 7$
- $1 + 2 + 4 + 8 = 15$
- $1 + 2 + 4 + 8 + 16 = 31$

- Can we describe the pattern?
  - $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

Proving $1 + 2 + 4 + \ldots + 2^n = 2^{n+1} - 1$

- We could try proving it normally…
  - We want to show that $1 + 2 + 4 + \ldots + 2^n = 2^{n+1}$.
  - So, what do we do now? We can sort of explain the pattern, but that's not a proof…
- We could prove it for n=1, n=2, n=3, … (individually), but that would literally take forever…

Instead, Let's Use Induction

$$\begin{array}{l} P(0) \\ \forall\, k\, (P(k) \rightarrow P(k+1)) \\ \hline \therefore\ \forall\, n\, P(n) \end{array}$$

1. Prove P(0) — **Base Case**
2. Let k be an arbitrary integer ≥ 0
   3. Assume that P(k) is true — **Inductive Hypothesis**
   4. ...
   5. Prove P(k+1) is true — **Inductive Step**

6.	P(k) → P(k+1)	Direct Proof Rule
7.	∀ k (P(k) → P(k+1))	Intro ∀ from 2-6
8.	∀ n P(n)	Induction Rule 1&7

Conclusion