Elementary Number Theory

- **Modulo Arithmetic (definition, properties, & different notation)**
- **Computing $a^m \pmod{n}$**
- **Euler's Phi Function (definition, formula)**
- **Euler's Theorem, Fermat's Little Theorem, and the Existence of Inverses**
- **Computing Inverses (later)**
- **Chinese Remainder Theorem**
- **Generators exist modulo 2, 4, p^e, and $2p^e$**

Algorithm from Knuth, Vol. 2, p. 320

Algorithm A ***(Modern Euclidean algorithm).*** **Given nonnegative integers u and v, this algorithm finds their greatest common divisor.**

A1. [Check v = 0] If $v = 0$, the algorithm terminates with u as the answer.

A2. [Take u mod v] Set $r \leftarrow u \bmod v$, $u \leftarrow v$, $v \leftarrow r$, and return to A1. (The operations of this step decrease the value of v, but they leave $\gcd(u, v)$ unchanged.)

Theorem (Lamé). ***Let $\phi = (1 + \sqrt{5})/2$. Let $0 \leq u, v < N$ in Algorithm A. Then the number of times step A2 is repeated is $\leq \lfloor \log_\phi(\sqrt{5}N) \rfloor - 2$.***

Algorithm from Knuth, Vol. 2, p. 320

Algorithm A *(Modern Euclidean algorithm)*. Given nonnegative integers u and v, this algorithm finds their greatest common divisor.

A1. [Check v = 0] If $v = 0$, the algorithm terminates with u as the answer.

A2. [Take u mod v] Set $r \leftarrow u \bmod v$, $u \leftarrow v$, $v \leftarrow r$, and return to A1. (The operations of this step decrease the value of v, but they leave $\gcd(u, v)$ unchanged.)

Theorem. *The running time for computing the greatest common divisor of two positive integers $\leq N$ is*

$$\ll \log N(\log\log N)^2 \log\log\log N.$$

Theorem. ***Given integers a and b, not both 0, there exist integers u and v such that*** $au + bv = \gcd(a, b)$.

Example. $u = 567$ **and** $v = 245$

Comment: The average value of $\gcd(u, v)$ **is** $\asymp \log N$ **but "usually" it's much smaller.**

Probable Primes and the Like

- **The use of Fermat's Little Theorem**
- **The example $341 = 11 \times 31$**
- **The example $561 = 3 \times 11 \times 17$**