Chapter 5: RSA and Factorization

Math 495, Fall 2008

Hope College

October 22, 2008

Primality Testing

- In practice, the algorithms used for testing primality are fast, but they do not always produce the correct answer. (That is, the penalty for using a fast algorithm is that it doesn't always work.)
- However, the error probability is a known constant. By running the algorithm many times on the same input, the probability of error can be reduced below any pre-set threshold.
- The prime number theorem implies that, for large N, a randomly chosen integer between 1 and N will be prime with probability approximately $1/\ln N$. Thus, a randomly chosen 512-bit integer will be prime with probability about $1/\ln 2^{512} \approx 1/355$.

Monte Carlo Algorithms

- A **yes-biased Monte Carlo algorithm** is a randomized algorithm for a decision problem in which a "yes" answer is always correct, but a "no" answer may be incorrect.
- The **error probability** is a number ϵ such that the probability of getting an incorrect "no" answer for any given input (for which the answer should be "yes") is at most ϵ.
- A **no-biased Monte Carlo algorithm** is defined similarly. A "no" answer is always correct, but a "yes" answer may be incorrect.
- **Problem 5.1: Composites.** Instance: an integer $n \geq 2$. Question: Is n composite?
- We will cover two Monte Carlo algorithms for **Composites**.

Quadratic Residues

- Let p be an odd prime. An integer a is defined to be a **quadratic residue** mod p if $a \not\equiv 0 \pmod{p}$ and the congruence $y^2 \equiv a \pmod{p}$ has a solution $y \in \mathbb{Z}_p$. If $a \not\equiv 0 \pmod{p}$ and the congruence $y^2 \equiv a \pmod{p}$ has no solution, then A is called a **quadratic non-residue** mod p.
- Exercise: Find the quadratic residues and non-residues mod 13.
- Theorem: Let p be an odd prime, and let a be a quadratic residue mod p. Then the congruence $y^2 \equiv a \pmod{p}$ has precisely two solutions in $\mathbb{Z}_p$, and they are additive inverses of each other.

Quadratic Residues

- **Problem 5.2: Quadratic Residues.** Instance: An odd prime p and an integer a. Question: Is a a quadratic residue mod p?
- Theorem 5.9 (Euler's Criterion): Let p be an odd prime. An integer a is a quadratic residue modulo p if and only if $a^{(p-1)/2} \equiv 1 \pmod{p}$.
- Using Euler's Criterion with the Square and Multiply Algorithm gives an algorithm for answering **Quadratic Residues** with complexity $O((\log p)^3)$, which is a polynomial function in the number of bits needed to represent p.

Legendre and Jacobi Symbols

- If p is an odd prime and a is an integer, define the Legendre symbol by

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } a \equiv 0 \pmod{p} \\ 1 & \text{if } a \text{ is a quadratic residue mod } p \\ -1 & \text{if } a \text{ is a quadratic non-residue mod } p \end{cases}$$

- Theorem 5.10: $\left(\dfrac{a}{p}\right) \equiv a^{(p-1)/2} \pmod{p}$.
- Let n be an odd integer, and $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ its prime factorization. For any integer a define the Jacobi symbol to be

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{e_1} \left(\frac{a}{p_2}\right)^{e_2} \cdots \left(\frac{a}{p_k}\right)^{e_k}$$

Legendre and Jacobi Symbols

- Caution: If p is prime, then
$$\left(\frac{a}{p}\right) \equiv a^{(p-1)/2} \pmod{p}.$$
However, if n is odd but not prime, it may or may not be the case that
$$\left(\frac{a}{n}\right) \equiv a^{(n-1)/2} \pmod{n}.$$
If this congruence holds, then n is called an **Euler pseudo-prime** to the base a.
- It can be shown that, for any odd composite number n, n is an Euler pseudo-prime to the base a for at most half of the integers $a \in \mathbb{Z}_n^*$.
- If $1 \leq a \leq n-1$ and $\left(\frac{a}{n}\right) = 0$, then n is composite.

Solovay-Strassen Algorithm

- SOLOVAY-STRASSEN ALGORITHM
  - Input n. Is n composite?
  - Choose a random integer a with $1 \leq a \leq n-1$.
  - Compute $\left(\frac{a}{n}\right)$.
  - If $\left(\frac{a}{n}\right) = 0$, then return "composite."
  - Compute $a^{(n-1)/2} \mod n$. If $\left(\frac{a}{n}\right) \equiv a^{(n-1)/2} \pmod{n}$ then return "prime."
  - Otherwise, return "composite."
- An answer of "composite" is always correct, so this is a yes-biased Monte Carlo algorithm for **Composites**. We have $\epsilon \leq 1/2$.

Solovay-Strassen Algorithm

- If n is prime, the answer produced by the algorithm will be "prime."
- If n is composite, then the algorithm will answer "prime" at most half of the time.
- For a given integer n, we can run the algorithm m times in succession (choosing a different a each time). If an answer of "composite" ever returns, we can stop, because n is composite. If an answer of "prime" is returned every time, we still aren't certain n is prime, but if m is large, we can conclude that n is almost certain to be prime.

Solovay-Strassen Algorithm

- In particular, if

  $A =$ "a random odd integer n of a specified size is composite"

  and

  $B =$ "the algorithm answers 'n is prime' m times in succession"

  then $\Pr(B|A) \leq 2^{-m}$.
- We are really interested in $\Pr(A|B)$. It can be shown by Bayes' Theorem that (approximately)

$$\Pr(A|B) \leq \frac{\ln n - 2}{\ln n - 2 + 2^{m+1}}.$$

- If $n \approx 2^{512}$, then $m = 100$ makes $\Pr(A|B) < 10^{-27}$.

Implementing the Solovay-Strassen Algorithm

- $a^{(n-1)/2} \mod n$ can be computed in time $O((\log n)^3)$.
- How can we compute $\left(\frac{a}{n}\right)$ without first factoring n?
- If n is a positive odd integer,
  - If $m_1 \equiv m_2 \pmod{n}$ then $\left(\frac{m_1}{n}\right) = \left(\frac{m_2}{n}\right)$.
  - $\left(\frac{2}{n}\right) = \begin{cases} 1 & \text{if } n \equiv \pm 1 \pmod{8} \\ -1 & \text{if } n \equiv \pm 3 \pmod{8} \end{cases}$
  - $\left(\frac{m_1 m_2}{n}\right) = \left(\frac{m_1}{n}\right)\left(\frac{m_2}{n}\right)$.
  - If m is a positive odd integer,

$$\left(\frac{m}{n}\right) = \begin{cases} -\left(\frac{n}{m}\right) & \text{if } n \equiv m \equiv 3 \pmod{4} \\ \left(\frac{n}{m}\right) & \text{otherwise} \end{cases}$$

- This can be used to compute $\left(\frac{a}{n}\right)$ in time $O((\log n)^3)$.

Miller-Rabin Algorithm

- MILLER-RABIN ALGORITHM
  - Input n. Is n composite?
  - Write $n-1 = 2^k m$ where m is odd.
  - Choose a random integer a with $1 \leq a \leq n-1$.
  - If $a^m \equiv 1 \pmod{n}$, return "prime."
  - For i from 0 to $k-1$, if $a^{2^i m} \equiv -1 \pmod{n}$, return "prime."
  - Otherwise, return "composite."
- Theorem: If n is prime, the MILLER-RABIN ALGORITHM returns "prime". Thus, this is a yes-biased Monte Carlo algorithm. The error probability can be shown to be at most 1/4.
- This algorithm runs in time $O((\log n)^3)$.