A Few Primality Testing Algorithms

Donald Brower

April 2, 2006

0.1 Introduction

These notes will cover a few primality testing algorithms. There are many such, some prove that a number is prime, others prove that a number is composite. To prove a number composite is easy—just provide a nontrivial factor of it. Since division is a polynomial time operation, this shows that COMPOSITE, the decision problem of recognizing composite numbers, is in NP, so PRIMES is in co-NP. To prove a number prime is a little more subtle. Suffice to say, there is a primeness certificate which can be verified in polynomial time, so PRIMES is in NP. Thus PRIMES is in NP $\cap$ co-NP and this was the state of affairs until 2003, when a polynomial time algorithm was discovered to decide PRIMES.

0.2 Definitions

By integer we mean the set of positive and negative whole numbers, including zero. We symbolize them by $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$. Let $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$ be the non-negative integers (i.e. natural numbers).

Given two integers a and b we say that a **divides** b (in symbols $a|b$) iff there is an integer k such that $ak = b$.

An integer n is **prime** iff the only divisors of n are 1 and n. If an integer is not prime we say that it is **composite**.

The fundamental theorem of arithmetic says that every integer can be decomposed into a product of a unique collection of prime numbers, that is, the primes are the building blocks of multiplication. Formally we say that every $n \in \mathbb{Z}$ can be written uniquely as a product $n = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$

for some $k > 0$, powers $\alpha_1, \ldots, \alpha_k > 0$ and distinct primes $p_1, \ldots, p_k$. Of course, *calculating* the factorization may not be easy or quick.

0.3 Trial Division

The simplest way to test for the integer n for primeness is to see if any numbers less than n divide it. After some thought, one sees that only primes up to $\sqrt{n}$ need to be tested, since if $a, b > \sqrt{n}$ then $ab > \sqrt{n}\sqrt{n} = n$. When n is "small", this algorithm is not too slow, and most of the following algorithms use trial division for such cases. The problem is that when n in "large" the search space is also "large".

0.4 Euclid's algorithm

Given two integers a and b we know that 1 divides both. The question is whether any other numbers also divide both. Since there may be more than one, we ask for the largest. We say an integer d is the **greatest common divisor of** a **and** b iff d divides both a and b and for every $e \in \mathbb{Z}$, if $e|a$ and $e|b$ then $e|d$. Abbreviate greatest common divisor as "gcd" and notate it as $d = \gcd(a, b) = (a, b)$. Both forms may be used, the first being more computer science-ish, the latter is more classical.

If $\gcd(a, b) = 1$ then we say that a and b are **coprime**.

In light of the fundamental theorem of arithmetic, by finding the gcd of two integers we are really constructing the largest common substring between the two prime factorizations. Since factoring can be difficult it is surprising that the gcd can be calculated rather easily.

Algorithm 1 $\gcd(x, y)$

Require: $x, y \in \mathbb{Z}$ with $x \geq y \geq 0$ and $x > 0$

1: **while** $y > 0$ **do**
2: $\quad x, y \leftarrow y, x \bmod y$
3: **return** x

The proof that algorithm **gcd** returns the gcd of x and y is omitted.

0.5 Faster GCD

Euclid does not have the last word on gcd algorithms; there are faster methods for computing the gcd of two numbers. Here is one, not the fastest, but still clever (see Crandall, Pomerence ch.9). It is called the Binary GCD Algorithm since it is especially suitable for numbers in binary representation. We start with some simple identities involving the gcd.

Theorem. *(Silver, Terzian, Stein)*

1. *If* x, y *are both even then* $\gcd(x, y) = 2 * \gcd(x/2, y/2)$
2. *If* x *is even and* y *is not then* $\gcd(x, y) = \gcd(x/2, y)$
3. *(Euclid)* $\gcd(x, y) = gcd(x - y, y)$
4. *If* x, y *are both odd then* $|x - y|$ *is even and less than* $\max\{x, y\}$

In the given implementation, let $v(m) = \max\{k : 2^k | m\}$ to be the logarithm of the maximum power of 2 which divides m. The algorithm is listed in the figure.

Algorithm 2 Binary $\gcd(x, y)$

```
[Initialization]
b = min{v(x), v(y)}
x = x/(2^v(x))
y = y/(2^v(y))
[Loop]
while x ≠ y do
        x, y = min{x, y}, |x − y| /(2^v(|x−y|))
return 2^b * x
```

0.6 Euler Phi

The Euler phi function $\phi(n)$ gives the number of positive integers less than and coprime to n. For example, $\phi(5) = 4$ since $(5, 1) = (5, 2) = (5, 3) = (5, 4) = 1$, and $\phi(12) = 4$ since only 1, 5, 7, and 11 are less than and coprime to 12. It satisfies the following properties:

1. $\phi(1) = 1$

2. If p is prime and $k \geq 1$ then $\phi(p^k) = p^{k-1}(p-1)$

3. If m and n are coprime, $\phi(mn) = \phi(m)\phi(n)$.

Using these relations we can calculate $\phi(12)$ without counting: $\phi(12) = \phi(4)\phi(3) = \phi(2^2) \cdot 2 = 2 \cdot 2 = 4$. The relations also give an easy way to calculate $\phi(n)$ given a prime factorization of n.

0.7 Group Theory and Modular Arithmetic

A **group** is given by a tuple $(G, *)$ where G is a set and $* : G \times G \to G$ is a binary operation with the following properties for a fixed constant $e \in G$:

1. (identity) $e * x = x * e$ for all $x \in G$

2. (inverse) For every $x \in G$ there is an element $x' \in G$ such that $x * x' = x' * x = e$

3. (associativity) $x * (y * z) = (x * y) * z$ for every $x, y, z \in G$

We will denote the group $(G, *)$ by G if there is no confusion over what the operation is.

A few examples of groups:

- $(\mathbb{Z}, +)$ the group of integers under addition
- $(\mathbb{Z}_n, +)$ The group of integers under addition modulo n for any $n \in \mathbb{N}$
- $(S_n, \circ)$ The group of permutations of n elements were $\circ$ is function composition
- $(\mathbb{Q}, +)$ The group of rational numbers under addition
- $(\mathbb{R}, +)$ The group of real numbers under addition
- $(\mathbb{R} \setminus \{0\}, \cdot)$ The group of nonzero real numbers under multiplication

Given a group G, a **subgroup** H of G, denoted $H \leq G$, is a group whose set of elements is a subset of G and whose operation is a restriction of G's to that subset. Given an element $a \in G$ we can form the set $\langle a \rangle = \{a^n : n \in \mathbb{Z}\}$ where a^n is shorthand for $a * a * \cdots * a$(n times). This set forms a

subgroup of G and is called the **subgroup generated by** a. If there is an element $g \in G$ such that $\langle g \rangle = G$ then we say G is **cyclic**.

If G is a group, let the order of G be the cardinality of the underlying set of G, and write $|G|$ for this number. (N.B. some authors write $\#G$ for this). If $|G| < \infty$ then we say G is a finite group. It is a theorem of Lagrange that if G is a finite group and H is a subgroup of G then the order of H divides the order of G. For what follows, the most important consequence of Lagrange's theorem is that if H is a proper subgroup of a finite group G then $|H| \leq \frac{1}{2}|G|$.

We will mainly be concerned with two families of groups. One is $(\mathbb{Z}_n, +)$ with $n \in \mathbb{N}$, as seen above. These groups are cyclic with generator 1, finite and have order $|\mathbb{Z}_n| = n$.

The other family of groups is $(\mathbb{Z}_n^*, \cdot)$ with $n \in \mathbb{N}$ and

$$\mathbb{Z}_n^* = \{x : 1 \leq x < n \text{ and } \gcd(x, n) = 1\}$$

under multiplication modulo n. The coprime requirement ensures the existence of inverses. These groups are finite with order $|\mathbb{Z}_n^*| = \phi(n)$, where $\phi(n)$ is the Euler phi function evaluated at n.

Observe that $|\mathbb{Z}_n^*| = n - 1$ if and only if n is prime. In this case, $\mathbb{Z}_n^*$ is also cyclic. Thus, giving a generator for $\mathbb{Z}_n^*$ provides a certificate that n is prime. This is the idea behind showing $PRIMES \in NP$.

The group $\mathbb{Z}_n^*$ is not necessarily cyclic for arbitrary n. For example, the group $\mathbb{Z}_8^*$ is not cyclic. In fact, the groups $\mathbb{Z}_n^*$ which are cyclic can be characterized precisely.

Theorem. *$\mathbb{Z}_n^*$ is cyclic if and only if n is either 2, 4, p^k, or $2p^k$ where k is any positive integer and p is any odd prime.*

I was going to add a proof of this theorem, but it would take us on a long digression. If you want to find it look for "primitive roots" in any number theory textbook.

0.8 Fermat's Theorem

Theorem. *Given a prime p and element $x \in \mathbb{Z}_p^*$*

$$x^{p-1} \equiv 1 \pmod{p}$$

This is because the number of elements in the subgroup generated by x must divide the number of elements in $\mathbb{Z}_p^*$, which is $p-1$ when p is prime. As you might expect, the argument can be extended to $\mathbb{Z}_n^*$ for arbitrary n.

Theorem. *(Euler's Theorem) Given positive integer n and element $x \in \mathbb{Z}_n^*$*

$$x^{\phi(n)} \equiv 1 \pmod{n}$$

0.9 Chinese Remainder Theorem

Theorem. *Let $n_1, \ldots, n_k$ be a sequence of pairwise coprime numbers with product $n = \prod_{i=1}^{k} n_i$. For any sequence $r_1 \in \mathbb{Z}_{n_1}, \ldots, r_k \in \mathbb{Z}_{n_k}$ there is a unique $r \in \mathbb{Z}_n$ with $r = r_i \pmod{n_i}$ for $1 \leq i \leq k$.*

0.10 Legendre and Jacobi Symbols

An element $a \in \mathbb{Z}_n^*$ is said to be a **quadratic residue** is there exists an $x \in \mathbb{Z}_n^*$ such that $a = x^2 \pmod{n}$. In other words, a quadratic residue is an element with a square root.

Theorem. *(Euler's Criterion) For a prime p an element $a \in \mathbb{Z}_p^*$ is a quadratic residue if and only if $a^{\frac{p-1}{2}} = 1 \pmod{p}$*

Given a prime p and element $a \in \mathbb{Z}_p^*$ we define the **Legendre symbol** to be $\left[\frac{a}{p}\right] = \begin{cases} 1 & \text{if } a \text{ is a quadratic residue } \pmod{p}, \\ -1 & \text{otherwise.} \end{cases}$

It turns out that $\left[\frac{a}{p}\right] = a^{\frac{p-1}{2}}$ since $a^{\frac{p-1}{2}}$ is either 1 or -1.

We can extend the Legendre symbol to any odd number, the extension is called the Jacobi symbol. For any odd integer n with prime factorization $p_1^{\alpha_1} \cdots p_k^{\alpha_k}$, and integer a coprime to n define the **Jacobi symbol** to be $\left[\frac{a}{n}\right] = \prod_{i=1}^{k} \left[\frac{a}{p_i}\right]^{\alpha_i}$. As with the gcd, we do not need to know a complete factorization to compute the Jacobi symbol for two integers. We use the following relations to derive an algorithm

Theorem. *Given integers a,b, and n, the Jacobi symbol satisfies the following properties, when it is defined:*

1. $\left[\frac{ab}{n}\right] = \left[\frac{a}{n}\right]\left[\frac{b}{n}\right]$.

2. *If* $a \equiv b \pmod{n}$ *then* $\left[\frac{a}{n}\right] = \left[\frac{b}{n}\right]$.

3. *If* a *and* n *are both odd and coprime then* $\left[\frac{a}{n}\right] = (-1)^{\frac{a-1}{2}\frac{n-1}{2}} \left[\frac{n}{a}\right]$.

4. $\left[\frac{1}{n}\right] = 1$.

5. $\left[\frac{2}{n}\right] = \begin{cases} -1 & n \equiv 3, 5 \pmod{8} \\ 1 & n \equiv 1, 7 \pmod{8} \end{cases}$

Algorithm 3 Jacobi evaluation(a, n)

Require: $a > 0$, n odd, and $\gcd(a, n) = 1$

```
r ← 1
while a ≠ 1 do
    if a is even then
        Let a = 2^k d where d is odd
        if k is odd and (n ≡ 3 or 5 (mod 8)) then
            r ← −r
        a ← d
    else if a is odd and a > n then
        a ← a mod n
    else {a is odd and a < n}
        r ← (−1)^((a−1)/2 (n−1)/2) r
        a, n ← n, a
return r
```

The Jacobi symbol is nice, since it is defined over all odd denominators, but reduces to the Legendre symbol for prime denominators. This makes it easy to calculate.

0.11 Prime Testing

The general idea in testing for primeness is to take a statement of the form "If p is prime then $S(p)$" and look at its, possibly erroneous, converse "If $S(p)$ then p is prime". If S holds for some non-prime value n then say n is an S-pseudoprime. Some properties S are better than others, in the sense of not having very many pseudoprimes.

Fermat's test, derived from Fermat's theorem above, is one possible prime test. In fact, it is a very famous test, so famous that Fermat-pseudoprimes have their own name: **Carmichael numbers**. The formal definition states that n is a Carmichael number if for all $a \in \mathbb{Z}_n^*$

$$a^{n-1} \equiv 1 \pmod{n}.$$

0.12 Solovay-Strassen Composite Proving

Finally, a randomized algorithm.

Algorithm 4 Solovay-Strassen Composite Proving(n)

Require: Odd number n.

1: Choose $a \in \mathbb{Z}_n \setminus \{0\}$ uniformly at random.
2: **if** $\gcd(a, n) \neq 1$ **then**
3: **return** COMPOSITE
4: **else if** $\left[\frac{a}{n}\right] \not\equiv a^{\frac{n-1}{2}} \pmod{n}$ **then**
5: **return** COMPOSITE
6: **else**
7: **return** PRIME

We wish to show that the algorithm is always correct when it returns "COMPOSITE" and if the input is composite, then it returns "PRIME" with probability at most 1/2. (It follows from the first statement that if the input is prime then it returns "PRIME".)

Theorem. *If the SS algorithm returns "COMPOSITE" then n is composite.*

Proof. If SS returns "COMPOSITE" then it chose an element $a \in \mathbb{Z}_n$ with either $\gcd(a, n) \neq 1$ or $\left[\frac{a}{n}\right] \not\equiv a^{\frac{n-1}{2}} \pmod{n}$, both of which are only possible if n is composite. □

For any odd number n, define the set J_n by

$$J_n = \{a \in \mathbb{Z}_n^* : \left[\frac{a}{n}\right] \equiv a^{\frac{n-1}{2}} \pmod{n}\}.$$

Observe that $J_n = \mathbb{Z}_n^*$ for prime n. Now consider n composite.

Lemma. *If n is composite then $|J_n| \leq \frac{1}{2}|\mathbb{Z}_n^*|$.*

Proof. It is clear $J_n \subseteq \mathbb{Z}_n^*$ is a group, since $\left[\frac{a}{n}\right]\left[\frac{b}{n}\right] = \left[\frac{ab}{n}\right]$. All we need to do is show that it is a proper subgroup.

Suppose that $J_n = \mathbb{Z}_n^*$. Let $p_1^{k_1} \cdots p_t^{k_t}$ be a prime factorization of n. Put $q = p_1^{k_1}$ and $m = n/q$. Since n is odd, $p_1 \neq 2$, so $\mathbb{Z}_q^*$ is cyclic. Let g be a generator of $\mathbb{Z}_q^*$. Choose an element $a \in \mathbb{Z}_n^*$ that satisfies the following

$$\begin{aligned} a &\equiv g \pmod{q} \\ a &\equiv 1 \pmod{m}. \end{aligned}$$

Such an element exists by the Chinese Remainder theorem. Observe that $a \equiv 1 \pmod{p_i}$ for all $i \geq 2$. There are now two cases to consider: either $k_1 = 1$ or $k_1 > 1$.

If $k_1 = 1$ then $q = p_1$. Since n is not prime, $m \neq 1$. Compute the Jacobi symbol for a and n:

$$\begin{aligned} \left[\frac{a}{n}\right] &= \prod_{i=1}^{t} \left[\frac{a}{p_i}\right]^{k_i} \\ &= \left[\frac{a}{q}\right] \prod_{i=2}^{t} \left[\frac{a}{p_i}\right]^{k_i} \\ &= \left[\frac{g}{q}\right] \prod_{i=2}^{t} \left[\frac{1}{p_i}\right]^{k_i} \\ &= \left[\frac{g}{q}\right]. \end{aligned}$$

Because q is prime the Jacobi symbol reduces to the Legendre symbol. The generator of $\mathbb{Z}_q^*$ cannot be a quadratic residue making $\left[\frac{a}{n}\right] = \left[\frac{g}{q}\right] = -1$. Since $J_n = \mathbb{Z}_n^*$ by assumption, we have

$$a^{\frac{n-1}{2}} \equiv -1 \pmod{n}.$$

And since m divides n,

$$a^{\frac{n-1}{2}} \equiv -1 \pmod{m}.$$

But this contradicts our choice $a \equiv 1 \pmod{m}$.

Now suppose $k_1 > 1$. By assumption $J_n = \mathbb{Z}_n^*$. Thus,

$$a^{\frac{n-1}{2}} \equiv \pm 1 \pmod{n}$$

squaring both sides

$$a^{n-1} \equiv 1 \pmod{n}$$

and substituting for a (since $q|n$)

$$g^{n-1} \equiv 1 \pmod{q}.$$

Since g is a generator for $\mathbb{Z}_q^*$ its order is $\phi(q)$. By the above relation $\phi(q)|(n-1)$. Since $k_1 \geq 2$ the prime p_1 divides $\phi(q)$. Thus p_1 divides $n-1$. But p_1 also divides n giving a contradiction since no prime can divide both n and $n-1$. □

Theorem. *Given an odd integer n, the algorithm will always return "PRIME" if n is prime and will return "COMPOSITE" with probability at least $1/2$ if n is composite.*

Proof. If n is composite then the algorithm will return "PRIME" if it chooses an element $a \in J_n$. Since a is chosen uniformly at random, the probability of making a bad choice for a is

$$\frac{|J_n|}{|\mathbb{Z}_n| - 1} < \frac{|J_n|}{|\mathbb{Z}_n^*|} \leq \frac{\frac{1}{2}|\mathbb{Z}_n^*|}{|\mathbb{Z}_n^*|} = \frac{1}{2}.$$

□

0.13 Elliptic Curves

The previous algorithm tests whether a number is prime or not, but it makes a one-sided error with its answers. Hence, a better name for it may be a composite-prover, since that answer is always correct. The Goldwasser-Kilian algorithm provides a proof certificate along with an affirmative answer. It does this by relating the primeness of the integer in question to the primeness of a smaller number. After enough recursive steps the numbers are small enough to be checked for primeness using trial-division or other basic tests.

Given a field F whose characteristic is not 2 or 3, and two elements $A, B \in F$ with $4A^3 + 27B^2 \neq 0$ we can look at solutions in F to the equation $y^2 = x^3 + Ax + B$. These solutions will be in two variables (x, y), and amazingly enough, it is possible to put a group structure after including one extra element I which represents the point at "infinity". The intuition

behind the new group operation on these elements is very geometrical, but it can also be described algorithmically.

Given an elliptic curve (A, B) and a field F let $E_{A,B}(F)$ represent the associated elliptic curve group. Since we will be working with the fields $\mathbb{Z}_p$ with p prime, we will shorten this notation to $E_{A,B}(p)$. For some reason it is standard to write $\#_p(A, B)$ instead of $|E_{A,B}(p)|$ to represent the cardinality of the elliptic group.

There is also a notion of multiplication. Given an integer $q > 0$ and element $L \in E_{A,B}(n)$ define

$$[q]L = \begin{cases} L & \text{if } q = 1, \\ [q/2](L + L) & \text{if } q \text{ is even}, \\ L + [q - 1]L & \text{if } q \text{ is odd}. \end{cases}$$

Similar to the above approaches, given an integer n the idea is to associate the question of n being prime to a property of an associated group. In this case the group we associate is some elliptic curve group (A, B) over $\mathbb{Z}_n$. There are many possible groups, we pick one for which $\#_p(A, B) = 2q$ where q is prime. How do we know q is prime? We run Solovay-Strassen on it enough times to get a high probability of primeness. Keep in mind that elliptic curves need to be defined over a field, and we don't know if $\mathbb{Z}_n$ is a field or not—indeed, if we did we would already know whether or not n is prime. We just assume $\mathbb{Z}_n$ is a field and upon encountering an inconsistency we know our assumption was wrong. At this stage it might be better to call $E_{A,B}(n)$ a **pseudocurve**, and this is the standard terminology for such things. The main theorem behind the algorithm is thus:

Theorem (Goldwasser-Kilian). *Let $n > 1$ be an integer coprime to 6, (A, B) be a pseudocurve over $\mathbb{Z}_n$, and $L \in E_{A,B}(n)$ with $L \neq I$. Suppose $[q]L = I$ for some $q > n^{1/2} + 2n^{1/4} + 1$. Then if q is prime, n is prime.*

The resulting algorithm is described by Richard Crandall and Carl Pomerence in their book *Prime Numbers* as follows.

> Given a nonsquare integer $n > 2^{32}$ strongly suspected of being prime (in particular, $\gcd(n, 6) = 1$ and presumably n has already passed a probable prime test), this algorithm attempts to reduce the issue of primality of n to that of a smaller number q. The algorithm returns either the assertion "n is composite"

or the assertion "If q is prime then n is prime," where q is an integer smaller than n.