1 Lecture 15, February 3, 2000

We continue with algorithms in number theory, and discuss some cryptographic applications.

1.1 Chinese Remainder formula

Let $n = n_1 n_2 \ldots n_k$, where the n_i are relatively prime. Then Z_n can be viewed as a direct product of the Z_{n_i}, and Z_n^* can be viewed as a direct product of $Z_{n_i}^*$. Every $a \in Z_n$ can be represented as a k-tuple (a mod n_i). It is not hard to see that this map is a bijection. Computing the inverse of this bijection can be done as follows. Let $m_i = n/n_i$ and let $c_i = m_i((m_i)^{-1} \bmod n_i)$. Observe that $c_i = 1 \bmod n_i$ and $c_i = 0 \bmod n_j$ for $j \neq i$. Then a satifying $a \bmod n_i = a_i$ can be computed as $a = \sum c_i a_i \bmod n$.

1.2 Primality testing

Every element $a \in Z_n^*$ generates a cyclic subgroup. The *order* of an element a is the least value t such that $a^t = e$, which is also equal to the order of the subgroup generated by a. Hence $t|\phi(n)$.

This suggests the following compositeness test for odd n. Compute $2^{(n-1)}$ modulo n (using repeated squaring). If n is prime than the answer is necessarily 1 (because then $\phi(n) = n - 1$). If n is composite, the output would be 1 only if the order of 2 divides $n - 1$. This might happen, but is pretty rare, and for most composites will provide a certificate for their compositeness.

For some groups, a single element (called a *generator*, or a *primitive root*) generates the whole group. Such a group is called cyclic. It is known that $(Z_n^*, \cdot_n)$ is cyclic iff $n \in \{2, 4, p^e, 2p^e\}$ where p is an odd prime and $e \geq 1$.

In Z_n^*, $+1$ has at least two square roots, ± 1. If Z_n^* is cyclic, let g be a generator for Z_n^*, and then the only nontrivial square root of unity is $g^{(n-1)/2} = -1$. Hence $+1$ has exactly two square roots. If n is a product of two relatively primes p and q, then necessarily $+1$ has at least four square roots, corresponding to the four possibilities for the set of equations $a = \pm 1 \bmod p$ and $a = \pm 1 \bmod q$. This gives the following distinction between primes and composites. For $a \in Z_n^*$ let t be such that $a^t = 1$, and assume that $2|t$. Then for prime n necessarily $a^{t/2} = \pm 1$, whereas if n has at least two coprime factors, then for some a, t as above, $a^{t/2} \neq \pm 1$.

This leads to the following (Miller-Rabin) primality test for n. Let t be the largest integer such that $(n - 1) = 2^t k$. Choose a at random and use repeated squaring to compute the powers $b_i = a^{k2^i}$ for all $0 \leq i \leq t$. If none of them is equal 1 modulo n then n is composite. If for some i, $b_i \neq \pm 1$ and $b_{i+1} = 1$ then n is composite. It can be shown that when n is the product of two relatively prime numbers, then the fraction of $a \in Z_n^*$ on which the test recognizes compositeness is at least $3/4$. This leads to a randomized algorithm that with high probability outputs a certificate for compositeness. There are other more complicated algorithms that with high probability output a certificate for primality. It is not known whether there are deterministic algorithms that check primality, though it is reasonable to conjecture that there are such algorithms.

1.3 Secret key exchange

A *oneway function* is a function that is easy (polynomial time) to compute but hard to invert. If P=NP then oneway functions do not exist. However, relative to our current state of knowledge, random functions are abundant. Number theory offers a variety of (conjectured) oneway functions that have extra special properties that make them useful in cryptographic applications. For example, let p be a large prime, and g a generator for p. Then computing g^a is easy (by repeated squaring), but no efficient algorithm is known for the *discrete log* problem (given x, find a satisfying $g^a = x$). Hence the function $f(a) = g^a \bmod p$ can be thought of as a oneway function. It also enjoys an extra property – it is commutative in the sense that $(g^a)^b = (g^b)^a$.

Assume that Alice and Bob want to exchange messages over a communication channel without Eve being able to figure out the contents of the messages, depsite her ability to passively wiretap the communication channel. If Alice and Bob share a *private key* (a random string unknown to Eve) they can do so (using it as a one time pad, or more realistically, using is as a key to a private key encryption scheme such as DES, assuming that Eve has limited computational resources). A secret key exchange is a protocol that allows Alice and Bob to jointly generate a common private key without Eve being able to figure out what this key is. Diffie and Hellman, who initiated public key cryptography, suggested the following scheme.

Alice and Bob publicly agree on a prime p and generator g (these are no secret from Eve). Alice privately picks a random a, computes $x = g^a$, and sends x to Bob. Bob privately picks a random b, computes $y = g^b$, and sends y to Bob. The common secret key is then $c = (g^a)^b$, which Alice can compute as $y^a \bmod p$ and Bob can compute as $x^b \bmod p$. Currently, there is no known polynomial time procedure that Eve can apply so as to compute c from (p, g, x, y).

We now show a procedure for picking p and g for the Diffie-Helman scheme. To pick a large prime p, one can use random sampling. We use the known estimate of $n/\ln n$ as the number of primes up to value n. Hence picking a k bit number n at random one has probability $\Omega(1/\log k)$ that n is prime. Testing it for primality one will with high probability discard n if n is composite, and thereafter a new number is chosen and the primality test repeated. After $O(\log k)$ expected trials, a prime p is found. Moreover, the prime p is uniformly distributed over k-bit primes.

Picking a generator is more difficult. Even checking whether g is a generator for Z_p^* may be a difficult task. One way around this problem is as follows. Rather than pick p at random, pick a pair q and $p = 2q + 1$ such that both p and q are prime. It is reasonable to expected that such a pair will be found after $O((\log k)^2)$ random values of q are tried. Now the complete factorization of $(p - 1) = 2q$ is known. For a generator g, all its powers g^t with t relatively prime to $(p - 1)$ are also generators, hence roughly half the numbers in Z_p^* are generators. Picking x at random, we can test whether it is a generator by checking that $x^2 \neq 1$ and $x^q \neq 1$ (because the order of x must divide $|Z_p^*|$).

1.4 RSA

Oneway number theoretic functions sometimes posses a *trapdoor* – some auxiliary piece of information that makes the computation of the inverse easy. Often, the trapdoor information is the prime factorization of a large number n. This information may be kept secret even if n is public, as there are no known efficient algorithms to factor large numbers.

RSA is a public key cryptosystem suggested by Rivest, Shamir and Adleman. It can be used either as an encryption scheme or as a signature scheme.

Alice picks two large random primes p and q and computes $n = pq$. It also picks a random e relatively prime to $\phi(n) = (p-1)(q-1)$. It computes d such that $ed = 1 \bmod \phi(n)$ (e.g., using the extended GCD algorithm). Alice publishes (n, e) as its public key and keeps d (the so called *trapdoor*) as its secret key. (The values of p and q may be discarded at this point. However, Alice may want to keep them so as to somewhat speedup subsequent computations, by performing them modulo p and q and using Chinese remaindering so as to compute the result modulo n.) To encrypt a message m (a message is encoded as a number modulo n) and send it to Alice, Bob computes $m^e \bmod n$ and sends the result c to alice. To decrypt the encrypted message c, Alice computes $c^d \bmod n = m$.

There is no known algorithm to compute eth roots modulo n when n is the product of two large random primes. Hence anyone not possessing the trapdoor information (either d, or p and q) will find it intractable to decrypt messages sent to Alice.

RSA can be used as a digital signature scheme too. To sign m, Alice computes $s = m^d \bmod n$, and sends (m, s). Anyone can verify Alice's signature by checking that $m = s^e \bmod n$. Only Alice can sign because only she can compute eth roots modulo n.

1.5 Warning

Naive use of the cryptographic schemes above will often not result in a secure cryptosystem due to reasons that are beyond the scope of this lecture (partial information about encrypted messages may leak, for some messages signatures will be forgeable, etc., not to mention the risk of violating patent rights).

Cryptography is an area that makes use of negative results about algorithms – nonexistence of efficient algorithms for a problem (such as discrete log and factoring) is what makes the associated cryptographing schemes secure. Another warning about computational complexity based cryptography is that there are virtually no lower bounds known for the complexity of inverting oneway functions, and dramatic improvements in algorithmic theory for number theoretic functions may well show that schemes such as those described above are not secure.