Algebraic Cryptography
Exam 1 Review

For the first exam you should be able to do the following tasks:

- Explain the difference between symmetric and asymmetric ciphers, and be able to give examples of each.
- State several of the basic tasks of public key cryptography (message transmission, authentication, signatures, key exchange, secret sharing), and be able to provide examples of each of these in action (RSA, hashes with salts, DSA/RSA, Diffie–Hellman, Chinese Remainder Theorem / linear equations), along with how they work.
- Explain why it is important to store only hashed passwords for authentication, and then on top of this, why it is important to use salts.
- Explain how any public key transmission scheme with commutative encryption/decryption operations yields a signature scheme.
- Explain how to generate RSA keys (note: you would not need to explain why we can do primality testing fast).
- Explain why it is important that users generate *distinct* primes (i.e., different primes than others users) for use in their RSA keys.
- Be able to do any of the problems on the homework.
- Establish Big-O (or $\asymp$) time/length estimates for the bit operations/size of a given problem.
- Determine the time complexity of familiar algorithms (integer multiplication, gcd computation, modular exponentiation)
- Reduce one decision problem to another.
- Explain how the integer factorization decision problem reduces to the integer factorization search problem.
- Show that certain problems have polynomial time algorithm.
- Show that certain decision problems reduce to others in polynomial time.