Practice Questions for Exam 1
(Crypto Basics)

Question 1-Crypto:

Recall that a *symmetric-key cryptosystem* consists of three functions: a *key generator* G, an *encryption function* E, and a *decryption function* D. For any pair of users, say Alice (A) and Bob (B), G takes as input a string of random bits and produces as output a *shared key* K_{AB}. Either Alice or Bob can take a *plaintext message* x and produce a *ciphertext message* $y \longleftarrow E(x, K_{AB})$. The ciphertext y can be sent over an insecure channel, and the recipient can recover $x \longleftarrow D(y, K_{AB})$.

a) Briefly explain three basic requirements for such a system to be secure.

b) Why isn't symmetric-key cryptography sufficient as a foundation for secure Internet communication and, in particular, for secure Web-based commerce?

Question 2-Crypto:

Recall that a *public-key cryptosystem* consists of three functions: a *key generator* G, an *encryption function* E, and a *decryption function* D. Any user, say Alice (A), can use the key generator to generate a key pair (PK_A, SK_A), in which PK_A is Alice's *public key*, and SK_A is her *secret key*; she can then publish her name and public key in a directory. Subsequently, *anyone* who wants to send a private message x to Alice can look up PK_A in the directory, compute a ciphertext $y \longleftarrow E(x, PK_A)$, and send y to Alice over an open line. Only Alice can compute $x \longleftarrow D(y, SK_A)$, because only she knows the decryption key SK_A that corresponds to encryption key PK_A. Note that there is no need for a key-distribution center, as there is when one is using a symmetric-key cryptosystem, but there is a need for public-key directories.

a) What is a public-key *certificate*? What is the problem with public-key directories that certificates address?

b) In general, public-key cryptosystems are not fast enough for users who need to exchange long streams of traffic. How can they be used in conjunction with symmetric-key cryptosystems, which are faster, in order to send long streams securely?

Question 3-Crypto:

a) What is the *RSA public-key signature scheme*? That is, what are its constituent key-generation, signing, and verification functions?

b) What function must be *one-way* if this signature scheme is to be secure?

c) Note that the RSA functions can be used both as a public-key cryptosystem and as a public-key signature scheme. The decryption function and secret key of the cryptosystem are used to generate signatures, and the encryption function and public key of the cryptosystem are used to verify signatures. Is this true in general? That is, can any public-key cryptosystem be turned into a public-key signature scheme simply by reversing the roles of encryption and decryption?

Question 4-Crypto:

a) What is a *one-way hash function*?

b) How are one-way hash functions typically used in conjunction with public-key signature schemes, and why?