Highlights of Chapters 1-2.

Chapter 1: An Introduction to Cryptography

1. **Number theory concepts:**
   (a) Division Theorem: $a, b \in \mathbb{Z}$, $b \neq 0$, then there are $q, r \in \mathbb{Z}$ such that $a = qb + r$ with $0 \leq r < b$.
   (b) Definition of divisibility: $b \mid a$ if there is $k \in \mathbb{Z}$ such that $a = bk$.
   (c) Greatest common divisor: $d = \gcd(a, b)$ if (a) $d \mid a$ and $d \mid b$, and (b) if $e \mid a$, $e \mid b$ then $e \leq d$.
   (d) Euclid's algorithm (based on repeated long division).
   (e) Bezout's identity: $ax + by = c$ has solutions $x, y \in \mathbb{Z}$ if and only if $\gcd(a, b) \mid c$.
      - if (x_0, y_0) is one solution for $ax + by = c$, then all the solutions are given by $x = x_0 + \frac{bk}{d}$, $y = y_0 - \frac{ak}{d}$, for all $k \in \mathbb{Z}$, where $d = \gcd(a, b)$.
   (f) Fundamental theorem of arithmetic
      i. Every number has a factorization as a product of primes $n = p_1^{e_1} \cdots p_t^{e_t}$, where $p_1 < p_2 < \cdots < p_t$ are primes and $e_t \geq 1$.
      ii. The factorization into primes is unique up to a reordering of prime-power factors.
   (g) Definition of congruence: $a \equiv b \bmod m$ if $m \mid a - b$.
   (h) Properties of congruences, e.g., if $a \equiv b \bmod m$, then $a^k \equiv b^k \bmod m$, for all $k \geq 1$.
   (i) The congruence $ax \equiv b \bmod m$ has a solution if and only if $ax + my = b$ has a solution $x, y \in \mathbb{Z}$.
   (j) Definition and operations on $\mathbb{Z}/m\mathbb{Z} = \{0, 1, \ldots, m - 1 \bmod m\}$.
   (k) Units in $\mathbb{Z}/m\mathbb{Z}$ is the set $(\mathbb{Z}/m\mathbb{Z})^\times = \{a \bmod m : \gcd(a, m) = 1\}$. There are $\varphi(m)$ units.
   (l) Fast powering algorithm.
   (m) Fermat's little theorem: if p is prime, then $a^{p-1} \equiv 1 \bmod p$ for all $a \in \mathbb{Z}$ with $\gcd(a, p) = 1$.
   (n) Euler's theorem: $a^{\varphi(m)} \equiv 1 \bmod p$ for all $a \in \mathbb{Z}$ with $\gcd(a, m) = 1$.
   (o) The multiplicative order of $a \bmod m$ is the smallest $n \geq 1$ such that $a^n \equiv 1 \bmod m$, for $a \in \mathbb{Z}$ with $\gcd(a, m) = 1$. The order always divides $\varphi(m)$ (divides $p - 1$ if $m = p$ is prime).
   (p) A primitive root modulo p is $g \bmod p$ such that its multiplicative order is exactly $p-1$. There are $\varphi(p - 1)$ primitive roots modulo p.

2. **Cryptography:**
   (a) Substitution ciphers and Caesar (shift) ciphers.
   (b) Basics of frequency analysis to break a substitution cipher.
   (c) Symmetric ciphers: $e_k \colon M \to C$ and $d_k \colon C \to M$ such that $d_k(e_k(m)) = m$ for every $m \in M$ and $k \in K$.
   (d) Attacks: known plaintext attack, chosen plaintext attack.
   (e) Encoding schemes, blocksize.
   (f) Affine ciphers: $e_{(k_1,k_2)}(m) \equiv k_1 \cdot m + k_2 \bmod p$ and $d_{(k_1,k_2)}(c) \equiv k_1^{-1} \cdot (c - k_2) \bmod p$.
   (g) Hill ciphers (as affine cipher, but k_1 is a matrix and k_2 a vector).
   (h) Asymmetric ciphers (public key cryptography): $e_{k_{\text{pub}}} \colon M \to C$ and $d_{k_{\text{priv}}} \colon C \to M$ such that $d_{k_{\text{priv}}}(e_{k_{\text{pub}}}(m)) = m$ for every $m \in M$ and $k \in K$.

Chapter 2: Discrete Logarithms and Diffie-Hellman

1. **Introduction to Group Theory:**
   (a) Definition of group $(G, *)$.
   (b) Examples of groups.
   (c) The DLP (discrete logarithm problem) over a group G.
   (d) Order of an element in a group.
   (e) Lagrange's theorem: the order of an element a in a group G divides the order (size) of the group G.

2. **Cryptography:**
   (a) The DLP (discrete logarithm problem) over $\mathbb{F}_p^*$: given g and h find x such that $g^x \equiv h \bmod p$.
   (b) The index function in base g mod p, a.k.a. the logarithm function in base g mod p.
   (c) The Diffie-Hellman key exchange:
   - Fix p, and g mod p.
   - Alice picks $a \in \mathbb{Z}$, computes $A \equiv g^a \bmod p$, sends A to Bob.
   - Bob picks $b \in \mathbb{Z}$, computes $B \equiv g^b \bmod p$, sends B to Alice.
   - Compute key $k \equiv B^a \equiv A^b \bmod p$.

   (d) The Elgamal public key cryptosystem:
   - Fix p, and g mod p.
   - Alice picks private key $a \in \mathbb{Z}$, public key $A \equiv g^a \bmod p$, publish p, g, A.
   - Bob picks k mod p, encrypts message m as $(c_1, c_2) \equiv (g^k, mA^k) \bmod p$, and send.
   - Alice computes $x \equiv (c_1)^{-a} \bmod p$, and the message is $m \equiv c_2 \cdot x \bmod p$.

   (e) Order notation (big-O notation): we say $f(x) = O(g(x))$ if there are constants c, C such that $f(x) \leq c \cdot g(x)$ for all $x \geq C$.
   (f) Proposition: if the limit $\lim_{x\to\infty} f(x)/g(x)$ exists and it is finite, then $f(x) = O(g(x))$.
   (g) Brute force on DLP over $\mathbb{F}_p^*$ can be done in $O(k^2 2^k)$ basic steps, when the input is $O(k)$ bits long.
   (h) Collision algorithm to solve DLP: Shank's babystep-giantstep algorithm, solves DLP in $O(k2^{k/2})$ basic steps.
   - Fixed g, h, p find x such that $g^x \equiv h \bmod p$.
   - Let $n = 1 + \lfloor \sqrt{N} \rfloor$, where N is the order of g mod p (so $N = p - 1$ if g is a primitive root).
   - Compute two lists:
     - List 1: $1, g, g^2, \ldots, g^n \bmod p$.
     - List 2: $h, hu, hu^2, \ldots, hu^n \bmod p$, where $u \equiv g^{-n} \bmod p$.
   - Find a match in lists, so that $g^i \equiv hu^j \bmod p$.
   - Then $x = i + jn$ is a solution for $g^x \equiv h \bmod p$.