Topics in Cryptography — G22.3033-010 Fall 2002 — Problem Set 1
Due: Monday, October 7

Some problems refer to the Primer, which is the document *A Primer on Algebra and Number Theory for Computer Scientists (version 0.1)* available on the course web page.

For all problems, you may use as the “classical” algorithms for integer arithmetic, whose running times are as in Theorem 3.2 of the Primer.

1. Give a detailed proof of Theorem 3.5 in the Primer.

2. Prove that the Chinese Remainder Theorem (Theorem 2.6 in the Primer) can be made computationally effective, in the following sense:

   given integers \( n_1, \ldots, n_k \), and \( a_1, \ldots, a_k \), with \( n_i > 1 \), \( \gcd(n_i, n_j) = 1 \) for \( i \neq j \), and \( 0 \leq a_i < n_i \), we can compute \( x \) such that \( 0 \leq x < n \) and \( x \equiv a_i \pmod{n_i} \) in time \( O(L(n)^2) \), where \( n = \prod_i n_i \).

3. (a) Let \( p \) be a prime. We say that a sequence \((\alpha_i)_{i \geq 0}\), with each \( \alpha_i \in \mathbb{Z}_p \), satisfies a linear recurrence of order \( k \) if there are elements \( c_1, \ldots, c_k \in \mathbb{Z}_p \) such that

\[
\alpha_n = \sum_{i=1}^{k} c_i \alpha_{n-i} \quad \text{for} \quad n \geq k.
\]

   Given as input the number \( p \), the elements \( c_1, \ldots, c_k \), “starting values” \( \alpha_0, \ldots, \alpha_{k-1} \), and an integer \( n \geq 0 \), show how to compute \( \alpha_n \) in time \( O(k^3L(n)L(p)^2) \).

   (b) Improve this to \( O(k^2L(n)L(p)^3) \).

4. (a) Suppose you are given a prime \( p \), the prime factorization \( p - 1 = \prod_{i=1}^{r} q_i^{e_i} \), and an element \( \alpha \in \mathbb{Z}_p^* \). Show how to compute the order of \( \alpha \) in time \( O(rL(p)^3) \).

   (b) Improve this to \( O(L(r)L(p)^3) \).

For the next problem, we need the following notions from complexity theory:

- We say problem \( A \) is deterministic poly-time reducible to problem \( B \) if there exists a deterministic algorithm \( R \) for solving problem \( A \) that makes calls to a subroutine for problem \( B \), where the running time of \( R \) (not including the running time for the subroutine for \( B \)) is polynomial in the input length.

- We say that \( A \) and \( B \) are deterministic poly-time equivalent if \( A \) is deterministic poly-time reducible to \( B \) and \( B \) is deterministic poly-time reducible to \( A \).

5. Show that the following problems are deterministic poly-time equivalent:

   (a) Given a prime \( p \), a prime \( q \) that divides \( p - 1 \), an element \( \gamma \in \mathbb{Z}_p^* \) of order \( q \), and two elements \( \alpha, \beta \in \langle \gamma \rangle \), compute \( \gamma^{x'y} \), where \( x = \log_{\gamma} \alpha \) and \( y = \log_{\gamma} \beta \). This problem is called the Diffie-Hellman problem.

   (b) Given a prime \( p \), a prime \( q \) that divides \( p - 1 \), an element \( \gamma \in \mathbb{Z}_p^* \) of order \( q \), and an element \( \alpha \in \langle \gamma \rangle \), compute \( \gamma^{x^2} \), where \( x = \log_{\gamma} \alpha \).

   (c) Given a prime \( p \), a prime \( q \) that divides \( p - 1 \), an element \( \gamma \in \mathbb{Z}_p^* \) of order \( q \), and two elements \( \alpha, \beta \in \langle \gamma \rangle \), with \( \beta \neq [1 \pmod{p}] \), compute \( \gamma^{x'y} \), where \( x = \log_{\gamma} \alpha \) and \( y' \) is the multiplicative inverse modulo \( q \) of \( y = \log_{\gamma} \beta \).

   (c) Given a prime \( p \), a prime \( q \) that divides \( p - 1 \), an element \( \gamma \in \mathbb{Z}_p^* \) of order \( q \), and an element \( \alpha \in \langle \gamma \rangle \), with \( \alpha \neq [1 \pmod{p}] \), compute \( \gamma^{x'} \), where \( x' \) is the multiplicative inverse modulo \( q \) of \( x = \log_{\gamma} \alpha \).