Assignment 7 – Due Friday, November 11

The first two problems are from the book, and involve doing calculations for the two main algorithms in this chapter, Diffie-Hellman Key Exchange and ElGamal encryption:

1. Page 324, Problem 10.1
2. Page 325, Problem 10.6

The remaining problems look at how public parameters (q and α) can be found that work for Diffie-Hellman and ElGamal. The book doesn't describe how this is done, but the problems below lead you through a technique for doing this.

3. It's not immediately obvious how to find a generator for Z_q^*: You can test if a value α is a generator by taking all powers of α mod q, which works if q is small (like 11), but is obviously infeasible if q is around 2^{1024}, which is the size we use in practice. However, if we know the factorization of $q-1$, then we can test α more efficiently:

   (a) Recall that the powers of any element $a \in Z_q^*$ form a cyclic subgroup of Z_q^*. The notation we used for this is $\langle a \rangle = \{a^0, a^1, \cdots, a^{k-1}\}$, where k is the smallest positive integer such that $a^k \equiv 1 \pmod{q}$. The value k is called the *order* of a in Z_q^*. We mentioned in a previous class that the order of any element must divide the size of the group (extra credit if you can tell why this is true!). What is the size of Z_q^* in terms of q? Since the size of the group is also the Euler Totient function $\phi(q)$ (which is not the answer to the previous question), we will use this notation in the rest of the problem to refer to the size of Z_q^*.

   (b) In the problems you did from the book (10.1 and 10.6), the modulus q is 71. What is the prime factorization of $\phi(71)$?

   (c) If α is *not* a generator of Z_q^*, it means that the order of α in Z_q^* is smaller than $\phi(q)$ but divides evenly into $\phi(q)$. Let $p_1, p_2, \ldots, p_m$ denote the prime factors of $\phi(q)$. If α is not a generator of Z_q^* it must be the case that the order of α divides evenly into at least one of the following: $\phi(q)/p_1, \phi(q)/p_2, \ldots, \phi(q)/p_m$. Prove that this is true.

   (d) Explain why the observation in part (c) implies that α is a generator if and only if none of the following are congruent to 1 modulo q:

$$\alpha^{\phi(q)/p_1} \quad \alpha^{\phi(q)/p_2} \quad \cdots \quad \alpha^{\phi(q)/p_m}$$

(e) Compute the formulas given in part (d) for $\alpha = 7$ and $q = 71$ and explain how the results show that 7 is a generator of Z_q^*. (*Hint:* This involves computing modular powers — an easy way to do this is to use Mathematica and the built-in `PowerMod` function, like we have done for in-class examples. Mathematica is available on all UNCG lab computers, and is available remotely on the `linux.uncg.edu` system.)

(f) Repeat part (e), but with $\alpha = 5$. Is 5 a generator?

(g) Use these facts to create an algorithm for testing whether an element α is a generator of Z_q^*, when you are given both q and its prime factorization.

(h) One problem with this technique is that you must know the factorization of $\phi(q)$ in order to use this. It turns out that there are values called "safe primes" that are of the form $q = 2p + 1$, where p and q are both prime.[1] How can you find a large, random safe prime? (The algorithm for this will be randomized, and it's probably not clear to you how many randomized tests you need in order to find a safe prime, but you should at least be able to come up with the algorithm.)

(i) If q is a safe prime, what is the prime factorization of $\phi(q)$?

[1] Math trivia: In the formula given above, any p that satisfies this condition is called a "Sophie Germain prime," named after French mathematician Marie-Sophie Germain (1776–1831).