Homework 10

Due Thursday, November 19

Implementation Part

1. Implement Pollard's ρ algorithm to solve the discrete log problem $g^x = h \mod p$. You should define an algorithm `PollardRhoLog(g,h,p)` which takes as input a prime p and $g, h \in \mathbb{F}_p^*$ with g a primitive root, and outputs the solution to $g^x = h \mod p$. Your algortithm should use as a mixing function:
$$f(x) \equiv \begin{cases} gx \mod p & 0 \le x < p/3 \\ x^2 \mod p & p/3 \le x < 2p/3 \\ xh \mod p & 2p/3 \le x < p \end{cases},$$
And should compute $x_i = \underbrace{(f \circ f \circ \cdots \circ f)}_{i \text{ times}}(x)$ and $y_i = x_{2i}$. Exploit that each $x_i = g^{\alpha_i} h^{\beta_j}$ and similarly for the y_i. You will need to keep track of these exponents as well, but you should not be making a list (we described how to do this in class). (Hint: A collision will let you compute the discrete log of a power of h. Passing from this to the discrete log of h should look a lot like HW7 4(c)).

2. Use `PollardRhoLog` to solve the following..
   (a) $3^t \equiv 5 \mod 17$.
   (b) $19^t \equiv 24717 \mod 48611$. (Note, this is Example 5.52 in the book so you can double check if your algorithm worked).
   (c) $29^t \equiv 5953042 \mod 15239131$.
   (d) $2^t \equiv 2598854876 \mod 2810986643$

3. This problem goes hand in hand with Problem 4 in the written part of the assignment, implementing Pollard's ρ method to factor large numbers.
   (a) Program an algorithm `PollardRhoFactor(N,f,x = 2)` which implements the algorith described in Problem 4 to find a nontrivial factor of N. It should take as input a large number N, a mixing function f, and an initial value for the mixing function x (which will initialize to 2 if not given). It should also print the number k of steps it took to find the nontrivial factor and the ration $\sqrt{N}/k$ (for our analysis in Problem 4).
   (b) Test out `PollardRhoFactor` with mixing function $f(x) = x^2 + 1$ to find a nontrivial factor of:
      i. 2201
      ii. 9409613
      iii. 1782886219
   (c) Repeat part (b) with a mixing function of $f(x) = x^2 + 2$.
   (d) Repeat part (b) with a mixing function of $f(x) = x^2$.
   (e) Repeat part (b) with a mixing function of $f(x) = x^2 - 2$.
   (f) Test out `PollardRhoFactor` on some prime numbers.
   (g) Write a function `PollardRhoFactorize(N,f,x=2)` which repeatedly uses `PollardRhoFactor` to find a complete factorization of N.

Written Part

4. This problem goes hand in hand with Problem 3 in the impelementation part of this assignment. We describe how (the abstact version of) Pollard's ρ method can be used to factor large numbers N relatively quickly. It works best when N has a relatively small prime factor p. We first describe the method. Suppose you have a mixing function:

$$f : \mathbb{Z}/N\mathbb{Z} \to \mathbb{Z}/N\mathbb{Z}.$$

Let $x_0 = y_0 \in \mathbb{Z}_N\mathbb{Z}$, and compute $x_{i+1} = f(x_i)$ and $y_{i+1} = f(f(y_i))$. At each step compute:

$$g_i = \gcd(|x_i - y_i|, N).$$

   (a) Suppose f is sufficiently random and let p be the smallest prime divisor of N. Show that with high probability we find some $g_k = p$ for $k = \mathcal{O}(\sqrt{p})$.
   (b) Compare what happened in 3(b) and 3(c). Did one have a faster run time? Why?
   (c) Explain what happened in 3(d) when the mixing function was $f(x) = x^2$.
   (d) Explain what happened in 3(e) when the mixing function was $f(x) = x^2 - 2$.
   (e) Explain what happened in 3(f) when N was prime.

In class we stated and proved the forward direction of the following theorem.

Theorem 1. *Fix a cryptosystem with $\#\mathcal{M} = \#\mathcal{C} = \#\mathcal{K}$. The system has perfect secrecy if and only if the following two conditions hold.*

(1) Each key $k \in \mathcal{K}$ is used with equal probability.

(2) For each plaintext $m \in \mathcal{M}$ and ciphertext $c \in \mathcal{C}$ there exists a unique key $k \in \mathcal{K}$ with $e_k(m) = c$.

5. Complete the proof of Theorem 1 by proving the *only if* direction. That is, assuming conditions (1) and (2) hold, show the system has perfect secrecy.

6. Prove the following identities for binomial coefficients. (Parts (c) and (d) generalize computations in HW8 Problems 3(e) and 3(f)).

   (a) $\sum_{k=0}^{n} \binom{n}{k} = 2^n$
   (b) $\sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0$
   (c) $\sum_{k\geq 0} \binom{n}{2k} = 2^{n-1}$
   (d) $\sum_{k\geq 0} \binom{n}{2k+1} = 2^{n-1}$

7. Consider the elliptic curve E given by the equation $y^2 = x^3 - 2x + 4$. Let $P = (0, 2)$ and $Q = (3, -5)$.

   (a) Show $P, Q \in E$.
   (b) Compute $P \oplus Q$.
   (c) Compute $P \oplus P$.
   (d) Compute $P \oplus P \oplus P$.