MOCK FINAL EXAM

1. (a) Describe the Diffie-Hellman key exchange (over the group $\mathbf{F}_p^\times$). **(2 points)**

   (b) Let $p > 2$ be a prime number, and g be a primitive root modulo p, i.e. the discrete logarithm problem $g^x \equiv a \bmod p$ has a solution $1 \leqslant x \leqslant p-1$ for all $1 \leqslant a \leqslant p-1$. Assume there is a machine which solves the discrete logarithm problem for any input $1 \leqslant a \leqslant (p-1)/2$ in polynomial time. Prove that using this machine, the discrete logarithm problem can be solved for any input $1 \leqslant a \leqslant p-1$ in polynomial time. **(4 points)**

Solution. (a) Alice and Bob would like to agree on a residue class modulo p such that even though their whole communication is monitored by an eavesdropper, they can consider this residue class to be their secret. They publicly agree on the prime p and a coprime residue class g modulo p (preferably a primitive root, but this is not absolutely necessary).

In the first step Alice chooses $a \in \mathbf{N}$ and computes $A \equiv g^a \bmod p$; while Bob chooses $b \in \mathbf{N}$ and computes $B \equiv g^b \bmod p$. Then Alice sends A to Bob, and Bob sends B to Alice.

In the next step, Alice raises the incoming residue class B to power a modulo p; while Bob raises the incoming residue class A to power b modulo p. The pont is that they get the same residue class:

$$B^a \equiv (g^b)^a \equiv g^{ab} \equiv (g^a)^b \equiv A^b \bmod p.$$

(b) Let our algorithm be the following. Take the input a.

If $1 \leqslant a \leqslant (p-1)/2$, give it to the machine as an input. The output is $\log_g a$ by assumption, and the running time is polynomial. This was easy and from now on, we assume that we are in the complementary case $(p+1)/2 \leqslant a \leqslant p-1$.

In this second case $(p+1)/2 \leqslant a \leqslant p-1$, we compute first $b = p-a$. This is just a subtraction hence is done in polynomial time, and for the result, $1 \leqslant b \leqslant (p-1)/2$ obviously holds.

Now give b to our machine as an input. The output is $\log_g b$.

If $1 \leqslant \log_g b \leqslant (p-1)/2$, then we return $\log_g a = \log_g b + (p-1)/2$, while if $(p+1)/2 \leqslant \log_g b \leqslant p-1$, then we return $\log_g a = \log_g b - (p-1)/2$ (both computed in polynomial time). Of course, we have to prove that these are the correct results.

In any case,

$$b \equiv -a \bmod p,$$

then (using what we have learned from homework problems),

$$\log_g b \equiv \log_g a + \log_g(-1) \bmod p-1.$$

Therefore, to complete the solution, it suffices to show that $\log_g(-1) = (p-1)/2$. Clearly,

$$\left(g^{\frac{p-1}{2}}\right)^2 \equiv g^{p-1} \equiv 1 \bmod p$$

by Euler-Fermat. Then $g^{(p-1)/2}$ is ± 1 modulo p, and $g^{p-1} \equiv 1 \bmod p$. Therefore, $g^{(p-1)/2} \equiv -1 \bmod p$, and the proof is complete.

2. (a) Describe the elliptic curve ElGamal public key cryptosystem. **(2 points)**

   (b) Let $\mathbf{F}_7$ be the base field. How many points does the elliptic curve

$$\{[X,Y,Z] \in \mathbf{PF}_7^2 : Y^2Z = X^3 - XZ^2\}$$

   have? **(4 points)**

Solution. (a) Alice chooses a prime number $p > 3$, an elliptic curve E over the prime field $\mathbf{F}_p$, and a point P on the elliptic curve. She further chooses a positive integer n_A, and computes the point

$$Q = n_A P = \underbrace{P + \ldots + P}_{n_A \text{ many}}.$$

Now she publishes p, E, P, Q and keeps n_A in secret.

Anyone (say, Bob) can send her a message M (a point on the elliptic curve) using the following protocol. Bob chooses an ephemeral key $k \in \mathbf{N}$, and computes

$$C_1 = kP, \qquad C_2 = M + kQ.$$

Then he sends the pair (C_1, C_2) to Alice.

Now Alice computes $C_2 - n_A C_1$, obtaining

$$C_2 - n_A C_1 = M + kQ - n_A kP = M + kn_A P - n_A kP = M,$$

which is the original message.

(b) We know (from the general description) that there is a unique point $[0,1,0]$ satisfying that the Z-coordinate is zero. Apart from that, we can consider the affine curve

$$\{(x,y) \in \mathbf{F}_7^2 : y^2 = x^3 - x\}.$$

For $y = 0$, there are exactly three solutions: $x = 0, \pm 1$. Now let x run through $\mathbf{F}_7 \setminus \{0, \pm 1\}$ and check whether $x^3 - x$ is a square in $\mathbf{F}_7$ or not. To do this, record that the squares in $\mathbf{F}_7$ are $0,1,2,4$, and observe that for any $a \in \mathbf{F}_7^\times$, exactly one of $\pm a$ is a square. Note also that whenever we replace x with $-x$, $x^3 - x$ goes to its negative. Therefore, exactly one of ± 2 and exactly one of ± 3 gives a square. Of course, for each such (nonzero) square, there are two appropriate choices for y. Therefore, the number of such points is 4.

Together with the three points with vanishing y-coordinate, and the one point at ”infinity”, we obtain 8 points on our elliptic curve.

3. (a) Describe perfect secrecy. **(2 points)**

   (b) Let X, Y be random variables. Prove that

$$H(X,Y) \leqslant H(X) + H(Y).$$

   (4 points)

   Solution. (a) Let us denote by $\mathscr{M}, \mathscr{K}, \mathscr{C}$ the message, key and cipher sets, respectively, and let $M : \Omega \to \mathscr{M}, K : \Omega \to \mathscr{K}, C : \Omega \to \mathscr{C}$ be the random variables which choose the message, the key and the cipher at random. Then we say that the cryptosystem has perfect secrecy, if

$$\Pr(M = m) = \Pr(M = m \mid C = c)$$

   holds for all $m \in \mathscr{M}, c \in \mathscr{C}$.

   (b) We proved in the lecture that

$$H(X,Y) = H(Y) + H(X \mid Y).$$

   We also proved in the lecture that

$$H(X) \geqslant H(X \mid Y).$$

   Altogether,

$$H(X,Y) = H(Y) + H(X \mid Y) \leqslant H(X) + H(Y),$$

   and the proof is complete.

4. (a) Define key equivocation. **(2 points)**

   (b) Alice and Bob use the XOR cipher on t bits. Assuming that M and K are independent and uniformly distributed, compute the key equivocation. **(4 points)**

Solution. (a) Let us denote by $\mathscr{M}, \mathscr{K}, \mathscr{C}$ the message, key and cipher sets, respectively, and let $M : \Omega \to \mathscr{M}, K : \Omega \to \mathscr{K}, C : \Omega \to \mathscr{C}$ be the random variables which choose the message, the key and the cipher at random. Denoting by H the entropy of a random variable, the key equivocation is defined to be the conditional entropy

$$H(K \mid C).$$

(b) We proved in the lecture that when M and K are independent, then the formula

$$H(K \mid C) = H(M) + H(K) - H(C)$$

holds. We claim that $H(M) = H(C)$. Indeed, M and C are both uniform distributions on 2^t elements: this is clear by definition for M; while for C, it follows from the facts that each cipher c corresponds to 2^t pairs (m,k) via $c = e_k(m)$ and that there are 4^t possibilities for the pair (m,k), each of them having probability 4^{-t} (because of the independency of m and k). Therefore,

$$H(K \mid C) = H(K).$$

We can compute this from definition: there are 2^t keys, each of them occurs with probability 2^{-t}. Therefore,

$$H(K \mid C) = H(K) = -\sum_{j=1}^{2^t} 2^{-t} \log_2 2^{-t} = t.$$

Hence the key equivocation is t.