HAI709I : Fondements cryptographiques de la sécurité, Université de Montpellier, 2023

14/10/2024. Additional exercises to prepare for the mid-term exam.

Exercise 1. Let p be a prime number. A polynomial $f(x) = k + c_1 x + c_2 x^2$ is evaluated at pairwise distinct points a_1, a_2, a_3 modulo p,

$$\begin{aligned} s_1 &= f(a_1) \mod p, \\ s_2 &= f(a_2) \mod p, \\ s_3 &= f(a_3) \mod p. \end{aligned}$$

Find a formula that returns the value of k given a_1, a_2, a_3 and s_1, s_2, s_3 (you may use in this formula the usual arithmetic operations of addition, subtractions, multiplication, and inversion modulo p).

Exercise 2. (a) Find a quadratic polynomial $f(x) = c_0 + c_1 x + c_2 x^2$ with integer coefficients (not all coefficients are equal to 0 modulo 35) that has at least three different roots modulo 35, i.e.,

$$f(x_1) = 0 \mod 35,\ f(x_2) = 0 \mod 35,\ f(x_3) = 0 \mod 35$$

or explain why this is impossible.

(b) Find a quadratic polynomial $f(x) = c_0 + c_1 x + c_2 x^2$ with integer coefficients (not all coefficients are equal to 0 modulo 37) that has at least three different roots modulo 37, i.e.,

$$f(x_1) = 0 \mod 37,\ f(x_2) = 0 \mod 37,\ f(x_3) = 0 \mod 37$$

or explain why this is impossible.

(c) Find a quadratic polynomial $f(x) = c_0 + c_1 x + c_2 x^2$ with integer coefficients (not all coefficients are equal to 0 modulo 39) that has at least three different roots modulo 39, i.e.,

$$f(x_1) = 0 \mod 39,\ f(x_2) = 0 \mod 39,\ f(x_3) = 0 \mod 39$$

or explain why this is impossible.

Exercise 3. Suggest a polynomial time deterministic algorithm that takes as input a prime number p and a polynomial with integer coefficients

$$f(x) = c_0 + c_1 x + \ldots c_d x^d$$

(with degree $d < p$) and returns a number $a \in \{0, \ldots, p-1\}$ such that $f(a) \neq 0 \mod p$. The algorithm should run in time polynomial in the length of input, whcih is equal to $d \cdot \lceil \log_2 p \rceil$.

Exercise 4. We need to share a secret k (which is a bit string of length n) among four participants Alice, Bob, Charlie, Dan in such a way that the minimal groups that known the secret are

$$\{\text{Alice}, \text{Bob}\},\ \{\text{Alice}, \text{Charlie}\},\ \{\text{Alice}, \text{Dan}\},\ \{\text{Bob}, \text{Charlie}, \text{Dan}\}.$$

(a) Construct a secret sharing scheme with the required property.

(b) Show that in every secret sharing scheme for this problem Alice must receive a share with *at least* 2^n possible values (i.e., we cannot give to each participant only n bits of information).

Hint: Let Bob is given his secret share s_B. Can we say how many values of the secret key k are compatible with this value of s_B? Can we say how many values of Alice's secret share s_A are compatible with this value of s_B

(c)** Show that in every secret sharing scheme for this problem one of the participants must receive a share with strictly more than 2^n possible values (i.e., we cannot give to each participant a share of size exactly n bits).

Exercise 5. (a) Let $G_n : \{0,1\}^{n/2} \to \{0,1\}^n$ be a function such that for every input $\bar{x} = (x_1 \dots x_{n/2})$ taken from $\{0,1\}^{n/2}$, in the output bit string $(y_1 \dots y_n) := G_n(\bar{x})$ the first and the last bits (i.e., y_1 and y_n) are equal to each other. Show that G_n does not satisfy the definition of a pseudo-random generator.

(b) Let $G_n : \{0,1\}^{n/2} \to \{0,1\}^n$ be a function such that for every input $\bar{x} = (x_1 \dots x_{n/2})$ taken from $\{0,1\}^{n/2}$, in the output bit string $(y_1 \dots y_n)$:= $G_n(x_1 \dots x_n)$, the number of *ones* is even, i.e.,

$$y_1 \oplus y_2 \oplus \dots \oplus y_n = 0.$$

Show that G_n does not satisfy the definition of a pseudo-random generator.

(c) Let $G_n : \{0,1\}^{\sqrt{n}} \to \{0,1\}^n$ be a function such that for every input $\bar{x} = (x_1 \dots x_{\sqrt{n}})$ taken from $\{0,1\}^{\sqrt{n}}$, in the output bit string $(y_1 \dots y_n) := G(\bar{x})$ the first $\sqrt{n}$ bits (i.e., the bits $(y_1 \dots y_{\sqrt{n}})$) are exactly the same as the bits $(x_1 \dots x_{\sqrt{n}})$. In other words, $(x_1 \dots x_{\sqrt{n}})$ is a prefix of $(y_1 \dots y_n)$. Prove that G_n does not satisfy the definition of a pseudo-random generator.

Exercise 6. (a) If $\mathrm{P} = \mathrm{NP}$ then there exists a deterministic algorithm that takes as input the binary expansion of a number n and decides in polynomial time whether n is prime or not.

(b) If $\mathrm{P} = \mathrm{NP}$ then there exists a deterministic algorithm that takes as input the binary expansion of a number n and finds in polynomial time all prime factors of n.