COMS BC3262 April 15, 2026

Problem Set 5

Professor: Eysa Lee *Due: April 30, 2026*

Guidelines. Solutions will be graded for correctness and clarity. If you do not know how to solve a problem, you may write **"I don't know how to do this"** to receive 20% credit for any problem not marked as extra credit. In your proofs, justify every step and, if needed, write down definitions/theorems from class or the textbook and that you rely on. When refuting a claim, provide a counterexample and show that it does not satisfy the required definition.

Collaboration. You are allowed to discuss problems with other students in the class. However, you should write up your solutions on your own and may not read or copy the solutions of others. You should try solving each problem on your own before discussing with others. **If you work with others on a problem, you must note with whom you discussed the problem at the beginning of your solution write-up.** For example, "I discussed Problem 1 with Alice and Bob and Problem 2 with Carol." You may not use external resources apart from the textbooks.

Generative AI Policy. The use of generative AI for anything related to assignments is prohibited in this class. **It is always prohibited to input any substantial portion of any assignment into any AI resource.** It is also always forbidden to incorporate any substantive portion of an AI resource's output into your answers, even if you paraphrase it or rewrite it in your own words.

Problem 1 (RSA Example) 10 pts

Let $p = 3, q = 5$, and $N = pq = 15$. Your answers can be concise. You do *not* need to write more than what is being asked in the question (you can, but you don't have to).

- What is $\phi(N)$?
- Let $e = 3$. What is $d = e^{-1} \bmod \phi(N)$?
- For every value $x \in \mathbb{Z}_N^*$, compute $x^3 \bmod N$. Is the function $f(x) = x^3 \bmod N$ a permutation over $\mathbb{Z}_N^*$?
- For every value $x \in \mathbb{Z}_N^*$, compute $x^2 \bmod N$. Is the function $f(x) = x^2 \bmod N$ a permutation over $\mathbb{Z}_N^*$?

Problem 2 (Encrypting Bit-by-Bit) 20 pts

Assume $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is some public-key encryption scheme where the encryption algorithm encrypts a single bit message. Define the following scheme $(\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$ for encrypting arbitrary-length messages where $\mathsf{Enc}'_{pk}(m_1, \cdots, m_\ell) = \mathsf{Enc}_{pk}(m_1), \cdots, \mathsf{Enc}_{pk}(m_\ell)$ encrypts each bit m_i of the message separately and $\mathsf{Dec}'_{sk}(c_1, \ldots, c_\ell) = \mathsf{Dec}_{sk}(c_1), \cdots, \mathsf{Dec}_{sk}(c_\ell)$.

- If $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is CPA secure, is $(\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$ also guaranteed to be CPA secure? Either prove that this holds or give a counter-example. For a proof, you do *not* need to give a formal reduction. A clear high-level argument will suffice for arguing security.
- If $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is CCA secure, is $(\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$ also guaranteed to be CCA secure? Either prove that this holds or give a counter-example. For a proof, you do *not* need to give a formal reduction. A clear high-level argument will suffice for arguing security.

Problem 3 (Signature) — 10 pts

Assume $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ is a secure signature scheme. Consider a modified scheme that signs the first and second half of the message bits separately. That is, we define $(\mathsf{Gen}, \mathsf{Sign}', \mathsf{Verify}')$ where

- $\mathsf{Sign}'_{sk}(m)$: Let $m = m_1||m_2$, where m_1 denotes the first half of the bits and m_2 denotes the second half. Compute $\sigma_1 = \mathsf{Sign}_{sk}(m_1), \sigma_2 = \mathsf{Sign}_{sk}(m_2)$, and output $\sigma = \sigma_1||\sigma_2$
- $\mathsf{Verify}'_{pk}(m = m_1||m_2, \sigma = \sigma_1||\sigma_2)$. Accept if $\mathsf{Verify}_{pk}(m_1, \sigma_1)$ and $\mathsf{Verify}_{pk}(m_2, \sigma_2)$ both hold.

Is the modified scheme also guaranteed to be secure? Prove your answer or show otherwise.

Problem 4 ($\mathbb{Z}_N$ vs $\mathbb{Z}_N^*$) — Extra Credit 10 pts

Assume that the factoring assumption holds for some $\mathsf{GenModulus}$ that samples two random n-bit primes p, q and computes $N = pq$. Show that this implies that it is hard to find any $x \in \mathbb{Z}_N$ such that $x \notin \mathbb{Z}_N^*$. That is, the probability More formally, for any PPT algorithm $\mathcal{A}$ we have:

$$\Pr_{N \leftarrow \mathsf{GenModulus}(1^n)}[\mathcal{A}(N) \in \{1, \ldots, N-1\} \setminus \mathbb{Z}_N^*] = \mathsf{negl}(n)$$

where $\mathcal{A}(N)$ denotes $\mathcal{A}$'s output (i.e. guess x) when given N.