Full name: _____

Student ID: _____

- This exam is made of 11 pages and **10 questions**.
- You work individually on your answers. With the submission of your answers, you declare that this is your **individual work** and you did not ask any other person for assistance. You neither discussed with fellow students nor asked in an internet forum or chat for help.
- Explain and **motivate** all your answers.
- You have **180 minutes** to do the exam (+45 if you are entitled for extra time).
- Make sure your **handwriting is clear** and legible. Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.
- This is a **closed-book** exam. As such, you are *not* allowed to use any book, notes, cheatsheet, phone or laptop.
- You can use a **non-programmable calculator** to double-check your calculations.

Exercise 1 – Perfect Secrecy [10 points]

Consider the cipher where $\mathcal{M} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_4 = \{0, 1, 2, 3\}$, defined by the following encryption map:

$$e_k(m) = (3m + k) \mod 4$$

for all $m \in \mathcal{M}$ and $k \in \mathcal{K}$.

(a) [**2 points**] Draw the encryption table of this cipher.

(a) [**4 points**] Assuming plaintext and key distributions to be uniform, does this cipher satisfy perfect secrecy? Motivate your answer.

(b) [**4 points**] Suppose that we modify the encryption map as follows:

$$e'_k(m) = (2m + k) \mod 4$$

Is this modified cipher perfectly secret? Motivate your answer.

Exercise 2 – PRG and PRF [10 points]

(a) **[3 points]** Let G be a PRG with expansion factor ℓ such that $\ell(n) > n$ for all $n \in \mathbb{N}$, and consider this definition for G':

$$G'(s) = G(s)||G(s)_R \ ,$$

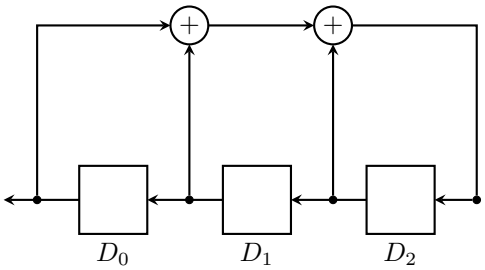
where $G(s)_R$ denotes the string $G(s)$ in reverse order. Is G' a PRG? Motivate your answer. It is not necessary to provide a full proof by reduction or a worked out counterexample, but you should at least provide the intuition behind your answer (for example: if you think that G' is not a PRG, then you should describe a strategy for a distinguisher that gives a non-negligible advantage).

(b) **[3 points]** An *involution* is a function $I : \{0,1\}^n \to \{0,1\}^n$ such that $I(I(x)) = x$ for all $x \in \{0,1\}^n$. Let $\mathbb{F} = \{f_s\}$ be a length-preserving family of functions $f_s : \{0,1\}^{|s|} \to \{0,1\}^{|s|}$ such that f_s is an involution for all $s \in \{0,1\}^*$. Show that f_s cannot be a PRP, by describing a successful strategy for a distinguisher.

(c) **[4 points]** Suppose now that we want to determine if the family $\mathbb{F} = \{f_s\}$ of point (b) above is indistinguishable from a *random involution*, rather than from a generic random permutation as usually done with a PRP. Give a definition of *Pseudorandom Involution* (PRI) for this specific case. In particular, show how the interaction between the distinguisher and the oracle should be modified compared to the usual definition of PRP/PRF.

Exercise 3 – LFSR, Boolean Functions and Stream Ciphers

[**10 points**] Consider the following LFSR:



(a) [**3 points**] Is this a maximum-length LFSR? Draw the transition graph where the nodes represent the states of the registers.

(b) [**3 points**] Let $f : \mathbb{F}_2^3 \to \mathbb{F}_2$ be the 3-variable Boolean function defined as follows: for each $x = (x_1, x_2, x_3) \in \mathbb{F}_2^3$, $f(x) = x_1 \oplus 1$ if $x_2 = x_3 = 0$; otherwise, $f(x) = x_1$. Write the truth table of f. Is f balanced?

(c) [**2 points**] Compute the Walsh coefficient $W_f(1, 0, 0)$ of the function f defined in (b). Based only on the value of this coefficient, what is the nonlinearity of f?

(d) [**2 points**] Compute the first 3 bits of the *filter* generator defined by the LFSR of point (a), filtered with the Boolean function f defined in point (b), starting from the state $(0, 1, 1)$.

Exercise 4 – Block Cipher Modes of Operation

[10 points]

(a) **[4 points]** Describe the ECB and CBC modes of operation for block ciphers, and provide an overview of their pros and cons.

(b) **[3 points]** Briefly describe the Feistel paradigm for designing a block cipher.

(c) **[3 points]** DES is the most famous example of a block cipher designed with the Feistel paradigm. Explain why we should not use it anymore today. Could the use of CBC mode improve the security properties of DES?

Exercise 5 – Message Authentication Codes and Authenticated Encryption

[10 points]

(a) **[5 points]** Alice and Bob use an unconditionally secure MAC where $\mathcal{M} = \{0, 1, \cdots, 12\}$, $\mathcal{A} = \mathbb{Z}_{17}$ and $\mathcal{K} = \mathbb{Z}_{17} \times \mathbb{Z}_{17}$. The tag generation algorithm is defined as usual for all keys $k = (i, j) \in \mathcal{K}$ and messages $m \in \mathcal{M}$ as:

$$\mathsf{Mac}_k(m) = (i + mj) \mod 17 \ .$$

Oscar discovered that Alice and Bob determine the next key to be used in their MAC using the following method: given the key $k = (i, j)$ used to authenticate a message m, the next message m' is authenticated with the key $k' = (i', j') = (i + 1, j + 1)$. Suppose now that Oscar observes the two consecutive message/tag pairs $(8, 3)$ and $(10, 7)$. What are the keys k and k'? Explain in detail how Oscar can retrieve them, assuming that the tag generation algorithm is public.

(b) **[3 points]** Turn the following informal description of a MAC scheme into a formal description, then prove its insecurity: to authenticate a message, we cut it in two halves, XOR together the two halves and then apply a pseudorandom function to the result of the XOR.

(c) **[2 points]** What does an authenticated encryption scheme add beyond CCA-security?

Exercise 6 – Hash Functions [10 points]

(a) **[3 points]** The hash function MD5 (which has been deprecated since at least 2008) has a digest length of 128 bits. How many input messages do we approximately need to hash before finding a collision with a probability of at least 50%? Would it be good enough for today's standards, in terms of security from birthday attacks?

(b) **[4 points]** Let $H : \{0,1\}^* \to \{0,1\}^n$ be a collision-resistant hash function. We define the new hash function H' as follows: given a message m, we parse it in two halves, i.e. $m = m_0||m_1$, then we compute $H'(m) = H(m_0) \oplus H(m_1)$. Is H' also collision-resistant? If not, show how an attacker can produce a collision.

(c) **[3 points]** Briefly describe the random oracle model, with its pros and cons. Name one concrete example where the random oracle model is necessary to prove the security of a construction.

Exercise 7 – Primality Testing and Group Generators [10 points]

(a) **[4 points]** Find a generator for the multiplicative group $\mathbb{Z}_{13}^*$. Motivate your answer by showing how to check the order of the generator.

(b) **[6 points]** Consider the number $p = 67$. How many witness for the primality of p do we need to know that p is composite with a probability less than 5%? Further, perform the Fermat's primality test on p for one witness a of your choice, showing the calculation in detail with the square-and-multiply algorithm.

Exercise 8 – RSA [10 points]

(a) **[3 points]** Given an RSA setup with public modulus n, public exponent e, and private exponent d, describe how to sign a message $m \in \mathbb{Z}_n$ and how to later verify the produced signature. In particular, describe what elements are needed by the signer and by the verifier in order to sign a message and verify a signature, respectively.

(b) **[4 points]** Given primes $p = 5$ and $q = 11$, public modulus $n = pq$, and public exponent $e = 3$, sign the message $m = 5$. Show all intermediate calculation steps.

(c) **[3 points]** In RSA, $e = 3$ is a common choice for the public exponent. Another common value is $e = 65537$. Explain why these are commonly used values (Hint: think about how we perform the encryption/verification operation from an algorithmic point of view).

Exercise 9 – Diffie-Hellman Key Agreement [10 points]

We consider the following variant of the Diffie-Hellman Key Agreement protocol. Given a finite cyclic group G (where the DLOG is assumed to be hard) and a generator g, Alice picks a random exponent a uniformly at random, computes $A = g^a$ and sends it to Bob. Bob picks a random exponent b uniformly at random, computes $B = g^b$ and sends it to Alice. Now, Alice computes the final key as $k_A = B \cdot g^a$, while Bob computes the final key as $k_B = A \cdot g^b$.

(a) **[4 points]** Is the protocol correct? In other words, are Alice and Bob generating the same key?

(b) **[6 points]** Discuss the security of this Diffie-Hellman variant.

Exercise 10 – Schnorr Identification Scheme and Signatures [10 points]

(a) [**6 points**] Given $p = 17$ and the generator $g = 3$ for $\mathbb{Z}_{17}^*$, Alice wants to convince Bob that she knows the DLOG of $y = 5$. Alice thus sends to Bob the value $t = 13$ as a commitment, while Bob sends back to Alice the challenge $c = 2$. Finally, Alice replies to Bob with $s = 14$. Does Bob believe Alice?

(b) [**4 points**] Briefly explain how the Schnorr identification scheme used above can be turned into a (non-interactive) signature scheme. In particular, what additional assumption do we need beside the conjectured hardness of the discrete logarithm problem to prove the security of the signature scheme?