ZKDL Camp Exercises

Distributed Lab

October 3, 2024

Contents

1 Group Theory and Polynomials

Exercise 1. Which of the following statements is **false**?

1. $(\forall a, b \in \mathbb{Q}, a \neq b)\ (\exists q \in \mathbb{R}) : \{a < q < b\}$.
2. $(\forall \varepsilon > 0)\ (\exists n_\varepsilon \in \mathbb{N})\ (\forall n \geq n_\varepsilon) : \{1/n < \varepsilon\}$.
3. $(\forall k \in \mathbb{Z})\ (\exists n \in \mathbb{N}) : \{n < k\}$.
4. $(\forall x \in \mathbb{Z} \setminus \{-1\})\ (\exists! y \in \mathbb{Q}) : \{(x+1)y = 2\}$.

Exercise 2. Denote $X := \{(x, y) \in \mathbb{Q}^2 : xy = 1\}$. Oleksandr claims the following:

1. $X \cap \mathbb{N}^2 = \{(1, 1)\}$.
2. $|X \cap \mathbb{Z}^2| = 2|X \cap \mathbb{N}^2|$.
3. X is a group under the operation $(x_1, y_1) \oplus (x_2, y_2) = (x_1 x_2, y_1 y_2)$.

Which statements are **true**?

a) Only 1.
b) Only 1 and 2.
c) Only 1 and 3.
d) Only 2 and 3.
e) All statements are correct.

Exercise 3. Does a tuple $(\mathbb{Z}, \oplus)$ with operation $a \oplus b = a + b - 1$ define a group?

a) Yes, and this group is abelian.
b) Yes, but this group is not abelian.
c) No, since the associativity property does not hold.
d) No, since there is no identity element in this group.
e) No, since there is no inverse element in this group.

Exercise 4. Consider the Cartesian plane $\mathbb{R}^2$, where two coordinates are real numbers. For two points A, B define the operation $\oplus$ as follows: $A \oplus B$ is the midpoint on segment AB. Does $(\mathbb{R}^2, \oplus)$ define a group?

a) Yes, and this group is abelian.

b) Yes, but this group is not abelian.

c) No, since the associativity property does not hold and there is no identity element in this group.

d) No, since the associativity property does not hold, but we might define an identity element nonetheless.

Exercise 5. Find the inverse of 4 in $\mathbb{F}_{11}$.

a) 8

b) 5

c) 3

d) 7

Exercise 6. Suppose for three polynomials $p, q, r \in \mathbb{F}[x]$ we have $\deg p = 3, \deg q = 4, \deg r = 5$. Which of the following is true for $n := \deg\{(p - q)r\}$?

a) $n = 9$.

b) n might be less than 9.

c) $n = 20$.

d) n is less than $\deg\{qr\}$.

Exercise 7. Define the polynomial over $\mathbb{F}_5$: $f(x) := 4x^2 + 7$. Which of the following is the root of $f(x)$?

a) 2

b) 3

c) 4

d) This polynomial has no roots over $\mathbb{F}_5$.

Exercise 8. Quadratic polynomial $p(x) = ax^2 + bx + c \in \mathbb{R}[x]$ has zeros at 1 and 2 and $p(0) = 2$. Find the value of $a + b + c$.

a) 0

b) -1

c) 1

d) Not enough information to determine.

Exercise 9. Which of the following is a **valid** endomorphism $f : X \to X$?

a) $X = [0, 1]$, $f : x \mapsto x^2$.

b) $X = [0, 1]$, $f : x \mapsto x + 1$.

c) $X = \mathbb{R}_{>0}$, $f : x \mapsto (x - 1)^3$.

d) $X = \mathbb{Q}_{>0}$, $f : x \mapsto \sqrt{x}$.

Exercise 10*. Denote by $\mathrm{GL}(2,\mathbb{R})$ a set of 2×2 invertable matrices with real entries. Define two functions $\varphi : \mathrm{GL}(2,\mathbb{R}) \to \mathbb{R}$:

$$\varphi_1\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad - bc,\ \varphi_2\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = a + d \tag{1}$$

Den claims the following:

1. φ_1 is a group homomorphism between multiplicative groups $(\mathrm{GL}(2,\mathbb{R}), \times)$ and $(\mathbb{R}, \times)$.
2. φ_2 is a group homomorphism between additive groups $(\mathrm{GL}(2,\mathbb{R}), +)$ and $(\mathbb{R}, +)$.

Which of the following is **true**?

a) Only statement 1 is correct.
b) Only statement 2 is correct.
c) Both statements 1 and 2 are correct.
d) None of the statements is correct.

2 Basics of Security Analysis

Exercise 1. Suppose that for the given cipher with a security parameter λ, the adversary $\mathcal{A}$ can deduce the least significant bit of the plaintext from the ciphertext. Recall that the advantage of a bit-guessing game is defined as $\mathsf{SSAdv}[\mathcal{A}] = \left|\Pr\left[b = \hat{b}\right] - \frac{1}{2}\right|$, where b is the randomly chosen bit of a challenger, while $\hat{b}$ is the adversary's guess. What is the maximal advantage of $\mathcal{A}$ in this case?

Hint: The adversary can choose which messages to send to challenger to further distinguish the plaintexts.

a) 1
b) $\frac{1}{2}$
c) $\frac{1}{4}$
d) 0
e) Negligible value ($\mathrm{negl}(\lambda)$).

Exercise 2. Consider the cipher $\mathcal{E} = (E, D)$ with encryption function $E : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ over the message space $\mathcal{M}$, ciphertext space $\mathcal{C}$, and key space $\mathcal{K}$. We want to define the security that, based on the cipher, the adversary $\mathcal{A}$ cannot restore the message (*security against message recovery*). For that reason, we define the following game:

1. Challenger chooses random $m \xleftarrow{R} \mathcal{M}, k \xleftarrow{R} \mathcal{K}$.
2. Challenger computes the ciphertext $c \leftarrow E(k, m)$ and sends to $\mathcal{A}$.
3. Adversary outputs $\hat{m}$, and wins if $\hat{m} = m$.

We say that the cipher $\mathcal{E}$ is secure against message recovery if the **message recovery advantage**, denoted as $\mathrm{MRadv}[\mathcal{A}, \mathcal{E}]$ is negligible. Which of the following statements is a valid interpretation of the message recovery advantage?

a) $\mathrm{MRadv}[\mathcal{A}, \mathcal{E}] := \left|\Pr[m = \hat{m}] - \frac{1}{2}\right|$
b) $\mathrm{MRadv}[\mathcal{A}, \mathcal{E}] := |\Pr[m = \hat{m}] - 1|$.
c) $\mathrm{MRadv}[\mathcal{A}, \mathcal{E}] := \Pr[m = \hat{m}]$
d) $\mathrm{MRadv}[\mathcal{A}, \mathcal{E}] := \left|\Pr[m = \hat{m}] - \frac{1}{|\mathcal{M}|}\right|$

Exercise 3. Suppose that f and g are negligible functions. Which of the following functions is not neccessarily negligible?

a) $f + g$
b) $f \times g$
c) $f - g$
d) f/g
e) $h(\lambda) := \begin{cases} 1/f(\lambda) & \text{if } 0 < \lambda < 100000 \\ g(\lambda) & \text{if } \lambda \geq 100000 \end{cases}$

Exercise 4. Suppose that $f \in \mathbb{F}_p[x]$ is a d-degree polynomial with d **distinct** roots in $\mathbb{F}_p$. What is the probability that, when evaluating f at n random points, the polynomial will be zero at all of them?

a) Exactly $(d/p)^n$.
b) Strictly less that $(d/p)^n$.

c) Exactly nd/p.

d) Exactly d/np.

Exercise 5-6. To demonstrate the idea of Reed-Solomon codes, consider the toy construction. Suppose that our message is a tuple of two elements $a, b \in \mathbb{F}_{13}$. Consider function $f : \mathbb{F}_{13} \to \mathbb{F}_{13}$, defined as $f(x) = ax + b$, and define the encoding of the message (a, b) as $(a, b) \mapsto (f(0), f(1), f(2), f(3))$.

Question 5. Suppose that you received the encoded message $(3, 5, 6, 9)$. Which number from the encoded message is corrupted?

a) First element (3).

b) Second element (5).

c) Third element (6).

d) Fourth element (9).

e) The message is not corrupted.

Question 6. Consider the previous question. Suppose that the original message was (a, b). Find the value of $a \times b$ (in $\mathbb{F}_{13}$).

a) 4

b) 6

c) 12

d) 2

e) 1

3 Field Extensions and Elliptic Curves

Warmup (Oleksandr in search of perfect field extension)

Exercise 1. Oleksandr decided to build $\mathbb{F}_{49}$ as $\mathbb{F}_7[i]/(i^2+1)$. Compute $(3+i)(4+i)$.

a) $6+i$.

b) 6.

c) $4+i$.

d) 4.

e) $2+4i$.

Exercise 2. Oleksandr came up with yet another extension $\mathbb{F}_{p^2} = \mathbb{F}_p[i]/(i^2+2)$. He asked interns to calculate $2/i$. Based on five answers given below, help Oleksandr to find the correct one.

a) 1.

b) $p-2$.

c) $(p-3)i$.

d) $(p-1)i$.

e) $p-1$.

Exercise 3*. After endless tries, Oleksandr has finally found the perfect field extension: $\mathbb{F}_{p^2} := \mathbb{F}_p[v]/(v^2+v+1)$. However, Oleksandr became very frustrated since not for any p this would be a valid field extension. For which of the following values p such construction would **not** be a valid field extension? Use the fact that equation $\omega^3 = 1$ over $\mathbb{F}_p$ has non-trivial solutions (meaning, two others except for $\omega = 1$) if $p \equiv 1 \pmod 3$. You can assume that listed numbers are primes.

a) 8431.

b) 9173.

c) 9419.

d) 6947.

Exercises 4-9. Tower of Extensions

You are given the passage explaining the topic of tower of extensions. The text has gaps that you need to fill in with the correct statement among the provided choices.

This question demonstrates the concept of the so-called **tower of extensions**. Suppose we want to build an extension field $\mathbb{F}_{p^4}$. Of course, we can find some irreducible polynomial $p(X)$ of degree 4 over $\mathbb{F}_p$ and build $\mathbb{F}_{p^4}$ as $\mathbb{F}_p[X]/(p(X))$. However, this method is very inconvenient since implementing the full 4-degree polynomial arithmetic is inconvenient. Moreover, if we were to implement arithmetic over, say, $\mathbb{F}_{p^{24}}$, that would make the matters worse. For this reason, we will build $\mathbb{F}_{p^4}$ as $\mathbb{F}_{p^2}[j]/(q(j))$ where $q(j)$ is an irreducible polynomial of degree 2 over $\mathbb{F}_{p^2}$, which itself is represented as $\mathbb{F}_p[i]/(r(i))$ for some suitable irreducible quadratic polynomial $r(i)$. This way, we can first implement $\mathbb{F}_{p^2}$, then $\mathbb{F}_{p^4}$, relying on the implementation of $\mathbb{F}_{p^2}$ and so on.

For illustration purposes, let us pick $p := 5$. As noted above, we want to build $\mathbb{F}_{5^2}$ first. A valid way to represent $\mathbb{F}_{5^2}$ would be to set $\mathbb{F}_{5^2} :=$ **4**. Given this representation, the zero of a linear polynomial $f(x) = ix - (i+3)$, defined over $\mathbb{F}_{5^2}$, is **5**.

Now, assume that we represent $\mathbb{F}_{5^4}$ as $\mathbb{F}_{5^2}[j]/(j^2-\xi)$ for $\xi = i+1$. Given such representation, the value of j^4 is **6**. Finally, given $c_0 + c_1 j \in \mathbb{F}_{5^4}$ we call $c_0 \in \mathbb{F}_{5^2}$ a **real part**, while $c_1 \in \mathbb{F}_{5^2}$ an **imaginary part**. For example, the imaginary part of number $j^3 + 2i^2\xi$ is **7**, while the real part of $(a_0 + a_1 j)b_1 j$ is **8**. Similarly to complex numbers, it motivates us to define the number's **conjugate**: for $z = c_0 + c_1 j$, define the conjugate as $\overline{z} := c_0 - c_1 j$. The expression $z\overline{z}$ is then **9**.

Exercise 4.

a) $\mathbb{F}_5[i]/(i^2+1)$
b) $\mathbb{F}_5[i]/(i^2+2)$
c) $\mathbb{F}_5[i]/(i^2+4)$
d) $\mathbb{F}_5[i]/(i^2+2i+1)$
e) $\mathbb{F}_5[i]/(i^2+4i+4)$

Exercise 5.

a) $1+i$
b) $1+2i$
c) $1+4i$
d) $2+3i$
e) $3+i$

Exercise 6.

a) $4+2i$
b) $4i$
c) 1
d) $1+2i$
e) $2+4i$

Exercise 7.

a) equal to zero.
b) equal to one.
c) equal to the real part.
d) $2(1+i)$
e) -4

Exercise 8.

a) $a_1 b_1$
b) $a_1 b_1 \xi$
c) $a_0 b_1$
d) $a_0 b_1 \xi$
e) $a_0 a_1$

Exercise 9.

a) $c_0^2 + c_1^2$
b) $c_0^2 - c_1^2 \xi$
c) $c_0^2 + c_1^2 \xi^2$
d) $(c_0^2 + c_1^2 \xi)j$
e) $(c_0^2 - c_1^2)j$

Elliptic Curves

Exercise 10. Suppose that elliptic curve is defined as $E/\mathbb{F}_7 : y^2 = x^3 + b$. Suppose $(2, 3)$ lies on the curve. What is the value of b?

Exercise 11. Sum of which of the following pairs of points on the elliptic curve $E/\mathbb{F}_{11}$ is equal to the point at infinity $\mathcal{O}$ for any valid curve equation?

a) $P = (2, 3), Q = (2, 8)$.
b) $P = (9, 2), Q = (2, 8)$.
c) $P = (9, 9), Q = (5, 7)$.
d) $P = \mathcal{O}, Q = (2, 3)$.
e) $P = [10]G, Q = G$ where G is a generator.

Exercise 12. Consider an elliptic curve E over $\mathbb{F}_{167^2}$. Denote by r the order of the group of points on E (that is, $r = |E|$). Which of the following **can** be the value of r?

a) $167^2 - 5$
b) $167^2 - 1000$
c) $167^2 + 5 \cdot 167$
d) 170^2
e) 160^2

Exercise 13. Suppose that for some elliptic curve E the order is $|E| = qr$ where both q and r are prime numbers. Among listed, what is the most optimal complexity of algorithm to solve the discrete logarithm problem on E?

a) $O(qr)$
b) $O(\sqrt{qr})$
c) $O(\sqrt{\max\{q, r\}})$
d) $O(\sqrt{\min\{q, r\}})$
e) $O(\max\{q, r\})$

4 Projective Coordinates and Pairing

Exercise 1. What is **not** a valid equivalence relation $\sim$ over a set $\mathcal{X}$?

(A) $a \sim b$ iff $a + b < 0$, $\mathcal{X} = \mathbb{Q}$.

(B) $a \sim b$ iff $a = b$, $\mathcal{X} = \mathbb{R}$.

(C) $a \sim b$ iff $a \equiv b \pmod 5$, $\mathcal{X} = \mathbb{Z}$.

(D) $a \sim b$ iff the length of $a =$ the length of b, $\mathcal{X} = \mathbb{R}^2$.

(E) $(a_1, a_2, a_3) \sim (b_1, b_2, b_3)$ iff $a_3 = b_3$, $\mathcal{X} = \mathbb{R}^3$.

Exercise 2. Suppose that over $\mathbb{R}$ we define the following equivalence relation: $a \sim b$ iff $a - b \in \mathbb{Z}$ $(a, b \in \mathbb{R})$. What is the equivalence class of 1.4 (that is, $[1.4]_\sim$)?

(A) A set of all real numbers.

(B) A set of all integers.

(C) A set of reals $x \in \mathbb{R}$ with the fractional part of x equal to 0.4.

(D) A set of reals $x \in \mathbb{R}$ with the integer part of x equal to 1.

(E) A set of reals $x \in \mathbb{R}$ with the fractional part of x equal to 0.6.

Exercise 3. Which of the following pairs of points in homogeneous projective space $\mathbb{P}^2(\mathbb{R})$ are **not** equivalent?

(A) $(1 : 2 : 3)$ and $(2 : 4 : 6)$.

(B) $(2 : 3 : 1)$ and $(6 : 9 : 3)$.

(C) $(5 : 5 : 5)$ and $(2 : 2 : 2)$.

(D) $(4 : 3 : 2)$ and $(16 : 8 : 4)$.

Exercise 4. The main reason for using projective coordinates in elliptic curve cryptography is:

(A) To reduce the number of point additions in algorithms involving elliptic curves.

(B) To make the curve more secure against attacks.

(C) To make the curve more efficient in terms of memory usage.

(D) To reduce the number of field multiplications when performing scalar multiplication.

(E) To avoid making too many field inversions in complicated algorithms involving elliptic curves.

Exercise 5. Suppose $k = 19$ is a scalar and we are calculating $[k]P$ using the double-and-add algorithm. How many elliptic curve point addition operations will be performed?

(A) 0.

(B) 1.

(C) 2.

(D) 3.

(E) 4.

Exercise 6. What is the minimal number of inversions needed to calculate the value of expression (over $\mathbb{F}_p$)

$$\frac{a-b}{(a+b)^4} + \frac{c}{a+b} + \frac{d}{a^2+c^2},$$

for the given scalars $a, b, c, d \in \mathbb{F}_p$?

(A) 1.
(B) 2.
(C) 3.
(D) 4.
(E) 5.

Exercise 7. Given pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ with G_1 — generator of $\mathbb{G}_1$ and $G_2 \in \mathbb{G}_2$ — generator of $\mathbb{G}_2$, which of the following is **not** equal to $e([3]G_1, [5]G_2)$?

(A) $e([5]G_1, [3]G_2)$.
(B) $e([4]G_1, [4]G_2)$.
(C) $e([15]G_1, G_2)$.
(D) $e([3]G_1, G_2)e(G_1, [12]G_2)$.
(E) $e(G_1, G_2)^{15}$.

Exercise 8*. *Unit Circle Proof.* Suppose Alice wants to convince Bob that she knows a point on the unit circle $x^2 + y^2 = 1$. Suppose we are given a symmetric pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ for $\mathbb{G}_1 = \mathbb{G}_2 = \langle G \rangle$ and Alice computes $P \leftarrow [x]G, Q \leftarrow [y]G$. She then proceeds to sending (P, Q) to Bob. Which of the following checks should Bob perform to verify that Alice indeed knows a point on the unit circle?

(A) Check if $e(P, Q)e(Q, P) = 1$.
(B) Check if $e([2]P, [2]Q) = e(G, G)$.
(C) Check if $e([2]P, Q)e(Q, [2]P) = 1$.
(D) Check if $e(P, P) + e(Q, Q) = 1$.
(E) Check if $e(P, P)e(Q, Q) = e(G, G)$.

5 Commitment Schemes

Exercise 1. Dmytro and Denis were watching a horse race. Confident in his ability to predict the outcome, Dmytro decided to commit to his prediction. However, in his haste, he forgot to use a blinding factor. Now, Dmytro is concerned that Denis might discover his prediction before the race ends, which would defeat the purpose of his commitment.

We define a dummy hash function $H(a) = (a \cdot 13 + 17) \pmod{41}$. Dmytro used a *hash-based commitment* and H as a hash function. Set of race horse numbers is $(3, 5, 8, 15)$. Help Denis to find out the horse number Dmytro have made a commitment to, if commitment equals $C = 39$.

(A) 3.
(B) 5.
(C) 8.
(D) 15.

Exercise 2. Denis made a setup (points G and U) for a Pedersen commitment scheme and commited values $(m, r) = (3, 7)$ to Dmytro by sending him $\mathcal{C} = [3]G + [7]U$. Dmytro did not verify the setup. Turns out that Denis knows that $U = [6]G$. Denis is planning to send a different message from the one he originally committed to to $m_2 = 15$. Which values (m_2, r_2) should he send to Dmytro at the opening stage?

(A) $(15, 5)$
(B) $(15, 7)$
(C) $(15, 4)$
(D) $(3, 5)$

Exercise 3. We define a dummy hash function $H(a, b) = (a \cdot 3 + b \cdot 7) \pmod{41}$. You have a Merkle tree built with depth 4 using hash function H with root equal 37. Position defines how leaves should be hashed:

- if left, then $h_i \leftarrow H(h_{i-1}, \text{branch}[i])$
- if right, then $h_i \leftarrow H(\text{branch}[i], h_{i-1})$

Which inclusion proof is valid for element 3?

(A) branch: $[4, 16, 13]$, position: [left, right, left]
(B) branch: $[1, 40, 3]$, position: [left, left, left]
(C) branch: $[5, 12, 13]$, position: [right, right, left]
(D) branch: $[4, 17, 13]$, position: [left, right, left]

Exercise 4. Given a polynomial $p(x) = x^3 - 10x^2 + 31x - 30$, Oleksandr wants to prove that $p(2) = 0$. To do that, according to the KZG commitment scheme, he constructs the quotient polynomial $q(x)$ and wants to show that $q(\tau) \cdot (\tau - 2) = p(\tau)$. Assuming Oleksandr has conducted these steps correctly, what value of $q(x)$ has Oleksandr calculated?

(A) $q(x) = 2x^2 + 4x - 6$
(B) $q(x) = x^3 - 10x^2 + 30x - 28$
(C) $q(x) = x^2 - 8x + 15$
(D) $q(x) = x^2 + 5x + 18$

6 Introduction to Zero-Knowledge Proofs

Exercise 1. When dealing with RSA protocol, one frequently encounters the following relation where e is a prime number and $n \in \mathbb{N}$:

$$\mathcal{R} = \left\{(w, x) \in \mathbb{Z}_n^\times \times \mathbb{Z}_n^\times : w^e = x\right\}$$

Which of the following is the language $\mathcal{L}_\mathcal{R}$ that corresponds to the relation $\mathcal{R}$?

(A) Integers from $\mathbb{Z}_n^\times$ which have a modular root of e-th degree.
(B) Integers from $\mathbb{Z}_n^\times$ which are divisible by e.
(C) Integers x from $\mathbb{Z}_n^\times$ with properly defined expression x^e.
(D) Integers from $\mathbb{Z}_n^\times$ which are prime.
(E) Integers from $\mathbb{Z}_n^\times$ for which e is a primitive root.

Exercise 2. Suppose that for some interactive protocol $(\mathcal{P}, \mathcal{V})$ during one round, the probability that the verifier $\mathcal{V}$ accepts a false statement is $1/8$. How many rounds of interaction are needed to guarantee 120 bits of security? Assume here that n bits of security means that the probability of accepting a false statement is at most 2^{-n}.

(A) 30.
(B) 40.
(C) 60.
(D) 90.
(E) 120.

Exercise 3. Recall that for relation $\mathcal{R} = \{(w, x) \in \mathbb{Z}_N^\times \times \mathbb{Z}_N^\times : x = w^2\}$ we defined the following interactive protocol $(\mathcal{P}, \mathcal{V})$ to prove that $x \in \mathcal{L}_\mathcal{R}$:

- $\mathcal{P}$ samples $r \xleftarrow{R} \mathbb{Z}_N^\times$ and sends $a = r^2$ to $\mathcal{V}$.
- $\mathcal{V}$ sends a random bit $b \in \{0, 1\}$ to $\mathcal{P}$.
- $\mathcal{P}$ sends $z = r \cdot w^b$ to $\mathcal{V}$.
- $\mathcal{V}$ accepts if $z^2 = a \cdot x^b$, otherwise it rejects.

Suppose we use the protocol $(\mathcal{P}, \mathcal{V}^*)$ where the "broken" verifier $\mathcal{V}^*$ always outputs $b = 1$. Which of the following statements is true?

(A) Both the soundness and completeness of the protocol are preserved.
(B) The soundness of the protocol is preserved, but the completeness is broken.
(C) The completeness of the protocol is preserved, but the soundness is broken.
(D) Both the soundness and completeness of the protocol are broken.

Exercise 4. What is the difference between the cryptographic proof and the proof of knowledge?

(A) Cryptographic proof is a proof of knowledge that is secure against malicious verifiers.
(B) Cryptographic proof is a proof of knowledge that is secure against malicious provers.
(C) Cryptographic proof merely states the correctness of a statement, while the proof of knowledge also guarantees that the prover knows the witness.
(D) While cryptographic proof states that witness exists for the given statement, the proof of knowledge makes sure to make this witness unknown to the verifier.
(E) Proof of knowledge does not require verifier to know the statement, while cryptographic proof does.

Exercise 5. What is the purpose of introducing the extractor?

(A) To introduce the algorithm that simulates the malicious verifier trying to extract the witness from the prover.
(B) To define what it means that the prover knows the witness.
(C) To give the verifier the ability to extract the witness from the prover during the interactive protocol.
(D) To define the security of the interactive protocol that uses a more powerful verifier that can extract additional information from the prover.
(E) To give prover more power to extract randomness generated by the verifier.

Exercise 6. What it means that the interactive protocol $(\mathcal{P}, \mathcal{V})$ is a zero-knowledge?

(A) The verifier $\mathcal{V}$ cannot know whether the given statement is true or false.

(B) The verifier $\mathcal{V}$ cannot know whether the prover $\mathcal{P}$ knows the witness.

(C) View of the prover $\mathcal{P}$ in the protocol is indistinguishable from the view of the verifier $\mathcal{V}$.

(D) Any view of any verifier $\mathcal{V}$ can be simulated using some polynomial-time algorithm, outputting computationally indistinguishable distribution from the given view.

(E) The prover $\mathcal{P}$ can convince the verifier $\mathcal{V}$ that the statement is true without knowing the witness.

Hint: View of the participant in the protocol consists of all data he has access to during the protocol execution. For example, verifier $\mathcal{V}$'s view consists of the messages he sends and receives, as well as the random coins he generates.

Exercise 7. Which of the following is **not** true about the Fiat-Shamir heuristic?

(A) If the public-coin protocol is sound, the Fiat-Shamir transformation preserves the soundness.

(B) The Fiat-Shamir heuristic does not break the completeness of the public-coin protocol it is applied to.

(C) Practically, it allows to convert any interactive protocol into a non-interactive one.

(D) To make Fiat-Shamir transformation pratical, the function modelling the random oracle should be hard to invert.

(E) It is reasonable to use SHA256 to model the random oracle in the Fiat-Shamir transformation.

7 Sigma Protocols

Exercises 1-5. In search of correct Schnorr's Identification Protocol...

You are given the protocol and five ways to implement it. Most of them lack the crucial properties. For each attempt, you need to determine whether the protocol is correct and, if not, specify which of the properties are violated.

Recall, that given the cyclic group $\mathbb{G}$ of order q, the prover wants to convince the verifier that he knows the discrete logarithm α of $h \in \mathbb{G}$ with respect to the generator $g \in \mathbb{G}$ (so that $g^\alpha = h$).
Here are five attempts to construct the protocol:

Attempt 1. Prover sends witness α to the verifier. Verifier checks whether $h = g^\alpha$.

Attempt 2. Prover chooses random $r \xleftarrow{R} \mathbb{Z}_q$ and sends $a \leftarrow \alpha + r$ to the verifier. Verifier checks whether $h = g^a$.

Attempt 3. Prover chooses random $r \xleftarrow{R} \mathbb{Z}_q$, calculates $a \leftarrow \alpha + r$ and sends both (a, r) to the verifier. Verifier checks whether $g^r h = g^a$.

Attempt 4. Prover chooses random $r \xleftarrow{R} \mathbb{Z}_q$, calculates $a \leftarrow g^r, z \leftarrow \alpha + r$ and sends (a, z) to the verifier. Verifier checks whether $a \cdot h = g^z$.

Attempt 5. Prover chooses random $r \xleftarrow{R} \mathbb{Z}_q$, calculates $a \leftarrow g^r$, and sends a to the verifier. Verifier chooses $e \xleftarrow{R} \mathbb{Z}_q$ and sends to the prover. Prover calculates $z \leftarrow \alpha e + r$ and sends to the prover. Verifier checks whether $a \cdot h^e = g^z$.

Below, mark whether the properties of *completeness*, *soundness*, and *zero-knowledge* hold for each attempt.

Attempt #	**1**	**2**	**3**	**4**	**5**
Completeness holds?	✓/✗	✓/✗	✓/✗	✓/✗	✓/✗
Soundness holds?	✓/✗	✓/✗	✓/✗	✓/✗	✓/✗
Zero-Knowledge holds?	✓/✗	✓/✗	✓/✗	✓/✗	✓/✗

Exercises 6-10. Non-Interactive Chaum-Pedersen Protocol.

This section explores how to make the previously considered Chaum-Pedersen protocol non-interactive. Fill in the gaps in the following text with the correct statements.

Recall that the Chaum-Pedersen protocol allows the prover $\mathcal{P}$ to convince the skeptical verifier $\mathcal{V}$ that the given triplet $(u, v, w) \in \mathbb{G}^3$ is a Diffie-Hellman (DH) triplet in the cyclic group $\mathbb{G}$ of prime order q with a generator $g \in \mathbb{G}$, meaning that $u = g^\alpha, v = g^\beta, w = g^{\alpha\beta}$ for some $\alpha, \beta \in \mathbb{Z}_q$. However, instead of making (α, β) as a witness, observe that β is sufficient. Indeed, if $u = g^\alpha, v = g^\beta$, then $w =$ **6**. Thus, the relation is:

$$\mathcal{R} = \left\{((u, v, w), \beta) \in \mathbb{G}^3 \times \mathbb{Z}_q : \boxed{7}\right\}$$

Now, we apply the *Fiat-Shamir Transformation*. Recall that prover, instead of getting the random challenge $c \xleftarrow{R} \mathcal{C} \subset \mathbb{Z}_q$ from the verifier interactively, calculates it as the hash function from the public statement (u, v, w) and the prover's commitment. For that reason, define the non-interactive proof system $\Phi = (\mathsf{Gen}, \mathsf{Verify})$ as follows:

- Gen: On input $(u, v, w) \in \mathbb{G}^3$,
  1. Sample $\beta_r \xleftarrow{R} \mathbb{Z}_q$ and compute the commitment **8**.
  2. Use the hash function **9** to get the challenge $c \leftarrow$ **10**.
  3. Compute response $\beta_z \leftarrow \beta_r + \beta c$ and output commitment (v_r, w_r) and β_z as a proof π.
- Verify: Upon receiving statement (u, v, w) and a proof $\pi = (v_r, w_r, \beta_z)$, the verifier:
  1. Recomputes the challenge c using the hash function.
  2. Accepts if and only if $g^{\beta_z} = v_r v^c$ and $u^{\beta_z} = w_r w^c$.

Exercise 6.

A) v^β
B) u^β
C) vu
D) v^u
E) $v^\beta u$

Exercise 7.

A) $v = g^\beta$ and $w = vu$
B) $v = g^\beta$ and $w = v^\beta$
C) $v = g^\beta$ and $w = u^\beta$
D) $u = g^\beta$ and $w = u^\beta$
E) $u/w = g^\beta$

Exercise 8.

A) $(v_r, w_r) = (g^{\beta_r}, g^{\beta_r\beta})$
B) $(v_r, w_r) = (g^{\beta_r}, w^{\beta_r})$
C) $(v_r, w_r) = (g^{\beta_r}, u^{\beta_r})$
D) $(v_r, w_r) = (g^\beta, g^{\beta_r})$
E) $(v_r, w_r) = (g^\beta, g^{\beta_r} g^\beta)$

Exercise 9.

A) $H : \mathbb{G}^3 \times \mathbb{G}^2 \to \mathcal{C}$
B) $H : \mathbb{G}^3 \times (\mathbb{G} \times \mathbb{Z}_q) \to \mathcal{C}$
C) $H : \mathbb{G}^3 \to \mathcal{C}$
D) $H : \mathbb{G}^3 \times \mathbb{Z}_q \to \mathcal{C}$
E) $H : \mathbb{G}^2 \times \mathbb{Z}_q \to \mathcal{C}$

Exercise 10.

A) $H((u, v, w), (v_r, w_r))$
B) $H((u, v, w), (v_r, \beta_r))$
C) $H(u, v, w)$
D) $H((u, v, w), \beta_r)$
E) $H((v_r, w_r), \beta_r)$

8 Introduction to SNARKs. Arithmetic Circuits. R1CS

8.1 R1CS In Rust

8.1.1 Introduction

This time, the task is a bit unusual: you need to implement a simple Rank-1 Constraint System (R1CS) in Rust. For that reason, consider a pretty simple problem: the prover $\mathcal{P}$ wants to convince the verifier $\mathcal{V}$ that he knows the modular cube root of y modulo p for the given $y \in \mathbb{F}_p$. Here, p is the BLS12-381 prime, which will become handy in the next tasks.

For that reason, we construct the circuit of the following form:

$$\mathtt{C}(x, y) = x^3 - y,$$

Here, we need only two constraints to check the correctness of the prover's statement:

1. $r_1 = x \times x$.
2. $r_2 = x \times r_1 - y$.

Therefore, the solution vector becomes $\mathbf{w} = (1, x, y, r_1, r_2)$. The goal of this task is to:

- Implement the basic Linear Algebra operations for R1CS in Rust.
- Implement the R1CS satisfiability check.
- Construct the matrices L, R, O to check the satisfiability of the given solution vector $\mathbf{w}$ (checking the cubic root of given y).

8.1.2 Task 1: Preparation

All the source code we are going to refer to is specified by the link below:

https://github.com/ZKDL-Camp/lecture-8-r1cs-qap

Download Rust[1] (in case you do not have one), clone/fork the repository and verify that everything compiles (just that, the code does not work yet). In case you are confused, the project is structured as follows:

- `src/main.rs` contains the entrypoint where you can test your implementation.
- `src/finite_field.rs` contains the $\mathbb{F}_p$ specification — you will not need it.
- `src/linear_algebra.rs` contains the basic Linear Algebra operations (with vectors and matrices) you need to implement.
- `src/r1cs.rs` contains the R1CS implementation where you also would need to implement a piece of functionality.

8.1.3 Task 2: Linear Algebra Operations

Now, recall that our ultimate goal is to construct the matrices L, R, O to check the following satisfiability condition:

$$L\mathbf{w} \odot R\mathbf{w} = O\mathbf{w},$$

[1] If you are the total beginner, you might find these official resources useful: https://www.rust-lang.org/learn

And additionally, for education purposes, we will want to check the satisfiability of any specified constraint, that is:

$$\langle \boldsymbol{\ell}_j, \mathbf{w} \rangle \times \langle \boldsymbol{r}_j, \mathbf{w} \rangle = \langle \boldsymbol{o}_j, \mathbf{w} \rangle.$$

For that reason, we need to have the Hadamard product (element-wise multiplication) and inner (dot) product of two vectors and the matrix-vector product. For that reason, implement the following functions in the `linear_algebra.rs` module:

1. `Vector::dot(&self, other: &Self) -> Fp` — the inner product of two vectors.
2. `Vector::hadamard_product(&self, other: &Self) -> Self` — the Hadamard (elementwise) product $\mathbf{v} \odot \mathbf{u}$ of two vectors.
3. `Matrix::hadamard_product(&self, other: &Self) -> Self` — the Hadamard (elementwise) product $A \odot B$ of two matrices.
4. `Matrix::vector_product(&self, other: &Vector) -> Vector` — the matrix-vector product $A\mathbf{v}$.

To test the correctness of your implementation, run

```
cargo test linear_algebra
```

8.1.4 Task 3: R1CS Satisfiability Check

Now, we need to implement the R1CS satisfiability check. For that reason, implement the following functions in the `r1cs.rs` module:

1. `R1CS::is_satisfied(&self, witness: &Vector<WITNESS_SIZE>) -> bool` — the function that checks the satisfiability of the given solution vector $\mathbf{w}$.
2. `R1CS::is_constraint_satisfied(&self, witness: &Vector<WITNESS_SIZE>, j: usize) -> bool` — the function that checks whether the j-th constraint is satisfied.

To test the correctness of your implementation, run

```
cargo test r1cs
```

8.1.5 Task 4: R1CS for Cubic Root

Now, as the final step, construct the matrices L, R, O for the given R1CS problem and check the satisfiability of the solution vector $\mathbf{w} = (1, x, y, r_1, r_2)$ where x is the cubic root of y modulo p. For that reason, insert the missing pieces of code in the `main.rs` file. This file will automatically:

1. Generate a random valid witness.
2. Construct the R1CS with the given matrices L, R, O.
3. Check the satisfiability of the given solution vector.

Hint. In the lecture, we considered a bit more complicated circuit

$$\mathtt{C}(x_1, x_2, x_3) = x_1 \times x_2 \times x_3 + (1 - x_1) \times (x_2 + x_3), \quad x_1 \in \{0, 1\}, \quad x_2, x_3 \in \mathbb{F}_p$$

You might take a look at how this circuit is implemented in the `r1cs.rs` file in the `tests` module and adapt it to the cubic root problem.