Zero-Knowledge Proofs

Practice Workbook

Exercises, Problems, and Case Studies

Companion to the ZK Proofs Tutorial

How to Use This Workbook

This workbook is designed to complement the Zero-Knowledge Proofs Tutorial with hands-on practice. Each section contains:

- **Warm-up exercises** to build foundational understanding
- **Practice problems** at varying difficulty levels
- **Challenge questions** for deeper exploration
- **Real-world case studies** connecting theory to practice

Difficulty Levels

★ **Beginner**	Fundamental concepts and direct applications
★★ **Intermediate**	Requires synthesis of multiple concepts
★★★ **Advanced**	Complex analysis and novel applications

Section 1: Theory and Concepts

1.1 Properties of Zero-Knowledge Proofs

Exercise 1.1.1 (★ Beginner)
For each scenario below, identify which ZK property would be violated if the condition were not met:

a) A dishonest prover successfully convinces the verifier of a false statement 95% of the time
b) An honest prover with a valid solution is rejected by the verifier
c) The verifier learns the prover's secret password after verification
d) The verifier can determine which of two possible solutions the prover used

Answers: _____

Exercise 1.1.2 (★★ Intermediate)
Design a simple protocol for proving you know a password without revealing it. Your protocol should:

- Use a hash function
- Include a random challenge
- Satisfy all three ZK properties

Describe your protocol step-by-step:

Step 1: _____

Step 2: _____

Step 3: _____

1.2 Graph Three-Coloring

Exercise 1.2.1 (★ Beginner)
Consider a triangle graph (3 vertices, all connected). How many valid 3-colorings exist?

Answer: _____

Exercise 1.2.2 (★★ Intermediate)
A university has 5 classes: Math, Physics, Chemistry, Biology, History. The shared students are:

- Math shares students with Physics and Chemistry
- Physics shares students with Math, Chemistry, and Biology
- Chemistry shares students with all other classes

- Biology shares students with Physics and Chemistry
- History shares students only with Chemistry

a) Draw the graph representing this problem
b) Can you schedule all classes using only 3 time slots? If yes, provide one valid schedule.

Your answer:

Section 2: Mathematical Foundations

2.1 Set Theory and Notation

Exercise 2.1.1 (★ Beginner)
Given sets A = {1, 2, 3, 4, 5} and B = {3, 4, 5, 6, 7}, determine:

e) Is $3 \in A$?
f) Is $8 \in B$?
g) Is $\{3, 4\} \subseteq A$?
h) List all elements in $A \cap B$ (intersection)

Answers: _____

Exercise 2.1.2 (★★ Intermediate)
In cryptography, we often work with sets of valid keys. Define a set K of all 256-bit integers that:

- Are greater than 2^{100}
- Are prime numbers
- Are not divisible by any number less than 1000

Write this using proper set notation (use set-builder notation):

K = { _____ }

2.2 Modular Arithmetic

Exercise 2.2.1 (★ Beginner)
Calculate the following:

i) 47 mod 12 = _____
j) 100 mod 7 = _____
k) (15 + 23) mod 11 = _____
l) (8 × 9) mod 13 = _____

Exercise 2.2.2 (★★ Intermediate)
In a finite field F_{17} (integers modulo 17), compute:

m) (12 + 15) mod 17 = _____
n) (12 × 15) mod 17 = _____
o) The multiplicative inverse of 5 in F_{17} (find x where $5x \equiv 1 \mod 17$) = _____

Exercise 2.2.3 (★★★ Advanced)
Prove that in modular arithmetic: If $a \equiv b \pmod n$ and $c \equiv d \pmod n$, then $(a + c) \equiv (b + d) \pmod n$

Your proof:

2.3 Group Theory

Exercise 2.3.1 (★ Beginner)
Consider the set of integers {0, 1, 2, 3, 4} under addition modulo 5.

p) What is the identity element? _____
q) What is the inverse of 3? _____
r) Verify closure: (3 + 4) mod 5 = _____

Exercise 2.3.2 (★★ Intermediate)
Verify that the set of non-zero rational numbers under multiplication forms a group by checking all four axioms. Provide specific examples for each axiom.

Closure: _____

Associativity: _____

Identity: _____

Inverse: _____

2.4 Polynomials

Exercise 2.4.1 (★ Beginner)
For the polynomial $P(x) = 2x^3 - 5x^2 + 3x - 7$:

s) What is the degree? _____
t) Evaluate P(2) = _____
u) Evaluate P(0) = _____

Exercise 2.4.2 (★★ Intermediate)
Given three points: (1, 5), (2, 9), (3, 15), find the polynomial of least degree that passes through all three points using Lagrange interpolation.

Hint: For three points, you need a degree-2 polynomial: $P(x) = ax^2 + bx + c$

Your polynomial: P(x) = _____

Exercise 2.4.3 (★★★ Advanced)
Explain how the Schwartz-Zippel Lemma enables efficient verification of polynomial identities. Give a concrete example with a degree-3 polynomial and calculate the probability of a false positive when checking at 5 random points from a field of size 100.

Your explanation:

Section 3: Cryptographic Primitives

3.1 Discrete Logarithm Problem

Exercise 3.1.1 (★ Beginner)
Solve for k in the following small examples:

v) $2^k \equiv 8 \pmod{11}$ → k = _____
w) $3^k \equiv 4 \pmod{7}$ → k = _____

Exercise 3.1.2 (★★ Intermediate)
Explain why the discrete logarithm problem becomes computationally infeasible for large primes. What is the best-known algorithm for solving it, and what is its time complexity?

Your answer:

3.2 Diffie-Hellman Key Exchange

Exercise 3.2.1 (★★ Intermediate)
Alice and Bob want to establish a shared secret using Diffie-Hellman. They publicly agree on p = 23 and g = 5.

- Alice secretly chooses a = 6
- Bob secretly chooses b = 15

Calculate:

x) Alice sends $A = g^a \bmod p$ = _____
y) Bob sends $B = g^b \bmod p$ = _____
z) Alice computes shared secret $s = B^a \bmod p$ = _____
aa) Bob computes shared secret $s = A^b \bmod p$ = _____

Verify that Alice and Bob computed the same shared secret!

3.3 Elliptic Curves

Exercise 3.3.1 (★ Beginner)
Compare the security levels:

bb) A 256-bit ECC key is equivalent to approximately _____ bits of RSA security
cc) Why does this matter for blockchain applications?

Your answer:

Exercise 3.3.2 (★★★ Advanced)
Research and explain the difference between Montgomery curves (like Curve25519) and Edwards curves (like Ed25519). Why would you choose one over the other for:

dd) Key exchange protocols
ee) Digital signatures

Your analysis:

Section 4: Synthesis and Application

4.1 Building a Simple ZK Proof

Challenge Problem (★★★ Advanced)
Design a zero-knowledge proof system for proving you know a solution to a Sudoku puzzle without revealing the solution. Your design should include:

1. How to encode the Sudoku grid into a polynomial or commitment scheme
2. What constraints need to be checked (row, column, box constraints)
3. How the verifier can check these constraints without seeing the solution
4. Why your proof satisfies completeness, soundness, and zero-knowledge

Your protocol design:

4.2 Real-World Case Studies

Case Study 1: Privacy-Preserving Transactions

Scenario: A blockchain needs to verify that users have sufficient funds for transactions without revealing their actual balances.

Questions:

ff) What information needs to be proven? What needs to remain hidden?
gg) How would you use range proofs to verify amounts are positive?
hh) What cryptographic commitments would you use?

Your analysis:

Case Study 2: ZK Compression on Solana

Scenario: Solana uses ZK Compression to reduce on-chain storage costs while maintaining security.

Questions:

ii) How do Merkle trees enable ZK Compression?
jj) What data is stored on-chain vs. off-chain?
kk) How does this reduce costs while maintaining verifiability?
ll) What are the trade-offs?

Your analysis:

Case Study 3: Identity Verification

Scenario: A decentralized app needs to verify users are over 21 without revealing their exact birthdate or identity.

Questions:

mm) What statement needs to be proven?
nn) How could you use digital signatures and ZK proofs together?
oo) What role would a trusted authority play?
pp) How do you prevent replay attacks?

Your protocol design:

4.3 Biological Systems Connection

Challenge Question (★★★ Advanced)
The immune system must verify cells are 'self' without constantly checking their entire molecular makeup. This is analogous to zero-knowledge proofs.

Design a conceptual framework that maps:

qq) MHC (Major Histocompatibility Complex) presentation to cryptographic commitments
rr) T-cell receptor verification to the verification phase of ZK proofs
ss) Self vs. non-self recognition to completeness and soundness

How could this biological inspiration inform new ZK proof designs for distributed systems?

Your framework:

Answer Key

Section 1: Theory and Concepts

Exercise 1.1.1

tt) Soundness (dishonest prover should not succeed)
uu) Completeness (honest prover should always convince verifier)
vv) Zero-knowledge (no information should be revealed)
ww) Zero-knowledge (verifier learns which solution was used)

Exercise 1.1.2
Sample Protocol:
Step 1: Prover computes H(password) and sends to verifier
Step 2: Verifier sends random challenge nonce
Step 3: Prover computes H(password || nonce) and sends result
Step 4: Verifier computes H(stored_hash || nonce) and compares

Exercise 1.2.1
Answer: 6 valid 3-colorings (each vertex gets a different color, 3! = 6 permutations)

Exercise 1.2.2
Answer: Yes, can be done with 3 time slots. Sample schedule:

- Slot 1 (Red): Math, Biology
- Slot 2 (Blue): Physics, History
- Slot 3 (Green): Chemistry

Section 2: Mathematical Foundations

Exercise 2.1.1

xx) Yes, $3 \in A$
yy) No, $8 \notin B$
zz) Yes, $\{3, 4\} \subseteq A$
aaa) $A \cap B = \{3, 4, 5\}$

Exercise 2.2.1

bbb) $47 \bmod 12 = 11$
ccc) $100 \bmod 7 = 2$
ddd) $(15 + 23) \bmod 11 = 38 \bmod 11 = 5$
eee) $(8 \times 9) \bmod 13 = 72 \bmod 13 = 7$

Exercise 2.2.2

fff) $(12 + 15) \bmod 17 = 27 \bmod 17 = 10$
ggg) $(12 \times 15) \bmod 17 = 180 \bmod 17 = 10$
hhh) Multiplicative inverse of 5 in F_{17} is 7 (because $5 \times 7 = 35 \equiv 1 \bmod 17$)

Exercise 2.3.1

iii) Identity element: 0
jjj) Inverse of 3: 2 (because $3 + 2 \equiv 0 \mod 5$)
kkk) $(3 + 4) \mod 5 = 7 \mod 5 = 2$ (which is in the set, demonstrating closure)

Exercise 2.4.1

lll) Degree: 3
mmm) $P(2) = 2(8) - 5(4) + 3(2) - 7 = 16 - 20 + 6 - 7 = -5$
nnn) $P(0) = -7$

Section 3: Cryptographic Primitives

Exercise 3.1.1

ooo) $k = 3$ (because $2^3 = 8$)
ppp) $k = 5$ (because $3^5 = 243 \equiv 4 \mod 7$)

Exercise 3.2.1

qqq) $A = 5^6 \mod 23 = 15625 \mod 23 = 8$
rrr) $B = 5^{15} \mod 23 = 19$ (computed via repeated squaring)
sss) Alice: $s = 19^6 \mod 23 = 2$
ttt) Bob: $s = 8^{15} \mod 23 = 2$

Both computed the same shared secret: 2 ✓

Exercise 3.3.1

uuu) 256-bit ECC ≈ 3072-bit RSA
vvv) Matters for blockchains because: smaller keys = less storage, faster verification, reduced transaction size = lower fees and higher throughput

Notes on Other Exercises

The advanced exercises and case studies are open-ended and designed to encourage deep thinking and research. There are multiple valid approaches. Key considerations for evaluation:

- Does the solution address all three ZK properties?
- Is the mathematical reasoning sound?
- Are the cryptographic primitives used correctly?
- Does the design consider practical implementation?

Further Practice

To continue developing your ZK proof skills:

5. **Implement simple ZK protocols** in Python or Rust
6. **Explore ZK libraries:** circom, snarkjs, ZoKrates, gnark
7. **Study real implementations:** Zcash, Tornado Cash, ZK Rollups
8. **Read academic papers:** Pinocchio, Groth16, PLONK, STARK
9. **Participate in ZK communities:** ZKProof Workshop, PSE, 0xPARC

Recommended Projects

Beginner Project: Implement a simple commitment scheme and zero-knowledge proof for password verification

Intermediate Project: Build a range proof system to prove a value is within bounds without revealing it

Advanced Project: Create a ZK circuit for proving valid Sudoku solutions using a ZK framework

Expert Project: Design and implement a custom ZK protocol for a specific use case relevant to your work

Final Thoughts

Zero-knowledge proofs represent one of the most powerful tools in modern cryptography. By completing this workbook, you've taken significant steps toward mastering this complex but crucial technology.

Remember: understanding comes through practice. Don't just read about ZK proofs--implement them, break them, fix them, and build with them.

The future of privacy-preserving computation is being built today, and you now have the foundational knowledge to contribute to it.