# Practice-Only Lecture Notes: Hard Problems in Post-Quantum Cryptography

Meticulous Problem Design with Assessment Intent

## How to Use These Notes (Problems Only, With Design Intent)

Each problem is intentionally designed to verify specific competencies. For every problem, we provide:

- **Purpose:** the conceptual skill the student should gain.
- **Verifies:** what the student must demonstrate (knowledge/technique/proof skill/attack reasoning).
- **Deliverable:** what to submit (calculation, proof sketch, algorithm, complexity estimate, critique).

Problems progress *basic* → *intermediate* → *advanced*, and within each topic:

1. **Formalization** (write the game precisely, track distributions),
2. **Reductions/Equivalences** (search vs decision, embedding arguments),
3. **Attack Reasoning** (identify best-known lines of attack and their preconditions),
4. **Parameter Thinking** (how constraints interact; what breaks if changed).

## Contents

# 1 Lattice-Based: Learning With Errors (LWE)

## 1.1 Foundations: notation, distributions, and sanity checks

**Problem 1.1** (LWE-1: Dimensions and modular arithmetic sanity)**. Purpose:** Ensure comfort with the data types in LWE, modular arithmetic, and matrix-vector conventions.
**Verifies:** Correct handling of $(A, b) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$, and that the student can detect malformed instances.
**Deliverable:** A one-page write-up listing the type of each variable and a checklist for valid LWE instances.

Let $n, m, q \in \mathbb{N}$, $A \in \mathbb{Z}_q^{m\times n}$, $s \in \mathbb{Z}_q^n$, $e \in \mathbb{Z}^m$ with small entries, and define $b = As + e \bmod q$.

1. Write the dimensions of every term in $b = As + e \bmod q$.
2. Explain why $e$ is sampled from $\mathbb{Z}$ (not $\mathbb{Z}_q$) *before* reduction mod $q$.
3. Give three examples of invalid inputs $(A, b)$ that could not have arisen from the LWE sampling process (with justification).

**Problem 1.2** (LWE-2: Error magnitude and wrap-around)**. Purpose:** Build intuition for when "small errors" remain distinguishable after reduction mod $q$.
**Verifies:** Ability to reason about wrap-around probability and its cryptographic relevance.
**Deliverable:** A short calculation and a paragraph interpretation.

Assume each $e_i$ is supported on $\{-B, \ldots, B\} \subset \mathbb{Z}$, and $b_i = \langle a_i, s\rangle + e_i \bmod q$.

1. For a fixed integer representative of $\langle a_i, s\rangle$ in $[0, q-1]$, characterize when $b_i$ wraps around modulo $q$.
2. Give an upper bound on the probability of wrap-around if $\langle a_i, s\rangle$ is uniform in $\{0, \ldots, q-1\}$ and $e_i$ is uniform in $\{-B, \ldots, B\}$.
3. Explain (qualitatively) why excessive wrap-around complicates correctness in LWE-based encryption.

## 1.2 Formal games: search vs decision

**Problem 1.3** (LWE-3: Write the decision game formally)**. Purpose:** Train students to express security as an indistinguishability game with clear randomness.
**Verifies:** Proper game syntax: sampling order, adversary view, and advantage definition.
**Deliverable:** A complete game-based definition and Adv expression.

Write the LWE decision experiment $\mathsf{Exp}_{\mathcal{A}}^{\mathsf{dLWE}}(n, m, q, \chi)$, including: (i) sampling of $A, s, e$, (ii) sampling of challenge bit $b \in \{0, 1\}$, (iii) construction of the challenge pair, (iv) adversary output, and (v) advantage.

**Problem 1.4** (LWE-4: Search $\Rightarrow$ decision (easy direction))**. Purpose:** Ensure the student sees the trivial reduction direction.
**Verifies:** Ability to design a black-box reduction using an oracle.
**Deliverable:** Reduction algorithm (pseudocode) and correctness argument.

Assume an oracle $\mathcal{O}$ that solves search-LWE: on input $(A, b)$ sampled from $\mathcal{D}_0$, it returns $s$ with probability $\geq 2/3$. Construct a distinguisher $\mathcal{D}^{\mathcal{O}}$ for decision-LWE and analyze its advantage.

**Problem 1.5** (LWE-5: Decision $\Rightarrow$ search (nontrivial direction; outline))**. Purpose:** Practice hybrid/embedding logic that underlies many cryptographic reductions.
**Verifies:** Understanding of how to "program" distributions and recover secret coordinates.
**Deliverable:** A structured outline with explicit hybrids and what each query to the decision oracle tests.

Assume $q$ is prime and you have a decision-LWE distinguisher $\mathcal{A}$ with non-negligible advantage. Give an outline to recover $s$ (or $s$ coordinate-by-coordinate). Your outline must include:

1. how you re-randomize or transform samples without destroying the LWE form,

2. a hybrid argument or hypothesis test used to extract information about $s$,

3. how many oracle calls are needed and why.

## 1.3 Attack reasoning and parameter thinking

**Problem 1.6** (LWE-6: Primal vs dual attack identification)**. Purpose:** Teach students to classify attacks by viewpoint (BDD/primal vs dual) and required resources.
**Verifies:** Correct mapping from problem statement to attack pipeline and success condition.
**Deliverable:** Two short write-ups (primal and dual) with "input $\rightarrow$ lattice $\rightarrow$ target vector" flow.

For each of the following, describe the attacker objective and the "tell" used:

1. A primal (BDD-style) attack on LWE.

2. A dual attack using a short $y$ satisfying $y^{\top} A \equiv 0 \pmod{q}$.

Your answer must specify what distributional property distinguishes LWE samples from uniform in each case.

**Problem 1.7** (LWE-7: Failure modes under parameter changes)**. Purpose:** Build intuition for why parameters are chosen in "narrow windows."
**Verifies:** Ability to reason about edge cases and what breaks first (security vs correctness).
**Deliverable:** A table with three parameter modifications and predicted outcomes.

Consider three hypothetical changes (one at a time):

1. Reduce $q$ by a factor of 2 with the same $\chi$.

2. Keep $q$ fixed but double the error width (e.g., $\sigma \mapsto 2\sigma$).

3. Keep $q, \chi$ fixed but halve $m$.

For each change, predict (qualitatively) its effect on (i) correctness of decryption and (ii) security against lattice reduction.

**Problem 1.8** (LWE-8 (Advanced): Construct a distinguishing statistic)**. Purpose:** Force students to operationalize the "smallness mod $q$" intuition.
**Verifies:** Ability to design a statistical test and justify why it has bias.
**Deliverable:** A defined statistic, justification of bias, and discussion of limitations.

Suppose you are given a candidate short vector $y \in \mathbb{Z}^m$ with $y^\top A \equiv 0 \pmod{q}$. Propose a statistic $T(A, b)$ computed from $y^\top b \bmod q$ that tends to be smaller for LWE samples than for uniform. Explain: (i) what distribution you expect for $y^\top e$ (heuristically), (ii) why uniform $b$ makes $y^\top b$ uniform in $\mathbb{Z}_q$, (iii) what could go wrong if $y$ is not sufficiently short.

# 2 Lattice-Based: Short Integer Solution (SIS)

## 2.1 Existence and basic properties

**Problem 2.1** (SIS-1: Pigeonhole existence of collisions)**. Purpose:** Connect SIS to collision resistance and existence via counting.
**Verifies:** Comfort with combinatorial counting in $\mathbb{Z}_q^n$.
**Deliverable:** A clean counting argument and a constructed nonzero short vector.

Let $A \in \mathbb{Z}_q^{n \times m}$ and consider the set $\mathcal{X} = \{0, 1\}^m$.

1. Show that if $2^m > q^n$, then there exist distinct $x, x' \in \mathcal{X}$ with $Ax \equiv Ax' \pmod{q}$.

2. Define $z = x - x' \in \{-1, 0, 1\}^m$ and show $Az \equiv 0 \pmod{q}$.

3. Bound $\|z\|_2$ and $\|z\|_\infty$.

**Problem 2.2** (SIS-2: SIS as collision-finding in linear hash)**. Purpose:** Make explicit the standard binding/collision reduction.
**Verifies:** Ability to formalize reduction: collision $\mapsto$ SIS solution.
**Deliverable:** A reduction statement with assumptions on domain/range.

Define a hash $H : \mathcal{D} \to \mathbb{Z}_q^n$ by $H(x) = Ax \bmod q$ where $\mathcal{D} \subset \mathbb{Z}^m$ is a set of short vectors. Show that any collision $x \neq x'$ in $\mathcal{D}$ yields a nonzero $z = x - x'$ solving SIS with bound related to the diameter of $\mathcal{D}$.

## 2.2 Norms, solution sets, and structure

**Problem 2.3** (SIS-3: Kernel lattice viewpoint)**. Purpose:** Train the student to translate modular constraints into an integer lattice.
**Verifies:** Correct construction of the solution lattice and interpretation of short vectors.
**Deliverable:** Definition of a lattice $\Lambda$ of solutions and a proof that SIS asks for a short nonzero vector in it.

Given $A \in \mathbb{Z}_q^{n \times m}$, define

$$\Lambda = \{x \in \mathbb{Z}^m : Ax \equiv 0 \pmod{q}\}.$$

1. Prove $\Lambda$ is a full-rank lattice in $\mathbb{Z}^m$.

2. Show that SIS is exactly: find a nonzero $x \in \Lambda$ with $\|x\| \leq \beta$.

3. Explain how changing the norm from $\ell_2$ to $\ell_\infty$ changes what "short" means for attacks.

**Problem 2.4** (SIS-4 (Advanced): Parameter tradeoff reasoning)**. Purpose:** Build the parameter intuition: $m$ impacts existence; $\beta$ impacts hardness.
**Verifies:** Ability to reason with volume/entropy style heuristics.
**Deliverable:** A heuristic estimate and qualitative conclusion.

Give a heuristic argument for how large $\beta$ must be (as a function of $q, n, m$) before you expect $\Lambda$ to contain a nonzero vector with $\|x\|_2 \leq \beta$. Your answer should:

1. identify the "density" of $\Lambda$ inside $\mathbb{Z}^m$,

2. compare the number of lattice points in a ball of radius $\beta$ to the index of $\Lambda$,

3. conclude a threshold relationship (up to polynomial factors).

# 3 Lattice-Based: NTRU Search

## 3.1 Ring mechanics and problem embedding

**Problem 3.1** (NTRU-1: Ring arithmetic and invertibility checks)**. Purpose:** Ensure students can operate in $R_q = \mathbb{Z}_q[x]/(f(x))$ and understand when inverses exist.
**Verifies:** Ability to state invertibility conditions and compute in quotient rings.
**Deliverable:** A precise criterion and a worked small example (toy parameters).

Let $R_q = \mathbb{Z}_q[x]/(f(x))$ for monic $f$ of degree $N$.

1. State a criterion for when $f \in R_q$ (an element, not the modulus polynomial) is invertible in $R_q$.

2. In a toy ring (choose small $q, N$), pick a simple element $\bar{f}(x)$ and compute $\bar{f}^{-1}(x) \bmod (f(x), q)$, if it exists.

3. Explain why NTRU key generation must reject some sampled $f$.

**Problem 3.2** (NTRU-2: From public key to NTRU lattice equation)**. Purpose:** Make explicit the core relation and how it becomes a lattice problem.
**Verifies:** Ability to derive constraints and define the NTRU lattice.
**Deliverable:** Definition of $\Lambda_h$ and membership proof.

Given $h \equiv gf^{-1} \pmod{q}$ in $R_q$, show that $hf \equiv g \pmod{q}$. Define

$$\Lambda_h = \{(u, v) \in R^2 : u - hv \equiv 0 \pmod{q}\}.$$

Prove that $(g, f) \in \Lambda_h$ and explain why $(g, f)$ is expected to be short.

**Problem 3.3** (NTRU-3 (Advanced): Ambiguity and equivalent secrets)**. Purpose:** Train students to notice non-uniqueness and equivalence classes of secrets.
**Verifies:** Ability to reason about multiple short representations and normalization constraints.
**Deliverable:** A short essay with at least two distinct equivalences and how schemes fix them.

In NTRU-like systems, multiple $(f, g)$ may yield the same $h$. Identify at least two sources of equivalence (e.g., multiplication by units, sign, rotations in certain rings) and explain how practical schemes constrain keys to reduce ambiguity.

# 4 Code-Based: Syndrome Decoding (SD/DSD/QCSD)

## 4.1 Linear algebra basics and cosets

**Problem 4.1** (SD-1: Coset structure of syndrome equations)**. Purpose:** Establish that solutions form an affine space; decoding is weight minimization within a coset.
**Verifies:** Ability to prove coset properties and compute sizes.
**Deliverable:** Proof and size computation.

Let $H \in \mathbb{F}_2^{(n-k)\times n}$ have full rank and fix $s \in \mathbb{F}_2^{n-k}$.

1. Show $\{e \in \mathbb{F}_2^n : He^\top = s\}$ is either empty or an affine subspace of dimension $k$.

2. Show it is never empty when $H$ has full rank.

3. Compute the number of solutions in terms of $k$.

**Problem 4.2** (SD-2: Expected number of weight-$w$ solutions)**. Purpose:** Teach the "random syndrome" heuristic used in parameter selection.
**Verifies:** Ability to compute expectation and interpret regimes where unique solutions likely exist.
**Deliverable:** Calculation and regime discussion.

Assume $H$ is uniformly random full-rank. For fixed $s$, estimate the expected number of solutions $e$ with $\mathrm{wt}(e) = w$.

1. Use linearity to argue each fixed $e$ satisfies $He^\top = s$ with probability $2^{-(n-k)}$.

2. Conclude $\mathbb{E}[\#\{e : \mathrm{wt}(e) = w, He^\top = s\}]$.

3. Interpret what it means when this expectation is $\ll 1$, $\approx 1$, and $\gg 1$.

## 4.2 ISD thinking and complexity

**Problem 4.3** (SD-3: Prange ISD success probability)**. Purpose:** Verify comprehension of the simplest ISD algorithm and its probabilistic analysis.
**Verifies:** Ability to compute success probability and expected trials.
**Deliverable:** A derivation with combinatorial terms and an asymptotic discussion.

In Prange's algorithm, an "information set" $I$ of size $k$ is guessed, and the algorithm succeeds if the error has zero weight on $I$.

1. Derive $p = \Pr[\text{success}]$ in terms of $n, k, w$.

2. Give the expected number of trials $1/p$.

3. Explain how $w/n$ and $k/n$ affect $p$ qualitatively.

**Problem 4.4** (SD-4 (Advanced): Compare ISD refinements conceptually)**. Purpose:** Ensure students can articulate why modern ISD improves Prange (without memorizing constants).
**Verifies:** Ability to explain meet-in-the-middle / representations / partial weight splitting ideas.
**Deliverable:** A comparative explanation (2–3 pages) with a schematic of each method.

Give a conceptual comparison of at least three ISD variants (e.g., Prange, Stern, Dumer, BJMM-style):

1. What is guessed and what is solved deterministically?

2. Where does meet-in-the-middle or list-merging appear?

3. What resource dominates (time vs memory), and why?

No numeric constants required; focus on mechanisms.

### 4.3 QCSD structure and pitfalls

**Problem 4.5** (QCSD-1: Circulant blocks as polynomial multiplication)**. Purpose:** Make QC structure operational: convert between matrices and polynomials.
**Verifies:** Ability to implement the representation transformation on paper.
**Deliverable:** A worked example with $p \times p$ circulant block action.

Let $C$ be a $p \times p$ circulant matrix over $\mathbb{F}_2$ determined by first row $(c_0, \ldots, c_{p-1})$. Let $c(x) = \sum_{i=0}^{p-1} c_i x^i$. Show that multiplying $C$ by a vector $v$ corresponds to $c(x) \cdot v(x) \bmod (x^p - 1)$ under the natural identification.

**Problem 4.6** (QCSD-2 (Advanced): Symmetry and attack surface)**. Purpose:** Force students to analyze how structure can reduce effective security.
**Verifies:** Ability to propose plausible structural attack avenues and required conditions.
**Deliverable:** Threat analysis memo listing at least two structural shortcuts and how parameters mitigate them.

QC structure introduces cyclic shifts and module structure. Propose at least two distinct ways this could reduce the work factor for decoding (e.g., folding, exploiting repeated patterns, reducing dimension in transformed domains). For each, state:

1. what property of the QC ensemble is being exploited,

2. what additional assumption the attacker needs,

3. what countermeasure parameter choices can help.

## 5 Isogeny-Based Hard Problems

### 5.1 Definitions and graph intuition

**Problem 5.1** (ISO-1: Isogeny basics—kernel and quotient viewpoint)**. Purpose:** Verify the student knows what data defines an isogeny at a high level.
**Verifies:** Ability to explain kernel $\leftrightarrow$ isogeny relationship and quotient intuition.
**Deliverable:** A short explanation with definitions (no heavy EC arithmetic).

Let $\phi : E \to E'$ be an isogeny of elliptic curves over $\mathbb{F}_q$.

1. Define $\ker(\phi)$ and state why it is finite.

2. Explain informally why $E' \cong E/\ker(\phi)$ as groups.

3. State (without proof) how the degree relates to $\left|\ker(\phi)\right|$ in separable cases.

**Problem 5.2** (ISO-2: Path-finding heuristic in regular graphs)**. Purpose:** Build generic complexity intuition for graph-based isogeny problems.
**Verifies:** Ability to reason about meet-in-the-middle vs one-sided random walks.
**Deliverable:** A heuristic complexity estimate with assumptions stated.

Model the isogeny graph as a $d$-regular expander with $N$ vertices.

1. Estimate the expected time for a one-sided random walk from $E_1$ to hit $E_2$.

2. Estimate the expected time for a bidirectional meet-in-the-middle strategy (two frontiers that collide).

3. State how memory affects the second approach.

### 5.2 CSIDH-style action inversion

**Problem 5.3** (CSIDH-1: Group action axioms and what must be shown)**. Purpose:** Ensure students can state precisely what it means for an ideal class group to act on curves.
**Verifies:** Ability to list axioms and identify what "inversion" asks for.
**Deliverable:** A formal definition of a group action and mapping to the CSIDH statement.

Let $G$ be a group acting on a set $\mathcal{X}$ via $* : G \times \mathcal{X} \to \mathcal{X}$.

1. State the two axioms of a group action.

2. In CSIDH-style settings, interpret $E_1 = a * E_0$ as analogous to $g^a$ in Diffie–Hellman.

3. Formulate the inversion problem as a search problem and specify what counts as an "equivalent" solution.

**Problem 5.4** (CSIDH-2 (Advanced): Distinguish path-finding vs action inversion)**. Purpose:** Prevent a common confusion: endpoints in the same isogeny class do not uniquely determine the action element.
**Verifies:** Ability to separate representation issues and commutativity effects.
**Deliverable:** A careful argument (1–2 pages) with an explicit example of non-uniqueness (conceptual).

Explain why "find *some* isogeny path from $E_0$ to $E_1$" may be easier (or different) than "recover the specific secret class group element $a$". Your answer must discuss:

1. multiple decompositions of an action element into prime ideals,

2. commutativity and re-ordering of steps,

3. what information is lost when only endpoints are given.

## 6 Multivariate (MQ / IP Variants)

### 6.1 MQ formulation and linearization

**Problem 6.1** (MQ-1: Count monomials and set up relinearization)**. Purpose:** Ensure students can translate MQ into linear algebra over monomials.
**Verifies:** Correct counting and formation of the linear system in lifted space.
**Deliverable:** Monomial count and explicit lifted-variable mapping.

Let $f_1, \ldots, f_m$ be quadratic polynomials in $n$ variables over $\mathbb{F}_q$.

1. Count the number of distinct quadratic monomials $x_i x_j$ with $1 \le i \le j \le n$.

2. Define lifted variables $y_{ij} = x_i x_j$ and rewrite each $f_\ell$ as a linear function in $\{y_{ij}\}$ and $\{x_i\}$ and constant term.

3. Explain why this does *not* immediately solve MQ (what constraints are missing?).

**Problem 6.2** (MQ-2: Random-system solution heuristic)**. Purpose:** Build baseline intuition: random maps have about $q^{n-m}$ preimages.
**Verifies:** Ability to compute a heuristic expected number of solutions and interpret under/overdetermined regimes.
**Deliverable:** A short calculation and discussion.

Assume $F : \mathbb{F}_q^n \to \mathbb{F}_q^m$ behaves like a uniformly random function.

1. For a fixed $y \in \mathbb{F}_q^m$, compute the expected number of $x$ such that $F(x) = y$.

2. Discuss what changes when $m < n$, $m = n$, and $m > n$.

3. Explain why many MQ cryptosystems choose $m \approx n$ but rely on special trapdoor structure.

### 6.2 IP/key-recovery context

**Problem 6.3** (IP-1: Equivalence of private keys under affine changes)**. Purpose:** Teach that private keys are often defined up to equivalence; attacks may aim for any equivalent key.
**Verifies:** Understanding of conjugation by affine maps and what information it preserves.
**Deliverable:** A proof of equivalence relation and a short discussion of its cryptographic meaning.

Let $P = T \circ F \circ S$ with invertible affine maps $S, T$ and central map $F$.

1. Show that if $U, V$ are invertible affine maps, then

$$P = (T \circ V^{-1}) \circ (V \circ F \circ U) \circ (U^{-1} \circ S)$$

   is another decomposition with a "modified" central map.

2. Explain why recovering *exactly* $(S, F, T)$ may be unnecessary to break a signature scheme; an equivalent decomposition may suffice.

**Problem 6.4** (MQ-3 (Advanced): Hybrid attack design)**. Purpose:** Verify students can design a plausible solver combining guessing and algebra.
**Verifies:** Ability to propose a hybrid strategy, estimate complexity, and state assumptions.
**Deliverable:** Algorithm sketch + complexity expression in terms of $q, n, m, t$ (guessed vars).

Propose a hybrid algorithm that guesses $t$ variables and solves the remaining system using an algebraic method (e.g., Gröbner basis or linearization). Your answer must:

1. define the resulting reduced system size,

2. express the total cost as $q^t \cdot \mathrm{Cost}(n - t, m)$,

3. discuss how to choose $t$ to minimize the total cost (conceptually).

## 7 Hash-Based: Security Games (Preimage / 2nd-preimage / Collision / PRF)

### 7.1 Random oracle heuristics and bounds

**Problem 7.1** (HASH-1: Preimage work factor (random function heuristic))**. Purpose:** Ensure students can derive baseline $2^n$ bounds for $n$-bit outputs.

**Verifies:** Correct probability calculations for random guessing.
**Deliverable:** A probability derivation and expected trials estimate.

Model $H : \{0,1\}^* \to \{0,1\}^n$ as a random function. Given $y \in \{0,1\}^n$,

1. What is the success probability of one random query $x$ such that $H(x) = y$?

2. What is the expected number of queries to succeed?

3. How many queries are needed for success probability at least $1/2$? (Use $(1-p)^Q \approx e^{-pQ}$.)

**Problem 7.2** (HASH-2: Birthday bound for collisions)**. Purpose:** Verify the student can derive the $2^{n/2}$ phenomenon.
**Verifies:** Ability to compute collision probability after $Q$ random samples.
**Deliverable:** A derivation with approximation steps clearly labeled.

Let $H$ be a random function to $n$ bits. Consider $Q$ uniformly random inputs $x_1, \ldots, x_Q$.

1. Derive an approximation for $\Pr[\exists i \neq j : H(x_i) = H(x_j)]$.

2. Solve for $Q$ that yields constant collision probability (e.g., about $1/2$).

3. Explain why this does *not* immediately imply second-preimage attacks of cost $2^{n/2}$.

### 7.2 PRF distinguishing

**Problem 7.3** (HASH-3: PRF advantage definition from first principles)**. Purpose:** Train precise game-writing and advantage expression.
**Verifies:** Correct definition of oracles, randomness, and advantage.
**Deliverable:** A complete PRF game definition with $\mathrm{Adv}^{\mathrm{prf}}$.

Let $H_k$ be a keyed hash family. Write the PRF experiment where the challenger flips $b \in \{0,1\}$ and gives oracle access to either $H_k(\cdot)$ or a random function $R(\cdot)$. Define $\mathrm{Adv}^{\mathrm{prf}}_{\mathcal{A}}$.

**Problem 7.4** (HASH-4 (Advanced): Hybrid proof skeleton for a hash-based signature)**. Purpose:** Verify capability to structure a reduction using hybrids.
**Verifies:** Ability to identify which primitive property is used at which transition (PRF vs collision vs preimage).
**Deliverable:** A hybrid sequence (H0,H1,H2,...) with one-paragraph justification per step.

Consider a generic hash-based signature scheme that uses: (i) a keyed hash as a PRF to derive per-message randomness, and (ii) collision resistance of a compression function inside authentication paths. Sketch a hybrid proof that reduces EUF-CMA forgery to either breaking PRF security or finding a collision. Your hybrids must explicitly indicate what is replaced and what the simulator must answer.

## 8 Synthesis: Comparative and Integrative Problems

**Problem 8.1** (SYN-1: Map each hard problem to its "cryptographic role")**. Purpose:** Ensure students can connect assumptions to constructions and proof styles.
**Verifies:** Correct mapping (LWE→PKE/KEM; SIS→hash/commitment; SD→KEM; MQ→signatures; hash games→hash signatures; isogenies→key exchange/KEM).
**Deliverable:** A table with 2–3 sentence justification per row.

Create a table with columns:

Assumption | Typical Primitive | Proof style (game/reduction) | Dominant attack family.

Fill it for: LWE (search/decision), SIS, NTRU, SD/DSD/QCSD, isogeny path/action inversion, MQ/IP, and hash security games.

**Problem 8.2** (SYN-2 (Advanced): "What if structure helps the attacker?" essay)**. Purpose:** Train research-grade skepticism about algebraic structure.
**Verifies:** Ability to articulate why structure reduces sizes and where it can leak exploitable invariants.
**Deliverable:** A 3–5 page memo with at least three case studies.

Write a memo comparing three structured settings:

1. ring/module lattices (Ring-LWE / Module-LWE style),

2. QC codes (QCSD),

3. structured central maps in MQ.

For each, answer:

1. What efficiency benefit does the structure provide?

2. What invariants or symmetries might an attacker exploit?

3. What parameter/instance-generation mitigations are standard?