

End Sem Exam

Full Marks: 45, Time: 3 hours

Instructor: Sruthi Sekar

Date: 14th Nov 2025

Roll Number: _____ Name: _____

1. Answer each question on a new page of the answer booklet.
2. Do not use pencils. Pens only!
3. If you are using a fact *proved* (not just stated or given in exercise) in class during lectures, please mention the fact clearly and intuitively explain it before using it in your questions.

Problem 1 (4 marks (1+1+1+1)). Let $\Pi_1 = (\text{Gen}_1, \text{Enc}_1, \text{Dec}_1)$ and $\Pi_2 = (\text{Gen}_2, \text{Enc}_2, \text{Dec}_2)$ be two PKE schemes such that *at least one of the two* is IND-CPA secure against a quantum Evelyn, but we don't know which one. Now consider the following constructions $\Pi = ((\text{Gen}, \text{Enc}, \text{Dec}))$ using Π_1 and Π_2 . Each construction uses $\text{Gen}(1^n)$: generates $(pk_1, sk_1) \leftarrow \text{Gen}_1$, $(pk_2, sk_2) \leftarrow \text{Gen}_2$ and outputs $pk = (pk_1, pk_2)$, $sk = (sk_1, sk_2)$; and the Dec is defined in the natural way for each.

- a) $\text{Enc}(pk, m) := (\text{Enc}_1(pk_1, m), \text{Enc}_2(pk_2, m))$.
- b) $\text{Enc}(pk, m) := \text{Enc}_1(pk_1, \text{Enc}_2(pk_2, m))$.
- c) $\text{Enc}(pk, m) := \text{Enc}_2(pk_2, \text{Enc}_1(pk_1, m))$.
- d) $\text{Enc}(pk, m) := (\text{Enc}_1(pk_1, r), \text{Enc}_2(pk_2, m \oplus r))$ for $r \in_R \{0, 1\}^{|m|}$.

For each construction, is Π IND-CPA secure? Answer yes/no along with an intuitive explanation. (Full proof is not expected but a clear and complete reasoning is expected.)

Problem 2 (3 marks (1+1+1)). Let $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ be an IND-CPA secure PKE, $G : \{0, 1\}^n \rightarrow \{0, 1\}^\ell$ be a PRG and $\mathcal{F} := \{f_k : \{0, 1\}^n \rightarrow \{0, 1\}^\ell\}_{k \in \{0, 1\}^n}$ be a PRF. Assume Π has perfect correctness. For each scheme below, the intent is to design a PKE that uses fewer random bits than Π by making use of our favorite pseudorandom objects. For each part, is the modified $\Pi' = (\text{Gen}', \text{Enc}', \text{Dec}')$ (Dec' same as Dec in each part) IND-CPA secure? No detailed proof needed but a clear and complete intuitive reasoning is expected in each part.

- a) $\text{Gen}'(1^n) = \text{Gen}(1^n)$; $\text{Enc}'(pk, m; r) = \text{Enc}(pk, m; G(r))$.
- b) $\text{Gen}'(1^n) = \text{Gen}(1^n)$; $\text{Enc}'(pk, m; r) = \text{Enc}(pk, m; f_r(m))$.
- c) $\text{Gen}'(1^n; \rho || k)$: generate $(pk, sk) \leftarrow \text{Gen}(1^n; \rho)$, output $(pk || k, sk)$;
 $\text{Enc}'(pk || k, m) := \text{Enc}(pk, m; f_k(m))$.

Problem 3 (6 marks (3+3)). Let f be a length-preserving one-way function. Are the following functions one-way? If yes, give a formal proof by reduction, else give a complete attack through a counter-example.

- a) $f_1(x) := f(x) \oplus x$
- b) $f_2(x) := y_1 || y_2 || \dots || y_{n-1}$, where $y = y_1 || y_2 || \dots || y_{n-1} || y_n := f(x)$.

Problem 4 (4 marks (2+2)). For each part below, show that Π is not IND-CCA secure through an explicit attack.

1. Let $\Pi' = (\text{Gen}', \text{Enc}', \text{Dec}')$ be an IND-CCA secure PKE for $\mathcal{M} = \{0, 1\}$. Π is defined as:
 $\text{Gen}(1^n) = \text{Gen}'(1^n)$;
 $\text{Enc}(pk, m = m_1 || \dots || m_\ell) := \text{Enc}'(pk, m_1) || \dots || \text{Enc}'(pk, m_\ell)$; Dec the natural one.
2. $\text{Gen}(1^n)$: Pick $A \in_R \mathbb{Z}_q^{m \times n}$, $\mathbf{s} \in_R \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \chi^m$. Output $(pk = A, As + \mathbf{e}, sk = \mathbf{s})$;
 $\text{Enc}(pk, b)$: Sample $\mathbf{r} \in_R \{0, 1\}^m$, encode $b \in \{0, 1\}$ as $\hat{b} = b \cdot \lfloor q/2 \rfloor$.
Output $c := (\mathbf{r}^T A, \mathbf{r}^T (As + \mathbf{e}) + \hat{b})$.

Problem 5 (9 marks (1+3+3+2)). Recall what a group homomorphism is. Two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ are said to be homomorphic if there exists a bijection $\phi : \mathbb{G}_1 \rightarrow \mathbb{G}_2$ such that $\phi(g *_{\mathbb{G}_1} h) = \phi(g) *_{\mathbb{G}_2} \phi(h)$. For simplicity, we denote $*_{\mathbb{G}_1}$ by addition and $*_{\mathbb{G}_2}$ by multiplication. The following is a sigma protocol for the relation $\mathcal{R}_{\text{hom}} := \{((h, \phi), g) \in (\mathbb{G}_2 \times \mathcal{F}) \times \mathbb{G}_1 : \phi(g) = h\}$. On a high level, the prover wants to convince the verifier that it knows the pre-image of $h \in \mathbb{G}_2$ under ϕ . This is a useful generalization of the few sigma protocols that you have seen. P and V share the statement (h, ϕ) , and P additionally has the witness g .

- i) P picks $k \in_R \mathbb{G}_1$ and sends $I = \phi(k)$ to V .
- ii) V sends a random integer $r \in_R \{0, 1, \dots, N-1\}$ for some integer N .
- iii) P sends $s := (k + g \cdot r) \in \mathbb{G}_1$ to V .
- iv) V accepts if and only if $\phi(s) = I \cdot h^r$.

Prove that $\Pi_{\text{hom}} = (P, V)$ is a sigma protocol for $\mathcal{R}_{\text{hom}}$, i.e., prove the following:

- a) Show that Π_{hom} is complete.
- b) Show that Π_{hom} satisfies special soundness.
- c) Show that Π_{hom} satisfies honest verifier ZK.
- d) Can you show how this protocol captures the Chaum-pedersen sigma protocol (we saw in class) for $\mathcal{R}_{DH} := \{((h, u, v), x) \in \mathbb{G}^3 \times \mathbb{Z}_q : u = g^x \text{ and } v = h^x\}$?

Problem 6 (6 marks (3+3)). Given a commitment scheme $\Pi = (\text{Commit}, \text{Receive})$ for n -bit messages with computational hiding and perfect binding, consider $\Pi^{\text{mult}} = (\text{Commit}^{\text{mult}}, \text{Receive}^{\text{mult}})$ for $2n$ -bit messages, defined as below.

- $\text{Commit}^{\text{mult}}(m_1 || m_2; r_1 || r_2) := \text{Commit}(m_1; r_1) || \text{Commit}(m_2; r_2)$.
- $\text{Receive}^{\text{mult}}(\text{com}_1 || \text{com}_2, m_1 || m_2, r_1 || r_2)$: output 1 if $\text{Receive}(\text{com}_1, m_1, r_1) = 1$ and $\text{Receive}(\text{com}_2, m_2, r_2) = 1$, and 0, otherwise.

Show that Π^{mult} is a commitment scheme, satisfying:

- a) Π^{mult} is computational hiding (via a hybrid argument).
- b) Π^{mult} is perfect binding.

Problem 7 (6 marks (3+3)).

- a) Let $\mathcal{G}(1^n) \rightarrow (\mathbb{G}, q, g)$, where q is prime. Suppose CDH is hard relative to $\mathcal{G}$. Consider a special case of CDH, called the *Squaring*-Diffie-Hellman (SDH) problem, which finds g^{x^2} given g^x . Show that the SDH problem is also hard relative to $\mathcal{G}$, i.e., $\forall$ PPT **SqLogan**, $\exists$ negligible negl s.t. $\forall n$:

$$\Pr_{\substack{(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n) \\ x \in_R \mathbb{Z}_q}} [\text{SqLogan}(1^n, (\mathbb{G}, q, g), g^x) = h : h = g^{x^2}] \leq \text{negl}(n)$$

(**Hint:** computing square-roots is easy in prime-order groups and you will have to invoke **SqLogan** multiple times!)

- b) Consider the *worst-case* (for secrets $\mathbf{s}$) version of DLWE assumption, denoted by (n, m, q, χ) -DLWE': for all quantum polynomial **Evelyn** and all $\mathbf{s} \in \mathbb{Z}_q^n$, $\exists$ negligible negl s.t. $\forall n$:

$$\left| \Pr_{\substack{A \in_R \mathbb{Z}_q^{m \times n} \\ \mathbf{e} \leftarrow \chi^m}} [\text{Evelyn}(A, A\mathbf{s} + \mathbf{e}) = 1] - \Pr_{\substack{A \in_R \mathbb{Z}_q^{m \times n} \\ \mathbf{r} \in_R \mathbb{Z}_q^m}} [\text{Evelyn}(A, \mathbf{r}) = 1] \right| \leq \text{negl}(n)$$

Show that (n, m, q, χ) -DLWE' $\implies$ (n, m, q, χ) -DLWE. (**Hint:** Exploit linearity).

Problem 8 (7 marks (1+3+3)). Let $\Pi = (\text{Gen}, \text{Enc}, \text{Dec}, \text{Eval})$ be a fully homomorphic encryption. Now, consider the following interactive protocol $\Pi^{IP} = (P, V)$ based on the FHE Π . The (unbounded) prover P and verifier V share an instance $x \in L$ for an NP language L with corresponding relation $\mathcal{R}$ ($x \in L$ iff $\exists w: (x, w) \in \mathcal{R}$).

- i) P computes a witness w for x .
- ii) P samples a key pair $(pk, sk) \leftarrow \text{Gen}(1^{|x|})$ and sends $c \leftarrow \text{Enc}(pk, w)$ to V .
- iii) V tosses $b \in_R \{0, 1\}$ and sends c' to P , where
 - if $b = 1$, $c' \leftarrow \text{Eval}(pk, c, C_{\mathcal{R}}(x, \cdot))$
(Here, $C_{\mathcal{R}}(x, w)$ outputs 1 if $(x, w) \in \mathcal{R}$ and 0, otherwise.)
 - if $b = 0$, $c' \leftarrow \text{Enc}(pk, 0)$
- iv) P decrypts $b' = \text{Dec}(sk, c')$ and sends b' to V .
- v) V accepts if and only $b' = b$.

Formally prove that Π^{IP} is a computational honest verifier ZKP for L :

- a) Show that Π^{IP} is complete. When can it achieve perfect completeness?
- b) Show that Π^{IP} is sound. What is the soundness error?
- c) Show that Π^{IP} is computational honest-verifier ZK. Describe the simulator and formally prove that the simulated view is computationally indistinguishable from the real view.

Please state the exact properties you need from Π for each part above.

Definitions you might need

Discrete Log Assumption: The discrete log problem is hard relative to $\mathcal{G}$ if for all PPT Logan, $\exists$ negligible function negl such that $\forall n$

$$\Pr_{\substack{(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n) \\ h \in_R \mathbb{G}}} [\text{Logan}(1^n, (\mathbb{G}, q, g), h) = x : g^x = h] \leq \text{negl}(n)$$

Computational Diffie-Hellman (CDH) Assumption: The CDH problem is hard relative to $\mathcal{G}$ if for all PPT Logan, $\exists$ negligible function negl such that $\forall n$

$$\Pr_{\substack{(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n) \\ x, y \in_R \mathbb{Z}_q}} [\text{Logan}(1^n, (\mathbb{G}, q, g), g^x, g^y) = h : h = g^{xy}] \leq \text{negl}(n)$$

Decisional Diffie-Hellman (DDH) Assumption: The DDH problem is hard relative to $\mathcal{G}$ if for all PPT Logan, $\exists$ negligible function negl such that $\forall n$, the following is $\leq \text{negl}(n)$:

$$\left| \Pr_{\substack{(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n) \\ x, y \in_R \mathbb{Z}_q}} [\text{Logan}(1^n, (\mathbb{G}, q, g), (g^x, g^y, g^{xy})) = 1] - \Pr_{\substack{(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n) \\ x, y, z \in_R \mathbb{Z}_q}} [\text{Logan}(1^n, (\mathbb{G}, q, g), (g^x, g^y, g^z)) = 1] \right|$$

Factoring Assumption: $\text{GenModulus}(1^n) \rightarrow (N, p, q)$ such that $N = pq$, where p and q are n -bit primes (except with probability $\text{negl}(n)$). Factoring is hard relative to GenModulus if $\forall$ PPT algorithms Fac , $\exists \text{negl}$ such that $\forall n$:

$$\Pr_{(N, p, q) \leftarrow \text{GenModulus}(1^n)} [\text{Fac}(1^n, N) \rightarrow (p', q') \text{ s.t. } (p', q' > 1) \wedge (p' \cdot q' = N)] \leq \text{negl}(n)$$

RSA Assumption: $\text{GenRSA}(1^n) \rightarrow (N, e, d)$ such that $N = pq$ (p and q are n -bit primes), $e, d > 0$, $\gcd(e, \phi(N)) = 1$, and $ed = 1 \pmod{\phi(N)}$. The RSA problem is hard relative to GenRSA if $\forall$ PPT algorithms Inv , $\exists \text{negl}$ such that $\forall n$:

$$\Pr_{\substack{(N, e, d) \leftarrow \text{GenRSA}(1^n) \\ y \in_R \mathbb{Z}_N^*}} [\text{Inv}(1^n, N, e, y) \rightarrow x \text{ s.t. } (x \in \mathbb{Z}_N^*) \wedge (x^e = y \pmod{N})] \leq \text{negl}(n)$$

DLWE Assumption: The (n, m, q, χ) -DLWE assumption states that for all quantum polynomial Evelyn, $\exists$ negligible negl s.t. for all n :

$$\left| \Pr_{\substack{A \in_R \mathbb{Z}_q^{m \times n} \\ \mathbf{s} \in_R \mathbb{Z}_q^n, \mathbf{e} \leftarrow \chi^m}} [\text{Evelyn}(A, A\mathbf{s} + \mathbf{e}) = 1] - \Pr_{\substack{A \in_R \mathbb{Z}_q^{m \times n} \\ \mathbf{r} \in_R \mathbb{Z}_q^m}} [\text{Evelyn}(A, \mathbf{r}) = 1] \right| \leq \text{negl}(n)$$

SLWE Assumption: The (n, m, q, χ) -SLWE assumption states that for all quantum polynomial Evelyn, $\exists$ negligible negl s.t. for all n :

$$\Pr_{\substack{A \in_R \mathbb{Z}_q^{m \times n} \\ \mathbf{s} \in_R \mathbb{Z}_q^n, \mathbf{e} \leftarrow \chi^m}} [\text{Evelyn}(A, A\mathbf{s} + \mathbf{e}) = \mathbf{s}] \leq \text{negl}(n)$$

One Way Function (OWFs): $\{f_n : \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}\}_{n \in \mathbb{N}}$ is a *one-way function family* if for $\text{Gen}(1^n) \rightarrow \mathbf{f}_n$, $f_n(x)$ can be computed efficiently for all x ; and for every PPT algorithm Inv , $\exists \text{negl}$ such that $\forall n$

$$\Pr_{x \in_R \{0, 1\}^n} [\text{Inv}(1^n, f_n(x)) \in f_n^{-1}(f_n(x))] \leq \text{negl}(n)$$

Trapdoor Permutation Family (TDPs): $\Pi = (\text{Gen}, f, \text{Inv})$ is a *trapdoor permutation family* if for $\text{Gen}(1^n) \rightarrow (I, \text{td})$, $f_I(x)$ can be computed efficiently for all x ; and for every PPT algorithm Inv , $\exists \text{negl}$ such that $\forall n, \Pr_{x \in_R \{0,1\}^n} [\text{Inv}(I, f_I(x)) \in f_I^{-1}(f_I(x))] \leq \text{negl}(n)$. But, given td , it is easy to invert, i.e., $\exists$ efficient Inv such that $\Pr_{x \in_R \{0,1\}^n} [\text{Inv}(\text{td}, f_I(x)) \neq x] \leq \text{negl}(n)$.

Hard-core Predicate (HCP): A function $\text{hc}_I : \{0,1\}^n \rightarrow \{0,1\}$ is a *hard-core predicate* of a function family $\Pi = (\text{Gen}, f, \text{Inv})$ if for $\text{Gen}(1^n) \rightarrow (I, \text{td})$, hc_I can be computed in polynomial time, and for every PPT algorithm Pred , $\exists$ negligible function negl such that $\forall n$

$$\Pr_{x \in_R \{0,1\}^n} [\text{Pred}(I, f_I(x)) = \text{hc}_I(x)] \leq 1/2 + \text{negl}(n)$$

But, given td , it is easy to compute $\text{hc}_I(x)$, given $f_I(x)$.

Pseudorandom Generators (PRGs): Recall the definition of pseudorandom generators (PRGs). A deterministic polynomial-time algorithm G is a pseudorandom generator (PRG) if it is length-expanding: for any n , and any $k \in \{0,1\}^n$, $G(k) \in \{0,1\}^{\ell(n)}$ with $\ell(n) > n$, and satisfies: for every PPT adversary, Eve , there exists a negligible function negl such that $\forall n$

$$\left| \Pr_{k \in_R \{0,1\}^n} [\text{Eve}(G(k)) = 1] - \Pr_{r \in_R \{0,1\}^{\ell(n)}} [\text{Eve}(r) = 1] \right| \leq \text{negl}(n)$$

Pseudorandom Functions (PRFs): Recall the definition of pseudorandom functions (PRFs). A family of functions $\{f_k : \{0,1\}^n \rightarrow \{0,1\}^n\}_{k \in \{0,1\}^n}$ is a *pseudorandom function (PRF)* if for every PPT adversary, Eve , there exists a negligible function negl such that $\forall n$

$$\left| \Pr_{k \in_R \{0,1\}^n} [\text{Eve}^{f_k(\cdot)}(1^n) = 1] - \Pr_{R \in_R \mathcal{F}_n} [\text{Eve}^{R(\cdot)}(1^n) = 1] \right| \leq \text{negl}(n),$$

where $\mathcal{F}_n$ is the set of functions: $\mathcal{F}_n := \{f : \{0,1\}^n \rightarrow \{0,1\}^n\}$. Here $\text{Eve}^{f_k(\cdot)}$ and $\text{Eve}^{R(\cdot)}$ denote Eve getting oracle access to functions f_k and R , respectively.

Key Exchange protocol: A *two-party key exchange protocol* Π is a probabilistic protocol between two parties **Alice** and **Bob**, where τ represents the transcript of the protocol, i.e., all the messages exchanged between **Alice** and **Bob** during the protocol execution. At the end of the protocol **Alice** locally outputs $k_A \in \{0,1\}^n$ and **Bob** locally outputs $k_B \in \{0,1\}^n$ such that $\forall n$, $\Pr[k_A = k_B] = 1$. Π is said to be *secure against an eavesdropper* if for all PPT Eve , there exists a negligible function negl such that $\forall n$:

$$\left| \Pr_{(k,\tau) \leftarrow \Pi(1^n)} [\text{Eve}(\tau, k) = 1] - \Pr_{\substack{(k,\tau) \leftarrow \Pi(1^n) \\ r \in_R \{0,1\}^n}} [\text{Eve}(\tau, r) = 1] \right| \leq \text{negl}(n)$$

Public Key Encryption (PKE): A *public key encryption scheme* $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ with message space $\mathcal{M}$ satisfies correctness: for every n , for every $(pk, sk) \leftarrow \text{Gen}(1^n)$, and for every $m \in \mathcal{M}$, $\Pr[\text{Dec}(sk, \text{Enc}(pk, m)) = m] = 1$. Recall the following security notion for Π .

- **Chosen Plaintext Attack (CPA):** Define the following game $\text{PubK}_{\text{Eve}, \Pi}^{\text{cpa}}(n)$ between a challenger and the adversary Eve :

1. Challenger generates $(pk, sk) \leftarrow \text{Gen}(1^n)$ and sends pk to Eve . Eve , on input 1^n and pk , outputs a pair of messages m_0, m_1 with $|m_0| = |m_1|$.
2. Challenger picks $b \in_R \{0,1\}$; sends $c_b \leftarrow \text{Enc}(pk, m_b)$ to Eve .

3. Eve outputs a bit b' .
4. Output 1 if $b' = b$, and 0, otherwise.

Π is said to be *secure against a chosen plaintext attack* if for all PPT adversaries Eve, there exists a negligible function negl such that $\forall n, \Pr[\text{PubK}_{\text{Eve}, \Pi}^{\text{cpa}}(n) = 1] \leq 1/2 + \text{negl}(n)$.

- **Chosen Ciphertext Attack (CCA):** Define the following game $\text{PubK}_{\text{Mallory}, \Pi}^{\text{cca}}(n)$ between a challenger and the adversary Mallory:

1. Challenger generates $(pk, sk) \leftarrow \text{Gen}(1^n)$ and sends pk to Mallory and gives oracle access of $\text{Dec}_{sk}(\cdot)$ to Mallory. Mallory, on input $1^n, pk$ and oracle access to $\text{Dec}_{sk}(\cdot)$, outputs a pair of messages m_0, m_1 with $|m_0| = |m_1|$.
2. Challenger picks $b \in_R \{0, 1\}$; sends $c_b \leftarrow \text{Enc}(pk, m_b)$ to Mallory.
3. Mallory continues to get oracle access to $\text{Dec}_{sk}(\cdot)$ (not allowed to query c_b), outputs b' .
4. Output 1 if $b' = b$, and 0, otherwise.

Π is said to be *secure against a chosen ciphertext attack* if for all PPT adversaries Mallory, there exists a negligible function negl such that $\forall n, \Pr[\text{PubK}_{\text{Mallory}, \Pi}^{\text{cca}}(n) = 1] \leq 1/2 + \text{negl}(n)$.

Digital Signatures: A *digital signature scheme* $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ with message space $\mathcal{M}$ satisfies correctness. Π is *q-time existentially unforgeable under chosen-message attack* if for all PPT, Mallory $\exists$ a negligible negl such that $\forall n, \Pr[\text{SigForge}_{\text{Mallory}, \Pi}^{\text{q-cma}}(n) = 1] \leq \text{negl}(n)$. Here, the game $\text{SigForge}_{\text{Mallory}, \Pi}^{\text{q-cma}}(n)$, between a challenger and the adversary Mallory is defined as follows:

1. Challenger generates $(pk, sk) \leftarrow \text{Gen}(1^n)$ and gives, pk and oracle access of $\text{Sign}_{sk}(\cdot)$ to Mallory. Let $Q := \{m_1, \dots, m_q\}$ be all the queries made by Mallory.
2. Mallory outputs (m, σ) .
3. Output 1 if $\text{Vrfy}_{pk}(m, \sigma) = 1$ and $m \notin Q$, and 0, otherwise.

Collision Resistant Hash Function: A pair of PPT algorithms (Gen, H) with $\text{Gen}(1^n) \rightarrow k$ picks the key k which decides the specific hash function $H^k : \{0, 1\}^* \rightarrow \{0, 1\}^\ell$. (Gen, H) is said to be *collision resistant* if for all PPT adversary Collin, there exists a negligible negl such that $\forall n$,

$$\Pr_{k \leftarrow \text{Gen}(1^n)} [(x, x') \leftarrow \text{Collin}(k) : (H^k(x) = H^k(x')) \wedge (x \neq x')] \leq \text{negl}(n)$$

Commitment Scheme: A *commitment scheme* is a pair of algorithms $(\text{Commit}, \text{Receive})$ with syntax: for $m \in \{0, 1\}^\ell$, $\text{Commit}(1^n, m; r) = \text{com}$ (commits to message m using randomness r) and $\text{Receive}(\text{com}, m, r) = 0/1$, such that:

- **Correctness:** For all $n \in \mathbb{N}$, and $m \in \{0, 1\}^\ell$: $\Pr_r[\text{Receive}(\text{Commit}(1^n, m; r), m, r) = 1] = 1$.
- **Computational Hiding:** For any $m_0 \neq m_1 \in \{0, 1\}^\ell$, $\text{Commit}(1^n, m_0; r_0) \approx_C \text{Commit}(1^n, m_1; r_1)$.
- **Perfect Binding:** For any $\text{com} \in \{0, 1\}^*$, there **do not exist** two openings r_1, r_2 and $m_1 \neq m_2$ such that $\text{Receive}(\text{com}, m_1, r_1) = 1 = \text{Receive}(\text{com}, m_2, r_2)$.

If not perfect binding, a commitment scheme could also just satisfy computational binding:

Computational Binding: A commitment scheme $(\text{Commit}, \text{Receive})$ is said to be computationally binding if for all PPT adversary $\mathcal{A}$, there exists a negligible function negl such that $\Pr[(m_0, m_1, r_0, r_1) \leftarrow \mathcal{A}(1^n) : (m_0 \neq m_1) \wedge \text{Commit}(m_0, r_0) = \text{com} = \text{Commit}(m_1, r_1)] \leq \text{negl}(n)$.

Honest Verifier Zero Knowledge Proof (HVZKP): An interactive protocol $\Pi = (P, V)$ for a language L is an *honest verifier zero-knowledge proof system* if the following hold:

- **Completeness:** for every $x \in L$, $\Pr[1 \leftarrow \langle P, V \rangle(x)] = 1$ (i.e., verifier accepts for honest x).
- **Soundness:** for every $x \notin L$, $\exists \text{negl}$ such that for every P^* , $\Pr[1 \leftarrow \langle P^*, V \rangle(x)] \leq \text{negl}(n)$.
- **Honest Verifier Perfect/Computational ZK:** Π is *honest-verifier perfect ZK* if $\exists$ PPT, simulator Sim such that for all Spector and $\forall x \in L$

$$|\Pr[\text{Spector}(\text{View}_V(\langle P, V \rangle(x))) = 1] - \Pr[\text{Spector}(\text{Sim}(x)) = 1]| = 0$$

Computational HVZK is the same with the change that: Spector is PPT and the above difference in probability is $\leq \text{negl}(n)$.

Malicious Verifier Perfect/Computational ZK: An interactive protocol $\Pi = (P, V)$ for a language L is *malicious verifier perfect zero-knowledge proof system* if it satisfies completeness and soundness as above and the following holds: For every PPT V^* , there exists a (expected) PPT simulator Sim^{V^*} (Sim gets oracle access to V^*) such that for all Spector and $\forall x \in L$

$$|\Pr[\text{Spector}(\text{View}_{V^*}(\langle P, V^* \rangle(x))) = 1] - \Pr[\text{Spector}(\text{Sim}^{V^*}(x)) = 1]| = 0$$

Malicious Verifier Computational ZK is the same with the change that: Spector is PPT and the above difference in probability is $\leq \text{negl}(n)$.

Sigma Protocol: A *sigma protocol* $\Pi = (P, V)$ for a language $L \in \text{NP}$ is a specific interactive protocol with syntax: P on input (x, w) (where $(x, w) \in \mathcal{R}_L$), and V on input x have the following three rounds of communication: P sends a commitment com to V ; V reverts with a challenge c and P finally responds with an r . V finally accepts (outputs 1) or rejects (outputs 0). Π satisfies completeness and HVZK as in the above definitions and additionally satisfies the following special soundness: $\exists$ an efficient witness extractor Ext such that for $c \neq c'$ and two accepting transcripts $(\text{com}, c, r), (\text{com}, c', r')$, $\text{Ext}(x, (\text{com}, c, r), (\text{com}, c', r')) = w$ such that $(x, w) \in \mathcal{R}_L$.

Homomorphic Encryption for $\mathcal{F}$: $\Pi = (\text{Gen}, \text{Enc}, \text{Dec}, \text{Eval})$ with message space $\mathcal{M}$ is an IND-CPA secure PKE $(\text{Gen}, \text{Enc}, \text{Dec})$ equipped with an additional efficient Eval algorithm with syntax: on input ciphertext c corresponding to a message m under some public key pk and a function $f \in \mathcal{F}$, it outputs an evaluated $c' \leftarrow \text{Eval}(pk, c, f)$ such that the following additional properties hold:

- **Correctness of evaluation:** for $(pk, sk) \leftarrow \text{Gen}(1^n)$, $\forall m \in \mathcal{M}$ and $\forall f \in \mathcal{F}$,

$$\Pr[\text{Dec}(sk, \text{Eval}(pk, \text{Enc}(pk, m), f)) = f(m)] = 1$$

- **Compactness:** Size (bit-length) of the evaluated ciphertexts and the runtime of the decryption is independent of $|f|$.

When $\mathcal{F}$ consists of all functions computable by poly-sized circuits, Π is called an *FHE*.