# Zero-Knowledge Proofs (Lecture 5–8) — Summary

**Interactive Proofs.** A language $L \subseteq \{0,1\}^*$ has an *interactive proof* if there exists a (possibly unbounded) prover $P$ and a PPT verifier $V$ such that the protocol output $\langle P, V\rangle(x) \in \{0,1\}$ satisfies: *(Completeness)* $\forall x \in L : \Pr[\langle P, V\rangle(x) = 1] \geq 2/3$; *(Soundness)* $\forall x \notin L$ and any (possibly malicious) $P^*$: $\Pr[\langle P^*, V\rangle(x) = 1] \leq 1/3$. These constants can be made negligible by repetition. Interactive proofs can be strictly more powerful than one-shot NP proofs (e.g., IP = PSPACE) and can yield succinctness and zero-knowledge.

**Zero-Knowledge (ZK).** An interactive proof $\langle P, V\rangle$ for $L$ is (computational) ZK if for every (possibly malicious) $V^*$ there exists a PPT simulator $\mathsf{Sim}_{V^*}$ such that the verifier's *view* in a real interaction is computationally indistinguishable from the simulator's output:

$$\mathsf{View}[\langle P, V^*\rangle(x)] \;\approx_c\; \mathsf{Sim}_{V^*}(x)\,, \quad \forall x \in L.$$

Intuitively, the verifier "learns nothing" beyond the truth of $x \in L$.

**Example: Hamiltonicity (Blum).** For $L = \{G : G \text{ has a Hamiltonian cycle}\}$, the prover commits to a random isomorphic copy $\Pi(G)$ and the adjacency matrix, then the verifier challenges $c \in \{0,1\}$: if $c = 0$, open the isomorphism; if $c = 1$, open commitments revealing a Hamiltonian cycle in $\Pi(G)$. Assuming a perfectly binding, computationally hiding commitment, this is a ZK proof with perfect completeness, soundness error $\leq \frac{1}{2}$, and computational ZK; error becomes negligible by repetition.

**Proofs of Knowledge (PoK).** For NP languages with verifier $M$, a protocol is a PoK with knowledge error $\varepsilon$ if there exists an efficient *extractor* $\mathsf{Ext}$ such that for any $P^*$,

$$\Pr\left[M(x,w) = 1 :\; w \leftarrow \mathsf{Ext}^{P^*}(x)\right] \;\geq\; \Pr[\langle P^*, V\rangle(x) = 1] - \varepsilon\,.$$

Extractors may *rewind* $P^*$ (e.g., in Hamiltonicity, two accepting transcripts for the same commitment but distinct challenges yield the witness).

**Sigma Protocols.** Three-round, public-coin protocols $(u, c, z)$ with deterministic verification $\mathsf{verif}(x, u, c, z)$ that satisfy: *(Perfect completeness); (Special soundness):* from two accepting transcripts $(u, c, z)$ and $(u, c', z')$ with $c \neq c'$ one can extract a witness; *(Honest-Verifier ZK):* there is a simulator that on input $(x, c)$ outputs $(u, c, z)$ indistinguishable from an honest execution. Any $\Sigma$-protocol has soundness error at most $1/|\mathcal{C}|$, and both sequential and parallel compositions preserve these properties (for HVZK).

**Non-Interactive ZK (NIZK) & Fiat–Shamir.** Turning interaction into a single message $\pi$ is possible in the Random Oracle Model (ROM) via Fiat–Shamir: set the challenge as $c = H(x, u)$ and send $\pi = (u, z)$; the verifier recomputes $c$ and checks. In ROM, ZK follows by programming $H$ as to make it match the challenge of $\langle P, V\rangle$; PoK follows via rewinding at the RO query and reprogramming it to obtain two challenges; completeness is immediate. For NIZKs, completeness and soundness errors are required to be *negligible* since there is no repetition to amplify. In practice, one instantiates $H$ with a cryptographic hash (a heuristic).

**Key Takeaways.** (i) Interaction enables proofs beyond NP, succinct verification, and zero-knowledge. (ii) ZK is defined via indistinguishability from simulation. (iii) PoK formalizes "knowing a witness" via extraction under rewinding. (iv) $\Sigma$-protocols offer a clean template with special soundness and HVZK, and compose well. (v) Fiat–Shamir yields practica NIZKs from $\Sigma$-protocols in ROM.