We will need a commitment scheme (see HW2). A (non-interactive) commitment scheme consists of two main algorithms (Commit, Verify)

cryptographic analog of a sealed "envelope"

- $\mathrm{Commit}(m; r) \to c$: Takes a message m and randomness r and outputs a commitment c
- $\mathrm{Verify}(m, c, r) \to b$: Checks if c is a valid opening to m (with respect to randomness r)

[The commitment scheme might also take public parameters (see HW2), but for simplicity, we omit them / leave them implicit]

Requirements:

- Correctness: for all messages m:

sampled uniformly

$$\Pr[c \leftarrow \mathrm{Commit}(m; r) : \mathrm{Verify}(m, c, r) = 1] = 1$$

- Hiding: for all efficient adversaries A, if $(m_0, m_1) \leftarrow A$

randomness is uniformly random in both distributions

$$\{c \leftarrow \mathrm{Commit}(m_0; r) : c\} \overset{c}{\approx} \{c \leftarrow \mathrm{Commit}(m_1; r) : c\}$$

- Binding: for all efficient adversaries A, if

$$\Pr[(m_0, m_1, c, r_0, r_1) \leftarrow A : m_0 \neq m_1 \text{ and } \mathrm{Verify}(m_0, c, r_0) = 1 = \mathrm{Verify}(m_1, c, r_1)] = \text{negl.}$$

↳ We will require perfect binding [for every commitment c, there is only 1 possible m to which the prover can open c]

A ZK protocol for graph 3-coloring:

contains n nodes, m edges

prover (G)

- let $K_i \in \{0,1,2\}$ be a 3-coloring of G
- choose random permutation $\pi \xleftarrow{R} \mathrm{Perm}[\{0,1,2\}]$
- for $i \in [n]$: $c_i \leftarrow \mathrm{Commit}(K_i; r_i)$ for random r_i

verifier (G)

prover → verifier: $c_1, \ldots, c_n$

verifier: $(i, j) \xleftarrow{R} E$

verifier → prover: i, j

prover → verifier: $(K_i, r_i), (K_j, r_j)$

↳ accept if $K_i \neq K_j$ and $K_i, K_j \in \{0,1,2\}$

$\mathrm{Verify}(K_i, c_i, r_i) = 1 = \mathrm{Verify}(K_j, c_j, r_j)$

reject otherwise

Intuitively: Prover commits to a coloring of the graph

Verifier challenges prover to reveal coloring of a single edge

Prover reveals the coloring on the chosen edge and opens the entries in the commitment

Completeness: By inspection [if coloring is valid, prover can always answer the challenge correctly]

Soundness: Suppose G is not 3-colorable. Let $K_1, \ldots, K_n$ be the coloring the prover committed to. If the commitment scheme is perfectly binding, $c_1, \ldots, c_n$ uniquely determine $K_1, \ldots, K_n$. Since G is not 3-colorable, there is an edge $(i,j) \in E$ where $K_i = K_j$ or $i \notin \{0,1,2\}$ or $j \notin \{0,1,2\}$. [Otherwise, G is 3-colorable with coloring $K_1, \ldots, K_n$.] Since the verifier chooses an edge to check at random, the verifier will choose (i,j) with probability $1/|E|$. Thus, if G is not 3-colorable,

$$\Pr[\text{verifier rejects}] \geq \frac{1}{|E|}$$

Thus, this protocol provides soundness $1 - \frac{1}{|E|}$. We can repeat this protocol $O(|E|^2)$ times sequentially to reduce soundness error to

$$\Pr[\text{verifier accepts proof of false statement}] \leq \left(1 - \frac{1}{|E|}\right)^{|E|^2} \leq e^{-|E|} = e^{-m} \quad [\text{since } 1 + x \leq e^x]$$

<u>Zero Knowledge</u>: We need to construct a simulator that outputs a valid transcript given only the graph G as input.

Let V^* be a (possibly malicious) verifier. Construct simulator S as follows:

1. Choose $K_i \leftarrow \{0,1,2\}$ for all $i \in [n]$.
   Let $c_i \leftarrow \text{Commit}(K_i\,;\,r_i)$
   Give $(c_1, \ldots, c_n)$ to V^*.
2. V^* outputs an edge $(i,j) \in E$
3. If $K_i \neq K_j$, then S outputs (K_i, K_j, r_i, r_j).
   Otherwise, restart and try again (if fails λ times, then abort)

Simulator does not know coloring so it commits to a random one

Simulator succeeds with probability $2/3$ (over choice of $K_1, \ldots, K_n$). Thus, simulator produces a valid transcript with prob. $1 - \frac{1}{3^\lambda} = 1 - \text{negl}(\lambda)$ after λ attempts. It suffices to show that simulated transcript is indistinguishable from a real transcript.

- <u>Real scheme</u>: prover opens K_i, K_j where $K_i, K_j \xleftarrow{R} \{0,1,2\}$ [since prover randomly permutes the colors]
- <u>Simulation</u>: K_i and K_j sampled uniformly from $\{0,1,2\}$ and conditioned on $K_i \neq K_j$, distributions are identical

In addition, (i,j) output by V^* in the simulation is distributed correctly since commitment scheme is computationally-hiding (e.g. V^* behaves essentially the same given commitments to a random coloring as it does given commitment to a valid coloring

If we repeat this protocol (for soundness amplification), simulator simulate one transcript at a time

<u>Summary</u>: Every language in NP has a zero-knowledge proof

In many cases, we want a stronger property: the prover actually "knows" <u>why</u> a statement is true (e.g., it knows a "witness")

For instance, consider the following language:

$$\mathcal{L} = \{ h \in \underbrace{\mathbb{G}}_{\text{group of order } p} \mid \exists x \in \mathbb{Z}_p : h = \underbrace{g}_{\text{generator of } \mathbb{G}}{}^{x} \} = \mathbb{G}$$

Note: this definition of $\mathcal{L}$ implicitly defines an NP relation R:

$$R(h, x) = 1 \iff h = g^x \in \mathbb{G}$$

In this case, all statements in $\mathbb{G}$ are true (i.e., contained in $\mathcal{L}$), but we can still consider a notion of proving <u>knowledge</u> of the discrete log of an element $h \in \mathbb{G}$ — conceptually <u>stronger</u> property than proof of membership

<u>Philosophical question</u>: What does it mean to "know" something?

If a prover is able to convince an honest verifier that it "knows" something, then it should be possible to <u>extract</u> that quantity from the prover.

<u>Definition.</u> An interactive proof system (P, V) is a proof of knowledge for an <u>NP relation R</u> if there exists an efficient extractor $\mathcal{E}$ such that for any x and any prover P^*

proof of knowledge is parameterized by a specific <u>relation</u> R (as opposed to the language $\mathcal{L}$)

$$\Pr\left[w \leftarrow \mathcal{E}^{P^*}(x) : R(x, w) = 1\right] \geq \Pr\left[\langle P^*, V\rangle(x) = 1\right] - \underbrace{\varepsilon}_{\text{knowledge error}}$$

more generally, could be polynomially smaller