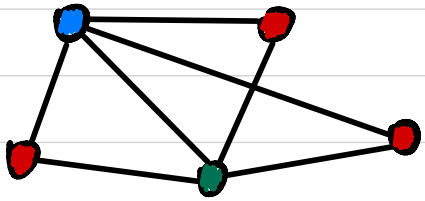
CSCI 1515 Applied Cryptography

This Lecture:

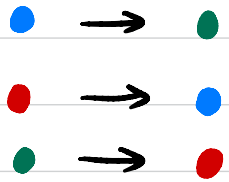
- Zero-Knowledge Proofs for All NP (Continued)
- Succinct Non-Interactive Arguments (SNARGs)
- SNARGs from PCP

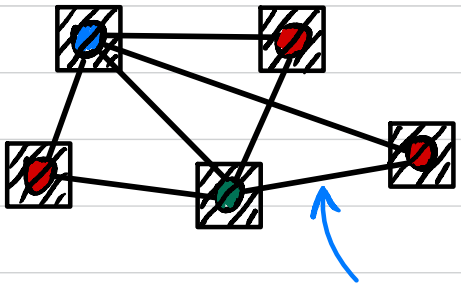
Zero-Knowledge Proof for Graph 3-Coloring (All NP)



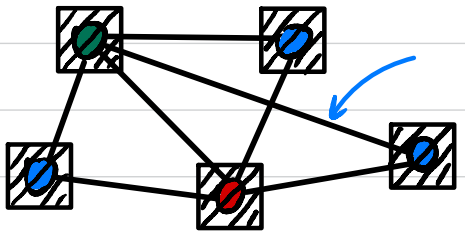
NP language $L = \{ G : G \text{ has 3-coloring} \}$

NP relation $R_L = \{ (G, 3COL) \}$





New Coloring:



If $G \notin L$, $\Pr[P^* \text{ is caught}] \geq \frac{1}{|E|}$

How to amplify soundness? Repeat k times, $k = \lambda \cdot |E|$

$\Pr[P^* \text{ isn't caught}] \leq \left(1 - \frac{1}{|E|}\right)^k = \left(1 - \frac{1}{|E|}\right)^{|E| \cdot \lambda} \simeq \left(\frac{1}{e}\right)^\lambda$ $\quad \left(1 - \frac{1}{n}\right)^n \simeq \frac{1}{e}$

$\Pr[P^* \text{ is caught at least once}] \geq 1 - \left(\frac{1}{e}\right)^\lambda$

Commitment Scheme

Sender		Receiver
$m \in \{0,1\}$		
Commit: $r \xleftarrow{\$} \{0,1\}^\lambda$, $c := Com(m;r)$	$\xrightarrow{c}$	
Open:	$\xrightarrow{(m,r)}$	Verify: $c = Com(m;r)$

- **Hiding:** $Com(0;r) \simeq Com(1;s)$
- **Binding:** Hard to find r, s s.t. $Com(0;r) = Com(1;s)$

Commitment Scheme

Hiding: $\text{Com}(0; r) \simeq \text{Com}(1; s)$

Binding:

Hard to find r, s s.t. $\text{Com}(0; r) = \text{Com}(1; s)$

Example 1: Hash-based commitment

$r \xleftarrow{\$} \{0,1\}^\lambda$

$\text{Com}(m; r) := H(r \| m)$

↑ Random Oracle

Hiding: randomness of r

Binding: collision resistance of H

Example 2: Pedersen Commitment

Cyclic group G of order q with generator g. $h \xleftarrow{\$} G$

↑ can be generated by Receiver

$h = g^x$. x hidden to Sender

$r \xleftarrow{\$} \mathbb{Z}_q$

$\text{Com}(m; r) = g^m \cdot h^r$

Hiding: OTP of h^r

Binding: DLOG of h

$\text{Com}(0; r) = \text{Com}(1; s)$

$h^r = g^1 \cdot h^s$

$\Downarrow$

$h^{r-s} = g$

$\Downarrow$

$h = g^{(r-s)^{-1}}$

Why are the schemes hiding & binding?

Zero-Knowledge Proof for Graph 3-Coloring

Input: $G = (V, E)$

Witness: $\phi: V \to \{0,1,2\}$

Prover

Randomly sample $\pi: \{0,1,2\} \to \{0,1,2\}$

$\forall v \in V,\ r_v \xleftarrow{R} \{0,1\}^\lambda,\ C_v := \mathsf{Com}(\pi(\phi(v)); r_v)$

$\xrightarrow{\{C_v\}_{v \in V}}$

Verifier

Randomly pick an edge $(u,v) \in E$

$\xleftarrow{(u,v)}$

Open commitments C_u & C_v

$\xrightarrow{\substack{\alpha = \pi(\phi(u)),\ r_u \\ \beta = \pi(\phi(v)),\ r_v}}$

Verify: $C_u = \mathsf{Com}(\alpha; r_u)$

$C_v = \mathsf{Com}(\beta; r_v)$

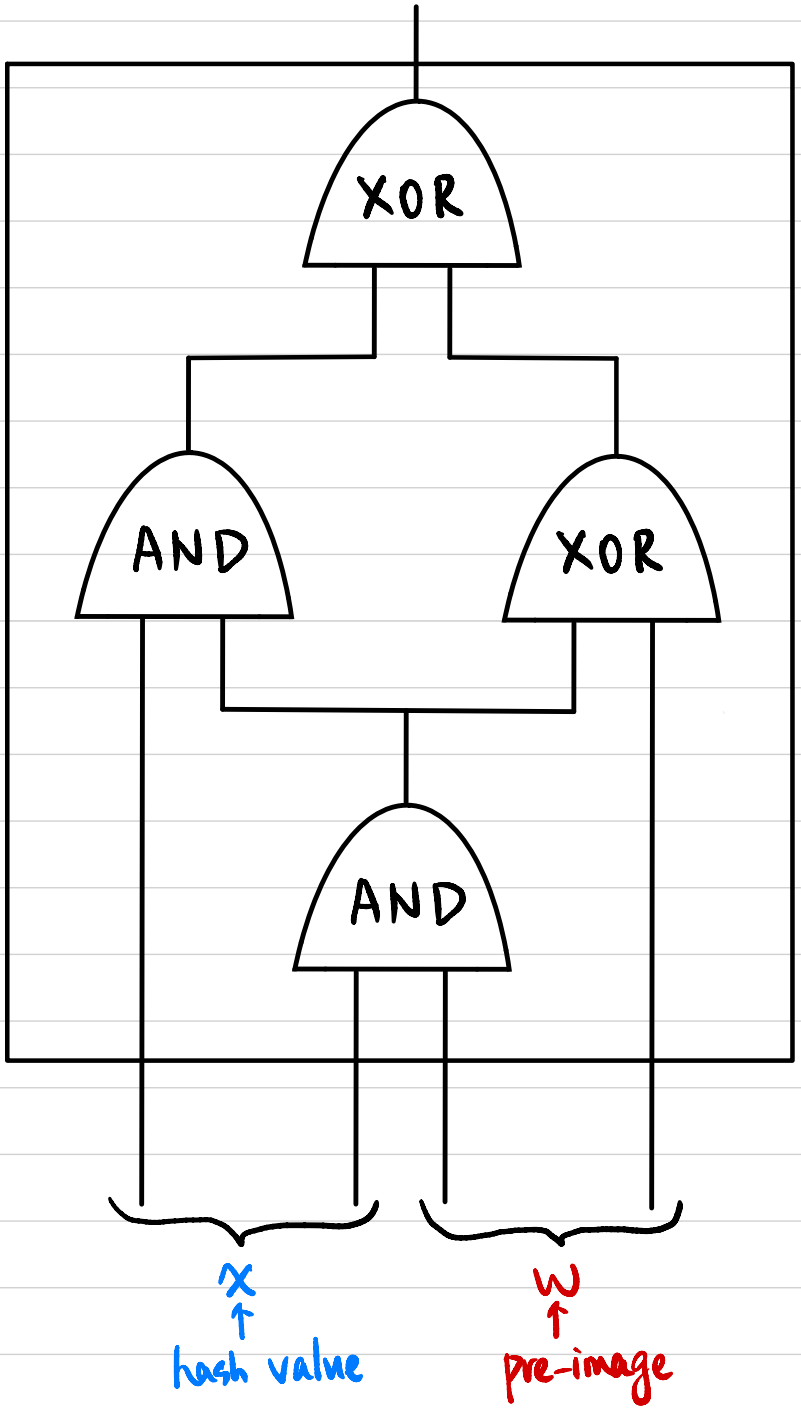
$\alpha, \beta \in \{0,1,2\},\ \alpha \neq \beta$

Completeness ?

Soundness? Binding of Commitment Scheme

Zero-Knowledge? Hiding of Commitment Scheme

Circuit Satisfiability (NP Complete)



NP language $L_C = \{x \in \{0,1\}^n : \exists w \in \{0,1\}^m \text{ s.t. } C(x,w) = 1\}$

NP relation $R_{L_C} = \{(x, w) : C(x, w) = 1\}$

(public) Statement

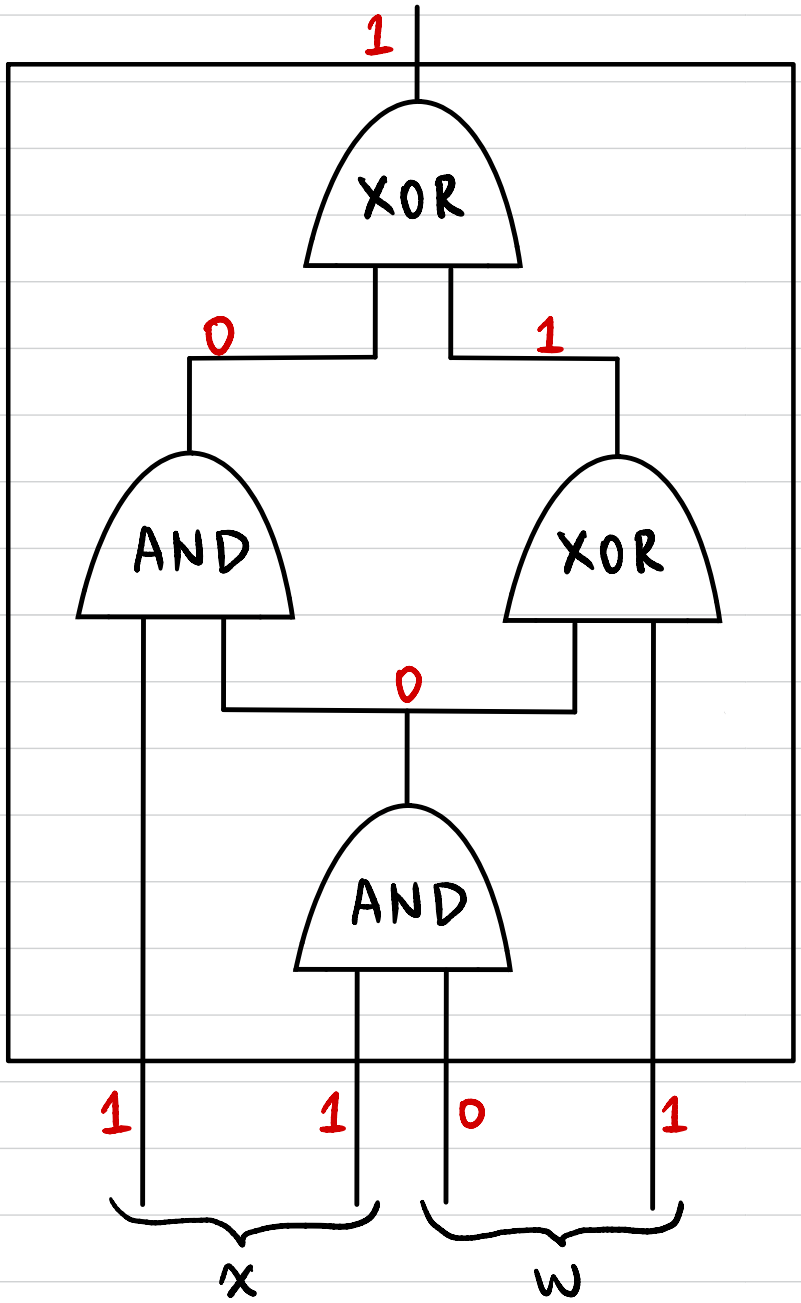
(secret) Witness

Example: pre-image of hash function

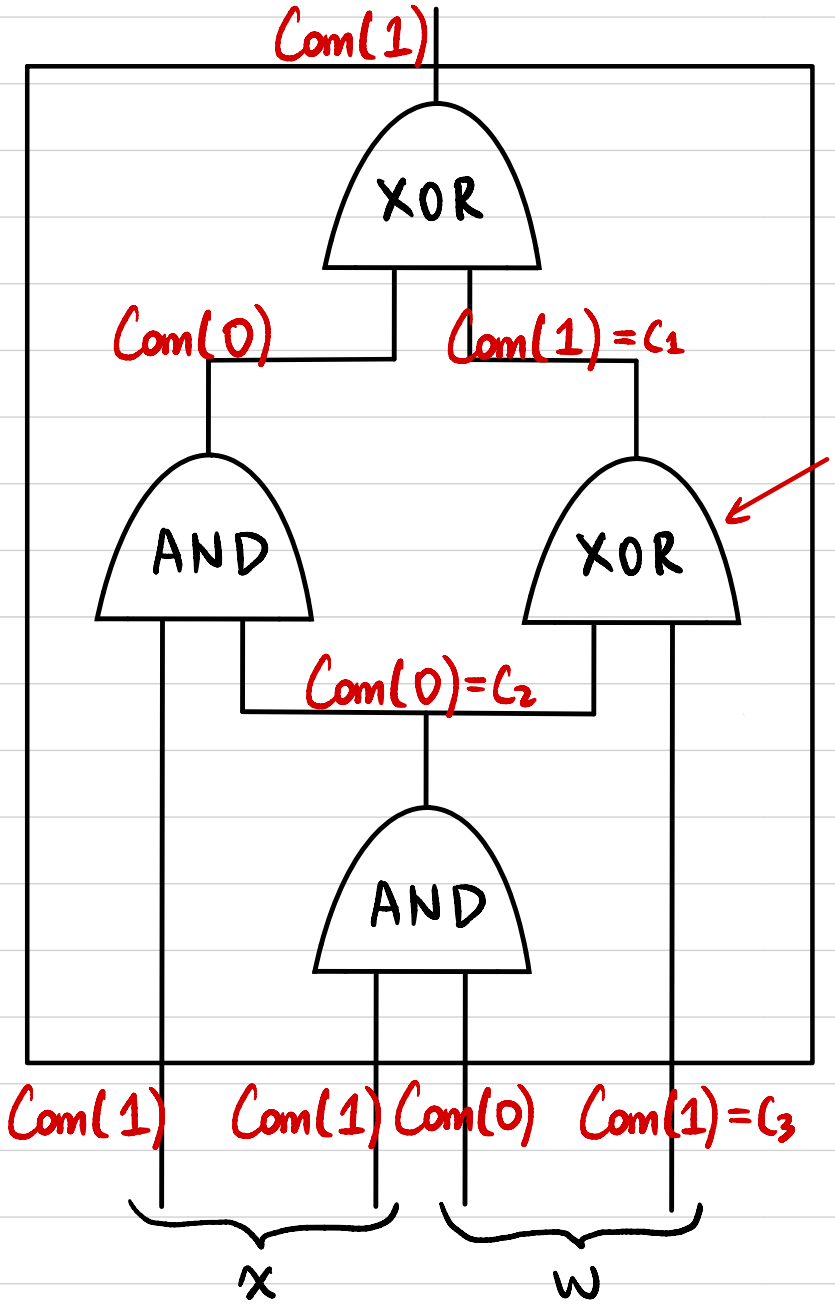
$C(x, w) = H(w) - x + 1$

$H(w) = x$

ZKP for Circuit Satisfiability



ZKP for Circuit Satisfiability



$$\begin{pmatrix} C_1 = Com(0) \\ C_2 = Com(0) \\ C_3 = Com(0) \end{pmatrix}$$

OR

$$\begin{pmatrix} C_1 = Com(1) \\ C_2 = Com(0) \\ C_3 = Com(1) \end{pmatrix}$$

OR

$$\begin{pmatrix} C_1 = Com(1) \\ C_2 = Com(1) \\ C_3 = Com(0) \end{pmatrix}$$

OR

$$\begin{pmatrix} C_1 = Com(0) \\ C_2 = Com(1) \\ C_3 = Com(1) \end{pmatrix}$$

Proof Systems for Circuit Satisfiability

NP relation $R_{\mathcal{C}} = \{(x, w) : C(x, w) = 1\}$

	NP	Σ-Protocol	(Fiat-Shamir) NIZK
	$P(x, w) \xrightarrow{w} V(x)$	$P(x, w) \rightleftarrows V(x)$	$P(x, w) \xrightarrow{\pi} V(x)$
Zero-Knowledge	NO	YES	YES
Non-Interactive	YES	NO	YES
Communication	$O(\|w\|)$	$O(\|C\| \cdot \lambda)$	$O(\|C\| \cdot \lambda)$
V's computation	$O(\|C\|)$	$O(\|C\|)$	$O(\|C\|)$

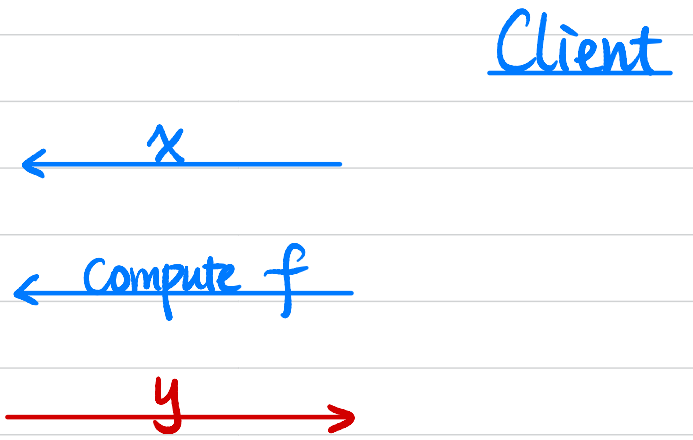
Can we have communication complexity & verifier's computational complexity sublinear in $|C|$ & $|w|$?

Succinct Non-Interactive Argument

- **SNARG:** Succinct Non-Interactive Argument
- **SNARK:** Succinct Non-Interactive Argument of Knowledge
- **zk-SNARG / zk-SNARK:** SNARG / SNARK + Zero-Knowledge

- **Succint:** $|\pi| = \text{poly}(\lambda, \log |C|)$
  Verifier runtime $\text{poly}(\lambda, |x|, \log |C|)$
- **Argument:** In Soundness / Proof of knowledge: $\forall$ PPT P^*

Verifiable Computation



Anonymous Transactions on Blockchain

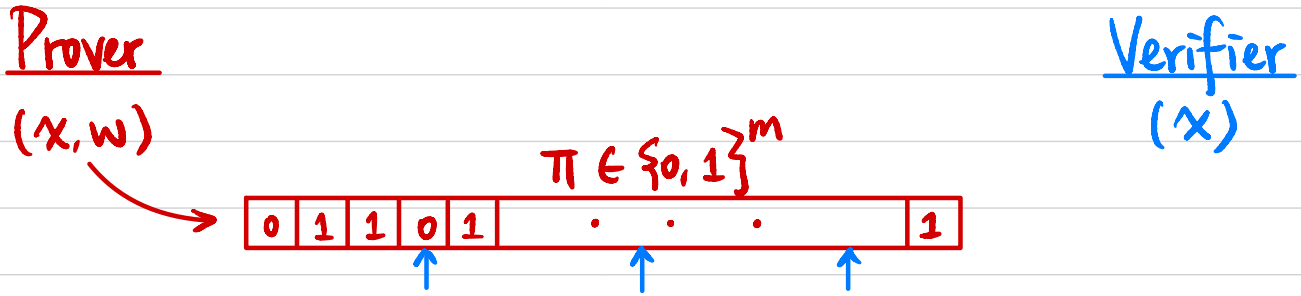
Transaction: $\boxed{vk_A, vk_B, 2BTC}$, $\sigma = Sign_{sk_A}(\cdot)$

Anonymous Transaction:

$$Com\left(\boxed{vk_A, vk_B, 2BTC}, \sigma = Sign_{sk_A}(\cdot)\right)$$

NIZk: valid transaction

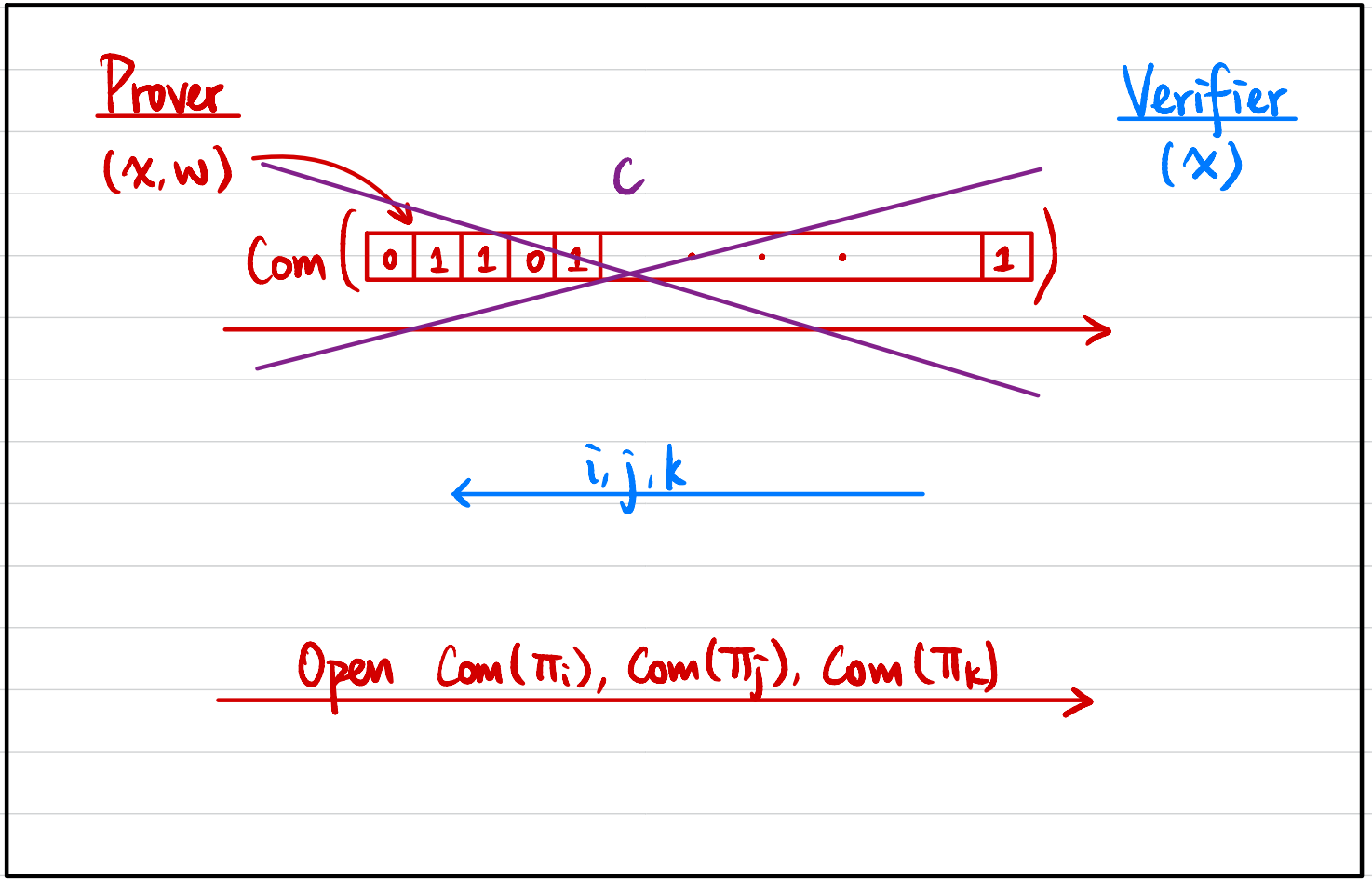
Probabilistically Checkable Proof (PCP)



PCP Theorem (Informal):

Every NP language has a PCP where the verifier reads only a constant number of bits of the proof.

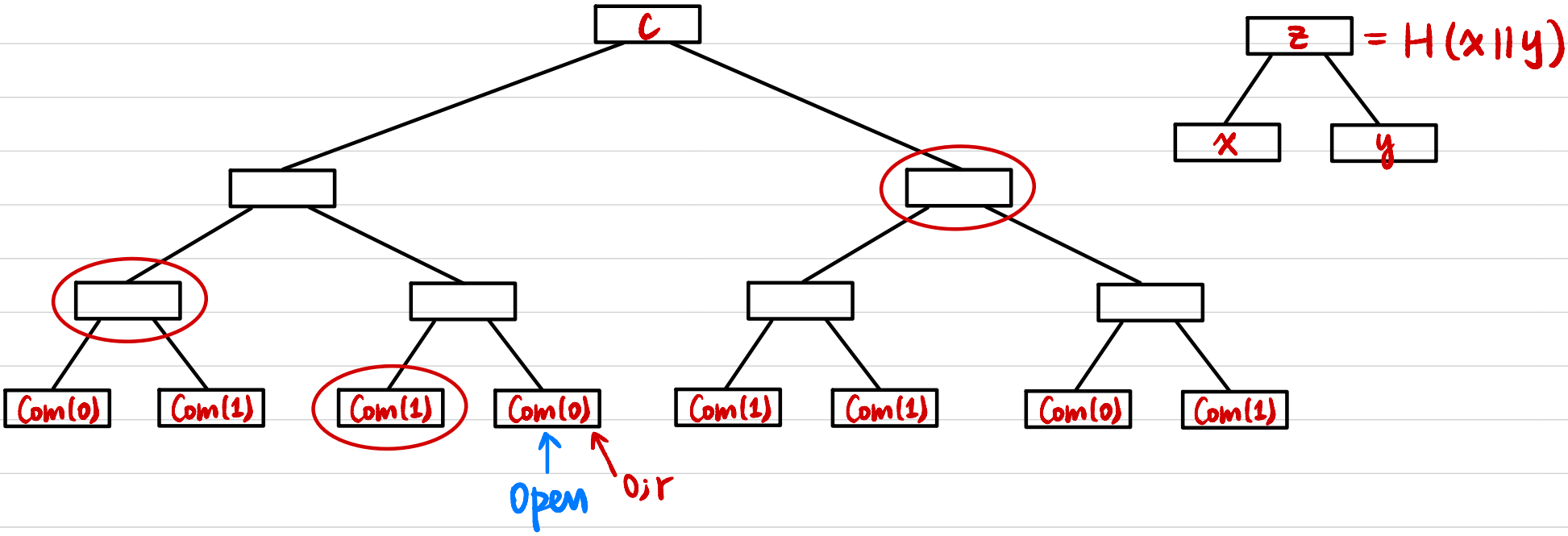
First Attempt



Is it succint?

Is it zk?

Merkle Tree



How to open commitment ?

Why hiding & binding ?