Correlation-intractability hash function: Let $R(x,y)$ be a binary relation.

We say a hash function $H : \mathcal{X} \to \mathcal{Y}$ is correlation intractable for the relation $R : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ if no efficient adversary can find an $x \in \mathcal{X}$ such that $R(x, H(x)) = 1$.

→ Technically, the correlation-intractable hash function takes a public hash key hk and the input x.

Back to Fiat-Shamir...

prover		verifier
	$\xrightarrow{\ \sigma\ }$	
	$\xleftarrow{\ c\ }$	
	$\xrightarrow{\ z\ }$	

For a statement $x \notin \mathcal{L}$, define "bad challenge" relation R_x as follows:

$$R_x(\sigma, c) = 1 \text{ if } \exists z : \mathrm{Verify}(x, (\sigma, c, z)) = 1$$

If H is correlation-intractable for R_x, then we can set $c \leftarrow H(hk, \sigma)$. Here, the hash key hk is part of the public parameters.

Soundness analysis: Suppose adversary outputs a proof $\pi = (\sigma, z)$ for a statement x.
By correlation-intractability, $R_x(\sigma, H(\sigma)) = 0$.
This means there does not exist z such that verifier accepts $(\sigma, H(\sigma), z)$.
Thus, the verifier is guaranteed to reject $\Rightarrow$ soundness follows.

Zero-knowledge: Candidate simulation strategy: use HVZK simulator of the underlying protocol
Problem: simulated transcript outputs (σ, c, z) where $c \xleftarrow{R} \{0,1\}^n$ but in the real scheme, $c = H(hk, \sigma)$
Solution: Sample a shift $\rho \xleftarrow{R} \{0,1\}^n$ and include with the crs $= (hk, \rho)$
Define the challenge to be $c \leftarrow H(hk, \sigma) \oplus \rho$

To simulate: Run HVZK simulator for underlying protocol to get (σ, c, z)
Sample hash key hk and set $\rho \leftarrow H(hk, \sigma) \oplus c$

Output crs $= (hk, \rho)$ and proof $\pi = (\sigma, c, z)$

Since $c \xleftarrow{R} \{0,1\}^n$, ρ is properly distributed and the scheme is ZK

We say that the relation $R(x,\cdot)$ is <u>sparse</u> if

$$\Pr_{y \leftarrow \mathcal{Y}} [R(x,y) = 1] = \text{negl}(\lambda)$$

In the case of Blum's protocol for graph Hamiltonicity (with statistically-binding commitments), for every choice of prover's first message σ, there is a <u>single</u> bad challenge (from the challenge space $\{0,1\}^\lambda$).
↳ BadChallenge relation is sparse.

Easy to see that random oracle is correlation-intractable for sparse relations (by definition).

<u>Goal</u>: Construct correlation-intractable hash function from a <u>concrete</u> cryptographic assumption.

We will show it for the class of "search" relations:
for every x, there exists a unique y such that $R(x,y) = 1$
moreover, it should be <u>efficient</u> to find the unique y for a given x where $R(x,y) = 1$

<u>Observe</u>: bad challenge for Blum's protocol is a search relation
(though as presented, not efficiently searchable)

<u>Fix</u>: Efficiently-searchable given a trapdoor (after tweaking protocol)
(we will use an extractable commitment, which we have from GSW)
↳ Will need to <u>hide</u> search relation within public hash key
(otherwise, extraction trapdoor is compromised)

Correlation-intractability for function f: hard to find x where $H(hk, x) = f(x)$

Correlation-intractability <u>without</u> hiding for search relations:
$\text{Setup}(1^\lambda, f) \to hk = f$
$H(hk, x) \to f(x) \oplus 0^{k-1} \| 1$ (i.e., flip the last bit of f)
But hash key completely leaks the function f! We need a correlation intractable hash function where hk hides the function f.

<u>Solution</u>: Encrypt the function and homomorphically evaluate f

First construction: from circular-secure FHE (not quite LWE, but close)

For an input $x \in \{0,1\}^\ell$ define the universal circuit $U_x(f) \to f(x)$

we will set t to be the length of an FHE ciphertext

- U_x takes description of function $f: \{0,1\}^\ell \to \{0,1\}^t$ (of bounded size) and outputs $f(x)$

Define hash key to be pk for FHE scheme and ct as an encryption of an arbitrary function g (e.g., the all-zeroes function)

- $hk = (pk, ct)$

Hash function is then

$$H(hk, x) := \text{FHE.Eval}(pk, U_x, ct)$$

To show that this is correlation-intractable for any function f, we use a hybrid argument:

Hyb_0: real game

adversary	challenger
	$(pk, sk) \leftarrow \text{FHE.KeyGen}(1^\lambda)$
	$ct \leftarrow \text{FHE.Encrypt}(pk, g)$
$\xleftarrow{\quad hk = (pk, ct) \quad}$
$\downarrow$
x

adversary wins if $H(hk, x) = f(x)$

flip the last bit of FHE.Decrypt$(sk, f(x))$

Hyb_1: define the function $f'(x) := \text{FHE.Decrypt}(sk, f(x)) \oplus 0^{t-1} \| 1$

set the ciphertext $ct \leftarrow \text{FHE.Encrypt}(pk, f')$

Hyb_0 and Hyb_1 are computationally indistinguishable by circular security of FHE (since f' depends on sk)

In Hyb_1, there <u>does not exist</u> x where $H(hk, x) = f(x)$. Suppose otherwise:

$$f(x) = H(hk, x) = \underbrace{\text{FHE.Eval}(pk, U_x, ct)}$$

Correctness of FHE $\Rightarrow$ <u>encryption</u> of $U_x(f') = \text{FHE.Decrypt}(sk, f(x)) \oplus 0^{t-1} \| 1$

Suppose we apply FHE.Decrypt$(sk, \cdot)$ to both sides:

$$\boxed{\text{FHE.Decrypt}(sk, f(x))} = U_x(f') = \boxed{\text{FHE.Decrypt}(sk, f(x)) \oplus 0^{t-1} \| 1}$$

Contradiction!

In Hyb_1, correctness of FHE implies <u>statistical</u> correlation intractability

In real scheme, (pk, ct) are <u>independent</u> of f, so f is perfectly hidden