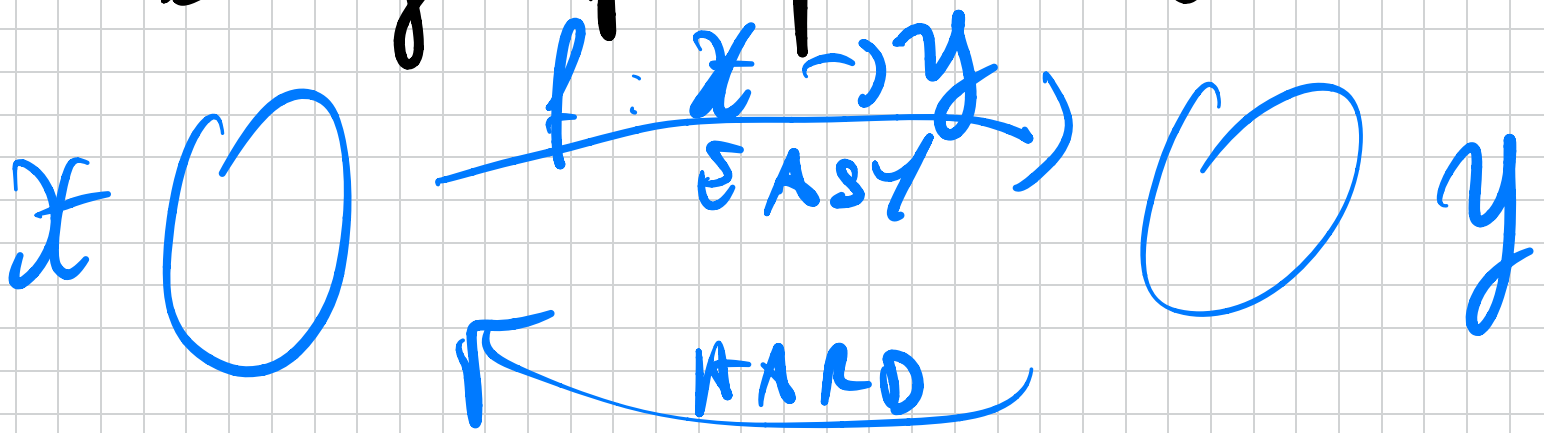
PSEUDORANDOMNESS

Our first step, towards efficient symmetric crypto. Moreover, pseudorandomness as well as what modern computers do. "simulate" real randomness.

We will see that ONE-WAY FUNCTIONS are enough for pseudorandomness.



Def (OWF). A function $f : \{0,1\}^n \to \{0,1\}^n$ is one-way, if $\forall$ PPT A:

$$\Pr_{x \leftarrow \{0,1\}^n}\left[f(x') = y : y = f(x),\ x' \leftarrow A(y) \right] \leq \text{negl}(n).$$

Example of $\text{negl}(n)$: e.g. 2^{-n}.

Exercise: Prove $2^{-n} = \text{negl}(n)$.

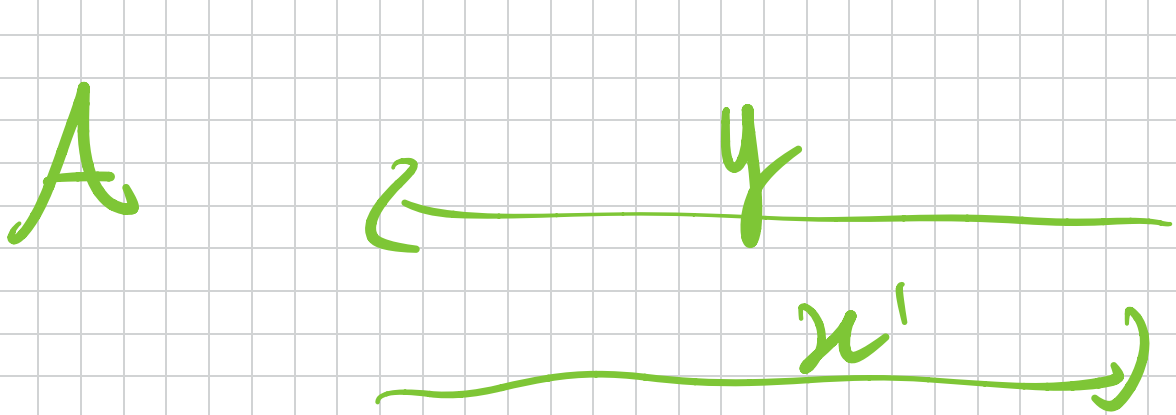
Exercise: $\varepsilon_1(n)\varepsilon_2(n) = \text{negl}(n)$, Then

$\varepsilon(n) = \varepsilon_1(n) + \varepsilon_2(n)$ also $\mathrm{negl}(n)$.

Exercise: $p(n) \cdot \varepsilon(n) = \mathrm{negl}(n)$

if $p(n) = \mathrm{poly}(n)$ and $\varepsilon(n) = \mathrm{negl}(n)$.

Alternative: We can think of this



$\mathcal{C}^{owf}$

$x \leftarrow \{0,1\}^n$

$y = f(x)$

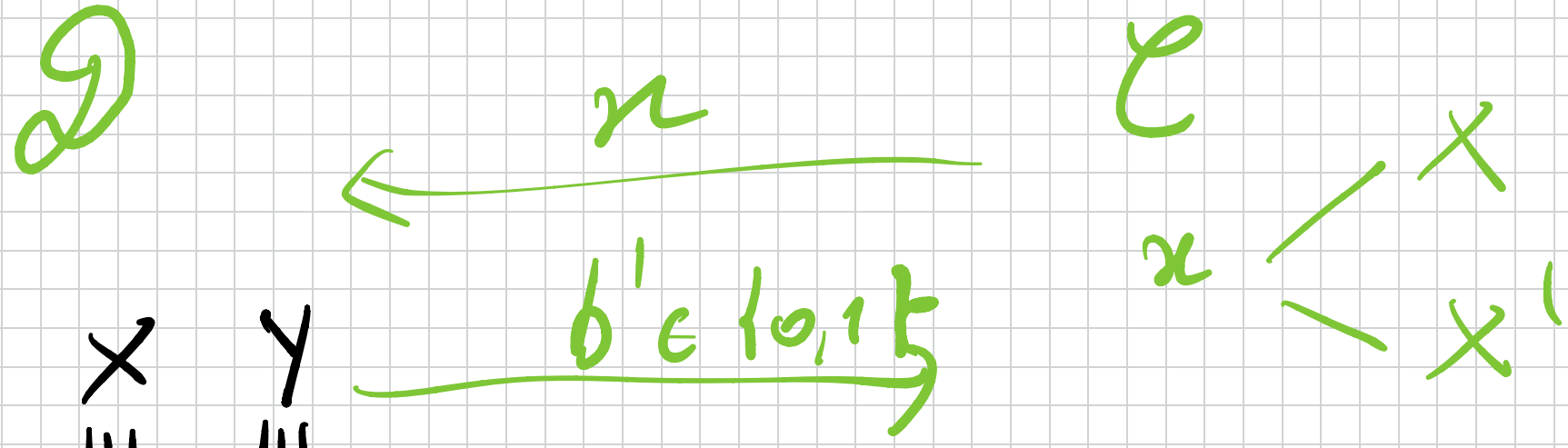
Win: $y = f(x')$

Let's define pseudorandomness: This means sequence of bits that are NOT random but they LOOK random. We capture this requirement using INDISTINGUISHABILITY (COMPUTATIONAL).

We have already seen stuff like this: STATISTICAL DISTANCE. Given X, X' RVs over some domain, $SD(X, X') \leq \varepsilon$ is equivalent to: $\forall D$:

$$\left| Pr[D(x) = 1 : x \leftarrow X] + \right.$$

$$- \Pr[\mathcal{D}(x)=1 : x \leftarrow X'] \Big| \leq \varepsilon .$$



Def. $\{X_n\}, \{Y_n\}$ are computationally indistinguishable ($X \approx_c Y$) : $\forall$ PPT $\mathcal{D}$

$$\left| \Pr[\mathcal{D}(z)=1 : z \leftarrow X_n] - \Pr[\mathcal{D}(z)=1 : z \leftarrow Y_n] \right| \leq \mathrm{negl}(n)$$

Exercise: If $X \overset{c}{\approx} Y$, and $Y \overset{c}{\approx} Z$ then $X \overset{c}{\approx} Z$.

With this, we can define PSEUDORANDOM GENERATORS.

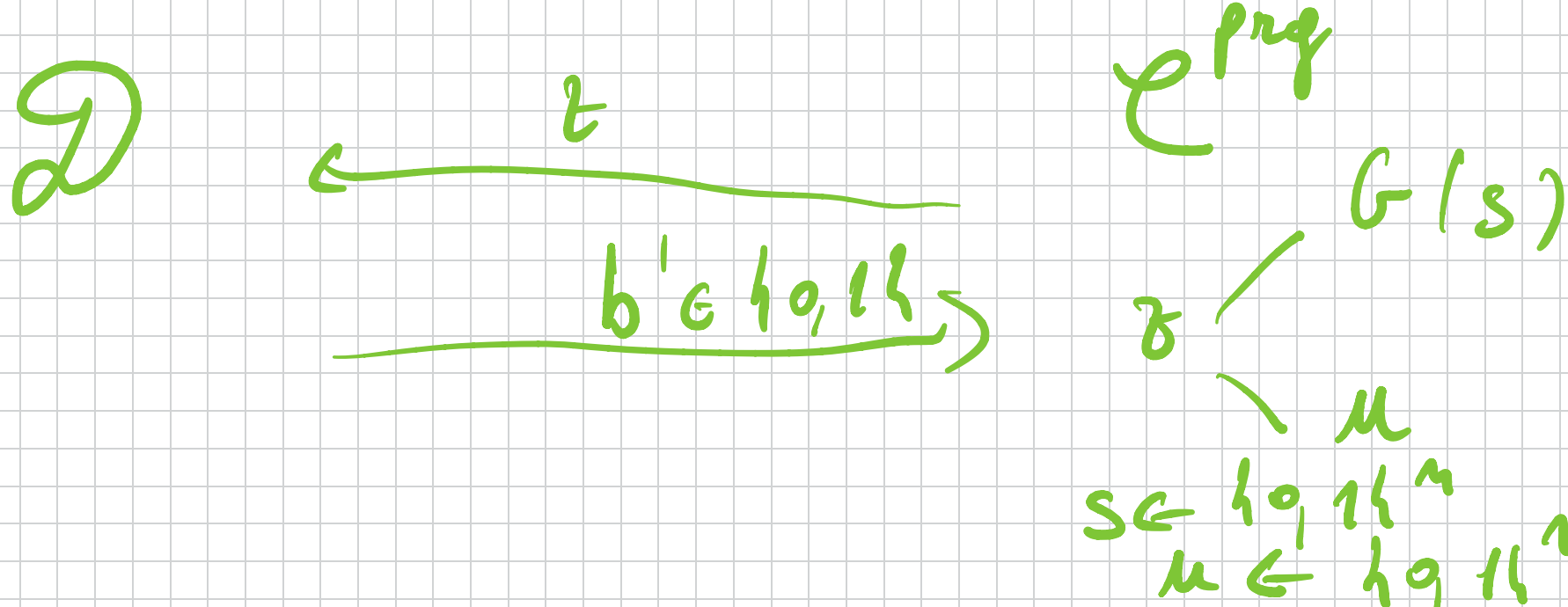
Def (PRG). A PRG $G : \{0,1\}^n \to \{0,1\}^{n+\ell}$ with $\ell \geq 1$ (the STRETCH) is

SECURE if:

$$G(U_n) \overset{\sim}{\approx}_c U_{n+\ell}$$

$U_n \equiv$ UNIFORM over $\{0,1\}^n$

$U_{n+\ell} \equiv$ " " $\{0,1\}^{n+\ell}$



(Looking ahead: A PRG is useful to do SKE beating Shannon!

$$Enc(k, m) = G(k) \oplus m = c$$

$$Dec(k, c) = G(k) \oplus c = m.$$

$|k| << |m|$!!!)

Before closing shop, let's understand how to build PRGs:

1) Use a randomness extractor to get a UNIFORM seed $s \in \{0,1\}^n$.

2) Define a simple PRG $G: \{0,1\}^n \to \{0,1\}^{n+1}$ with minimal stretch $\ell = 1$.

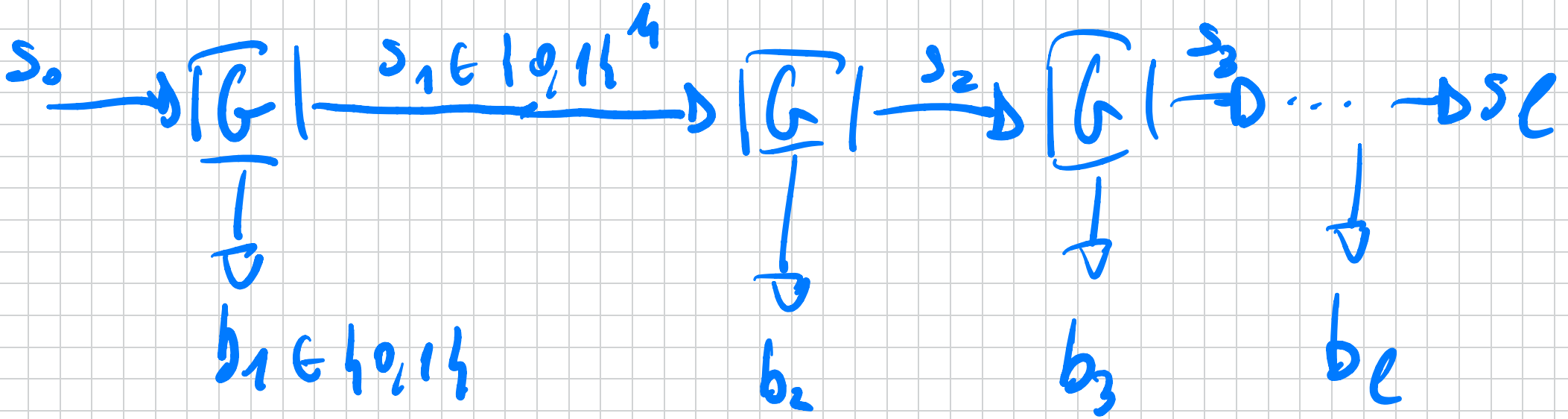
3) Use G to stretch any $\ell(n) = \text{poly}(n)$.

Theory vs practice:

- Randomness extraction as what we already studied. But in practice it is done using HASH FUNCTIONS.
- Theoretical G can be obtained from any OWF. Practical G is HEURISTIC.
- Stretch amplification is the same.
- (In practice the seed is REFRESHED periodically collecting new ENTROPY.)

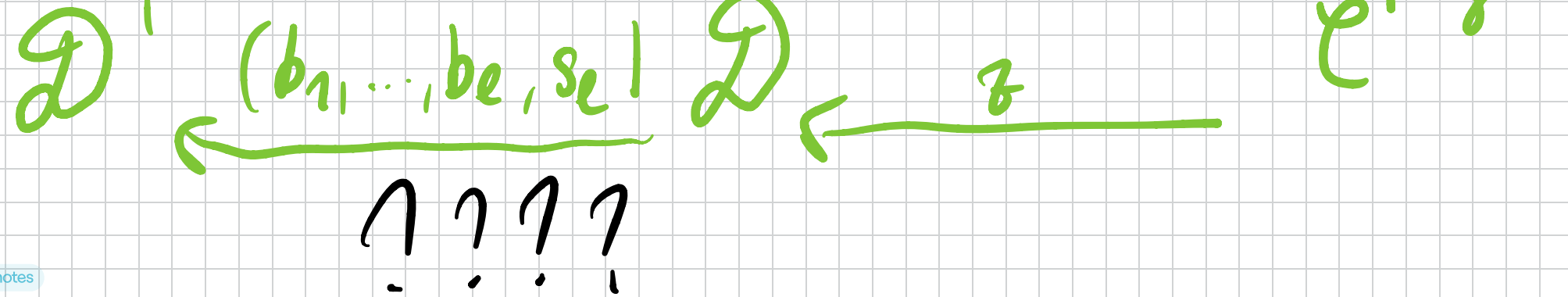
THM If there exists a PRG $G: \{0,1\}^n \to \{0,1\}^{n+1}$, then there exists a PRG

$G^\ell: \{0,1\}^n \to \{0,1\}^{n+\ell}$ for any $\ell(n) = \text{poly}(n)$

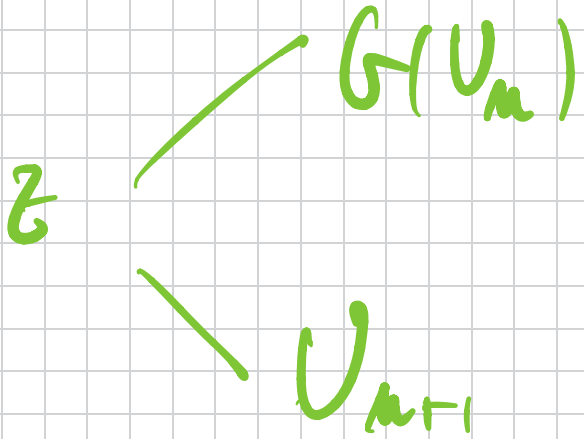


$$G^\ell(s_0) = (b_1, \ldots, b_\ell, s_\ell)$$

Proof. This is the idea: Assume G^ℓ not secure, $\exists$ PPT D' that can distinguish $G^\ell(U_n)$ from $U_{n+\ell}$ w.p. $\geq 1/p'(n)$ for some polynomial $p(\cdot)$. We want to build PPT D that can distinguish $G(U_n)$ from U_{n+1} w.p. $1/p(n)$.



Where do I put z ? ? ?



Hybrid argument:

$H_0(m) \equiv G^{\ell}(U_m)$

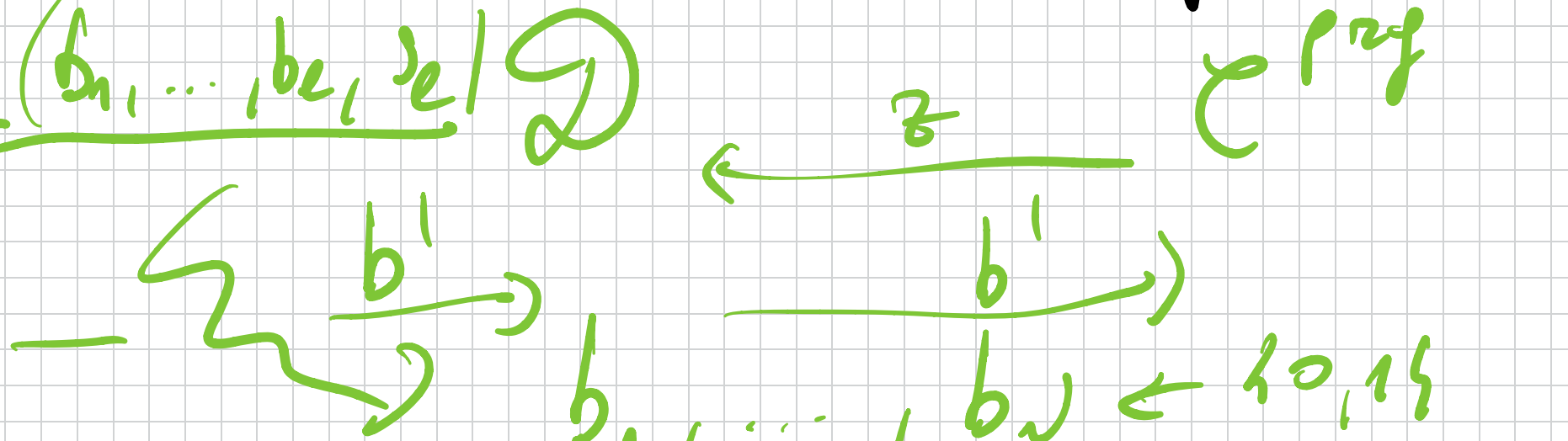
$b_1, \ldots, b_i \leftarrow \{0,1\}$

$H_i(m) \equiv$ $s_i \leftarrow \{0,1\}^n$ $(b_{i+1}, \ldots, b_\ell, s_\ell) = G^{\ell-i}(s_i)$

$H_\ell(m) \equiv U_{\ell+m}$

LEMMA $\forall i : H_J \approx_c H_{J+1}$.

Proof. By reduction (as before):



(TRUE in Both H_J, H_{J+1}.)

$z = s_{J+1} \| b_{i+1} \dots b_{\ell-i-1}$

$(b_{J+2}, \dots, b_\ell, s_\ell) = G^{\ell-i-1}(s_{J+1})$

(True in Both H_0, H_{0+1})

By the above observations:

$$\Pr[\mathcal{D}(z)=1 : z = G(s);\ s \leftarrow \{0,1\}^n]$$

$$= \Pr[\mathcal{D}'(b_1, \ldots, b_\ell, s_\ell) = 1 : (b_1, \ldots, b_\ell, s_\ell) \leftarrow H_{i(n)}]$$

$$\Pr[\mathcal{D}(z)=1 : z \leftarrow U_{n+\ell}] =$$

$$= \Pr\left[\mathcal{D}^1(b_1, \ldots, b_\ell, s_\ell) = 1 : \right.$$
$$\left. (b_1, \ldots, b_\ell, s_\ell) \in H_{n+1}(n) \right]$$

$$\Rightarrow \Big| \Pr\left[\mathcal{D}(z) = 1 : z = h(U_n) \right] \alpha$$
$$- \Pr\left[\mathcal{D}(z) = 1 : z \in U_{n+1} \right] \Big|$$

$$\geq 1/p^1(n).$$

$$\Rightarrow H_n \approx_c H_{n+1} \quad \blacksquare$$