2022201041

Build a Provably Secure PRG(Code)

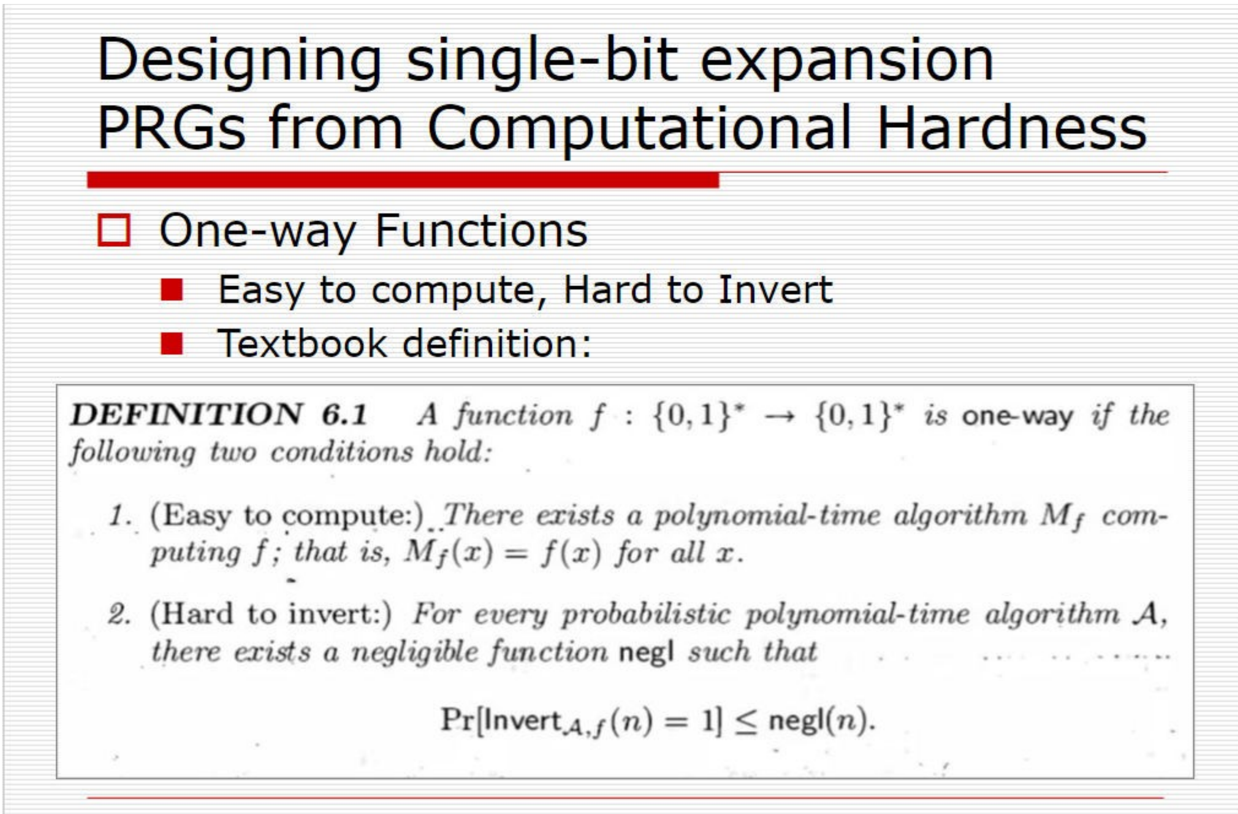
DEFINITION 3.14 *Let $\ell(\cdot)$ be a polynomial and let G be a deterministic polynomial-time algorithm such that for any input $s \in \{0,1\}^n$, algorithm G outputs a string of length $\ell(n)$. We say that G is a* pseudorandom generator *if the following two conditions hold:*

1. (Expansion:) *For every n it holds that $\ell(n) > n$.*
2. (Pseudorandomness:) *For all probabilistic polynomial-time distinguishers D, there exists a negligible function* negl *such that:*

$$\left| \Pr[D(r) = 1] - \Pr[D(G(s)) = 1] \right| \leq \mathsf{negl}(n),$$

where r is chosen uniformly at random from $\{0,1\}^{\ell(n)}$, the seed *s is chosen uniformly at random from $\{0,1\}^n$, and the probabilities are taken over the random coins used by D and the choice of r and s.*

The function $\ell(\cdot)$ is called the expansion factor *of G.*



Utility functions:

discrete_log: DLP One way function GENERATOR = 8173 MOD = 65521

Performs (GENERATOR^x) % (MOD)

Args: x (int): seed value

Returns: one way function value and get_hardcore_bit(x)

So, it is basically a deterministic polynomial time algorithm G, which takes input of n bits

and outputs l(n) bits where:

1. l(n) > n and

2. Output of G is computationally indistinguishable from uniform distribution.

Now, let’s figure out how to design a single-bit expansion PRGS from computational hardness.

HARDCORE PREDICATES

- Hardest bit of information about x to obtain from f(x)
- Textbook definition:

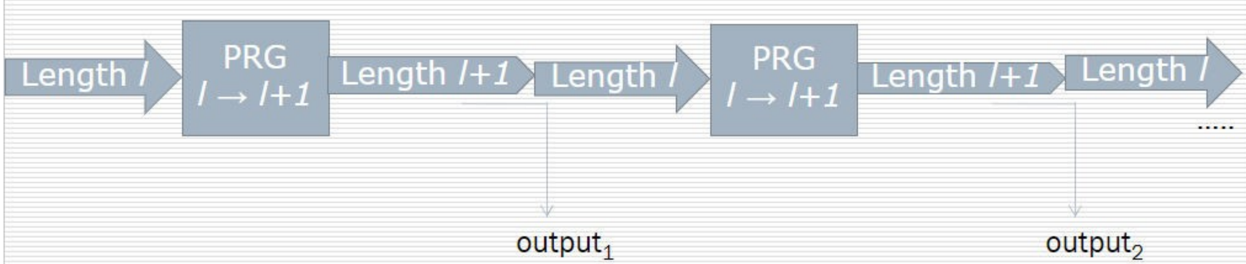
DEFINITION 6.5 *A function* hc : $\{0,1\}^* \to \{0,1\}$ *is a* hard-core predicate of a function f *if* **(1)** hc *can be computed in polynomial time, and* **(2)** *for every probabilistic polynomial-time algorithm* $\mathcal{A}$ *there exists a negligible function* negl *such that*

$$\Pr_{x \leftarrow \{0,1\}^n}[\mathcal{A}(f(x)) = \mathsf{hc}(x)] \leq \frac{1}{2} + \mathsf{negl}(n),$$

where the probability is taken over the uniform choice of x *in* $\{0,1\}^n$ *and the random coin tosses of* $\mathcal{A}$.

MSB(x) is a Hardcore predicate of Discrete Logarithm Problem

THEOREM 6.8 *Assume that there exists a pseudorandom generator with expansion factor* $\ell(n) = n + 1$. *Then for any polynomial* $p(\cdot)$, *there exists a pseudorandom generator with expansion factor* $\ell(n) = p(n)$.



1. Take the last bit from *l* + *1* length string for output
2. Apply *l′* times to get output of string *l′*