Strong One-Way Functions: A function $\mu: N \to R$ is called *negligible* if for every positive polynomial $p()$ and all sufficiently large n's, it holds that $\mu(n) < \frac{1}{p(n)}$.

A function $f: \{0,1\}^* \to \{0,1\}^*$ is called *strongly one-way* if the following two conditions hold:

(1) *Easy to compute:* There exists a deterministic polynomial-time algorithm A such that on input x algorithm A outputs $f(x)$: $A(x) = f(x)$.

(2) *Hard to invert:* For every probabilistic polynomial-time algorithm A', every positive polynomial $p()$, and all sufficiently large n's,

$$\Pr[A'(f(U_n), 1^n) \in f^{-1}(f(U_n))] < \frac{1}{p(n)}$$

Here U_n denotes a random variable uniformly distributed over $\{0,1\}^n$. The probability in (2) is taken over all possible values assigned to U_n and all possible internal coin tosses of A', with uniform probability distribution. A' is not required to output a specific pre-image of $f(x)$; any pre-image (element in the set $f^{-1}(f(x))$) will do. In case f is 1-1, the string x is the only pre-image of $f(x)$ under f; but in general there may be other pre-images.

The Auxiliary Input 1^n: In addition to an input in the range of f, the inverting algorithm A' is also given the length of the desired output (in unary notation). The main reason for this convention is to rule out the possibility that a function will be considered one-way merely because it drastically shrinks its input, and so the inverting algorithm just does not have enough time to print the desired output (the corresponding pre-image). Consider, for example, the function f_{len} defined by $f_{len}(x) = y$ such that y is the binary representation of the length of x ($f_{len}(x) = |x|$). Since $|f_{len}(x)| = \log_2|x|$, no algorithm can invert f_{len} on y in time polynomial in $|y|$; yet there exists an obvious algorithm that inverts f_{len} on $y = f_{len}(x)$ in time polynomial in $|x|$ ($|x| \longrightarrow 0^{|x|}$). In general, the auxiliary input $1^{|x|}$, provided in conjunction with the input $f(x)$, allows the inverting algorithm to run in time polynomial in the total length of the main input and the desired output.

Weak One-Way Function: A function $f: \{0,1\}^* \to \{0,1\}^*$ is called **weakly one-way** if the following two conditions hold:

① **Easy to compute**: There exists a deterministic polynomial-time algorithm A such that on input x algorithm A outputs $f(x)$: $A(x) = f(x)$.

② **Slightly hard to invert**: There exists a polynomial $p(\cdot)$ such that for every probabilistic polynomial-time algorithm A' and all sufficiently large n's,

$$\Pr[A'(f(U_n), 1^n) \notin f^{-1}(f(U_n))] > \frac{1}{p(n)}$$

Non-Uniformly Strong One-Way Function: A function $f: \{0,1\}^* \to \{0,1\}^*$ is called **non-uniformly strong one-way** if the following two conditions hold:

① **Easy to compute**: There exists a polynomial-time algorithm A such that on input x algorithm A outputs $f(x)$.

② **Hard to invert**: For every family of polynomial-size circuits $\{C_n\}_{n \in \mathbb{N}}$, every polynomial $p(\cdot)$, and all sufficiently large n's,

$$\Pr[C_n(f(U_n)) \in f^{-1}(f(U_n))] < \frac{1}{p(n)}$$

Here the probability is taken over all possible values of U_n.