The RSA collection satisfies the first condition for the definition of a one-way collection that it is easy to sample and compute. On input 1^n, algorithm I_{RSA} selects uniformly two primes, P and Q, such that $2^{n-1} \leq P < Q < 2^n$, and an integer e such that e is relatively prime to $(P-1)(Q-1)$. Algorithm I_{RSA} terminates with output (N, e), where $N = P.Q$. For an efficient implementation of I_{RSA}, we can use a probabilistic polynomial-time algorithm for generating uniformly or almost uniformly distributed primes.
The algorithm D_{RSA}, on input (N, e) selects almost uniformly an element in the set $D_{N,e} = \{1, \dots N\}$.
The output of F_{RSA}, on input $((N, e), x)$, is

$$RSA_{N,e}(x) = x^e \pmod{N}.$$

For the second condition, it is widely believed that the RSA collection is strongly one-way.

<u>Polynomial-time reductions</u>: Given two problems P_1 and P_2 we say that P_1 reduces to P_2 in polynomial time if there exists a polynomial-time algorithm R that takes an instance x of problem P_1 and converts it into instance y of problem P_2 such that we can use the solution of y to solve the instance x. We denote it is $P_1 \leq_p P_2$. We use $\leq_p$ sign which is similar to $\leq$ in terms of difficulty of solving the problems.

There are four possibilities for difficulty of P_1 and P_2:

① P_1 is easy, P_2 is easy. ($P_1 = P_2$)

② P_1 is easy, P_2 is difficult. ($P_1 < P_2$)

③ P_1 is difficult, P_2 is easy. ($P_1 > P_2$)

④ P_1 is difficult, P_2 is difficult. ($P = P_2$)

If $P_1 \leq_p P_2$ then it rules out the possibility of ③ because we can solve any instance x of P_1 easily by first converting x to $y \in P_2$ in polynomial-time using R (easy) and then solving y (easy), and then using the solution of y to solve x (easy).

<u>Inverting RSA $\leq_p$ Factoring</u> : Suppose the instance of Inverting RSA is $(N, e, x^e \pmod N)$. We convert this instance into a Factoring instance as (N). Suppose we have an efficient algorithm for factoring N. We will get P and Q easily. Now we can easily compute $\phi(N) = (P-1)(Q-1)$ and also $d \equiv e^{-1} \pmod{\phi(N)}$. Now using this information, we can easily invert the RSA instance: $(x^e)^d \equiv x \pmod{\phi(N)}$.

<u>Factoring $\leq_p$ Inverting RSA ?</u> : This is an <u>open problem</u>.

<u>The Chinese Remainder Theorem</u>: Let $m_1, m_2, \ldots, m_r$ denote r positive integers that are relatively prime in pairs, and let $a_1, a_2, \ldots, a_r$ denote any r integers. Then the congruences

$$x \equiv a_1 \pmod{m_1},$$
$$x \equiv a_2 \pmod{m_2},$$
$$\vdots$$
$$x \equiv a_r \pmod{m_r}.$$

have common solutions.

If x_0 is one such solution, then an integer x satisfies the above congruences if and only if x is of the form $x = x_0 + km$ for some integer k. Here $m = m_1 m_2 \ldots m_r$.

$\frac{m}{m_j}$ is an integer and $\left(\frac{m}{m_j}, m_j\right) = 1 \Rightarrow$ for each j there is an integer b_j such that $\left(\frac{m}{m_j}\right) b_j \equiv 1 \pmod{m_j}$.

For $i \neq j$ we have $\left(\frac{m}{m_j}\right) b_j \equiv 0 \pmod{m_i}$.

Let $x_0 = \sum_{j=1}^{r} \frac{m}{m_j} b_j a_j \Rightarrow x_0 \equiv \frac{m}{m_i} b_i a_i \equiv a_i \pmod{m_i}$

$\Rightarrow x_0$ is solution of the above congruences.

If x_0 and x_1 are two solutions then $x_0 \equiv x_1 \pmod{m_i}$ for $i = 1, 2, \ldots r \Rightarrow x_0 \equiv x_1 \pmod{m}$ because $m_1, m_2 \ldots m_r$ are relatively prime in pairs.

<u>Quadratic Residues</u>: For all a such that $(a, m) = 1$, a is called a quadratic residue modulo m if the congruence $x^2 \equiv a \pmod{m}$ has a solution. If it has no solution, then a is called a quadratic non-residue modulo m. For an odd prime p, exactly half the elements of Z_p^* are quadratic residues.