Randomized Algorithms

南京大学

尹一通

Polynomial Identity Testing (PIT)

Input: $f, g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv g?$

$\mathbb{F}[x_1, x_2, \ldots, x_n]$: ring of *n*-variate polynomials over field $\mathbb{F}$

$f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i_1, i_2, \ldots, i_n \geq 0} a_{i_1, i_2, \ldots, i_n} x_1^{i_1} x_2^{i_2} \cdots x_n^{i_n}$$

degree of f: maximum $i_1 + i_2 + \cdots + i_n$ with $a_{i_1, i_2, \ldots, i_n} \neq 0$

Input: $f, g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv g?$

equivalently:

Input: $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv 0?$

f is given as block-box: given any $\vec{x} = (x_1, x_2, \ldots, x_n)$

returns $f(\vec{x})$

or as product from: e.g. Vandermonde determinant

$$M = \begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^{n-1} \end{bmatrix}$$

$$f(\vec{x}) = \det(M) = \prod_{j<i} (x_i - x_j)$$

Input: $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv 0?$

fix an arbitrary $S \subseteq \mathbb{F}$

pick random $r_1, r_2, \ldots, r_n \in S$
uniformly and *independently* at random;
check whether $f(r_1, r_2, \ldots, r_n) = 0$;

$$f \equiv 0 \Rightarrow f(r_1, r_2, \ldots, r_n) = 0$$

Input: a polynomial $f \in \mathbb{F}[x]$ of degree d

Output: $f \equiv 0?$

fix an arbitrary $S \subseteq \mathbb{F}$

pick a *uniform* random $r \in S$;

check whether $f(r) = 0$;

$$f \equiv 0 \Rightarrow f(r) = 0$$

Fundamental Theorem of Algebra:

A degree d polynomial has at most d roots.

$$f \not\equiv 0 \Rightarrow \Pr[f(r) = 0] \leq \frac{d}{|S|}$$

Input: $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv 0?$

fix an arbitrary $S \subseteq \mathbb{F}$

pick random $r_1, r_2, \ldots, r_n \in S$
uniformly and *independently* at random;
check whether $f(r_1, r_2, \ldots, r_n) = 0$;

$$f \equiv 0 \Rightarrow f(r_1, r_2, \ldots, r_n) = 0$$

Schwartz-Zippel Theorem

$$f \not\equiv 0 \Rightarrow \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

Schwartz-Zippel Theorem

$$f \not\equiv 0 \quad \Rightarrow \quad \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

induction on n :

basis: $n=1$ single-variate case, proved by the *fundamental Theorem of algebra*

I.H.: Schwartz-Zippel Thm is true for all smaller n

Schwartz-Zippel Theorem

$$f \not\equiv 0 \quad \Rightarrow \quad \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

induction step:

k: highest power of x_n in f $\Rightarrow$ $\begin{cases} f_k \not\equiv 0 \\ \text{degree of } f_k \leq d - k \end{cases}$

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=0}^{k} x_n^i f_i(x_1, x_2, \ldots, x_{n-1})$$

$$= x_n^k f_k(x_1, x_2, \ldots, x_{n-1}) + \bar{f}(x_1, x_2, \ldots, x_n)$$

where $\bar{f}(x_1, x_2, \ldots, x_n) = \sum_{i=0}^{k-1} x_n^i f_i(x_1, x_2, \ldots, x_{n-1})$

highest power of x_n in $\bar{f} \; < k$

Schwartz-Zippel Theorem

$$f \not\equiv 0 \quad \Rightarrow \quad \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

$$f(x_1, x_2, \ldots, x_n) = x_n^k f_k(x_1, x_2, \ldots, x_{n-1}) + \bar{f}(x_1, x_2, \ldots, x_n)$$

$$\begin{cases} f_k \not\equiv 0 \\ \text{degree of } f_k \leq d - k \end{cases}$$

highest power of x_n in $\bar{f}$ $< k$

law of total probability:

$$\Pr[f(r_1, r_2, \ldots, r_n) = 0]$$

I.H. $\Rightarrow$ $\leq \frac{d-k}{|S|}$

$$= \Pr[f(\vec{r}) = 0 \mid f_k(r_1, \ldots, r_{n-1}) = 0] \cdot \Pr[f_k(r_1, \ldots, r_{n-1}) = 0]$$

$$+ \Pr[f(\vec{r}) = 0 \mid f_k(r_1, \ldots, r_{n-1}) \neq 0] \cdot \Pr[f_k(r_1, \ldots, r_{n-1}) \neq 0]$$

$$= \Pr[g_{r_1,\ldots,r_{n-1}}(r_n) = 0 \mid f_k(r_1, \ldots, r_{n-1}) \neq 0] \leq \frac{k}{|S|}$$

where $g_{x_1,\ldots,x_{n-1}}(x_n) = f(x_1, \ldots, x_n)$

Schwartz-Zippel Theorem

$$f \not\equiv 0 \quad \Rightarrow \quad \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

$$\Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d-k}{|S|} + \frac{k}{|S|} = \frac{d}{|S|}$$

Input: $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv 0?$

fix an arbitrary $S \subseteq \mathbb{F}$

pick random $r_1, r_2, \ldots, r_n \in S$
uniformly and *independently* at random;
check whether $f(r_1, r_2, \ldots, r_n) = 0$;

$$f \equiv 0 \Rightarrow f(r_1, r_2, \ldots, r_n) = 0$$

Schwartz-Zippel Theorem

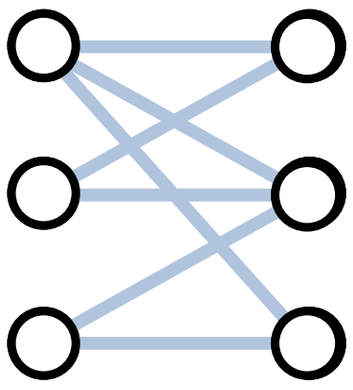
$$f \not\equiv 0 \Rightarrow \Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}$$

Applications of *Schwartz-Zippel*

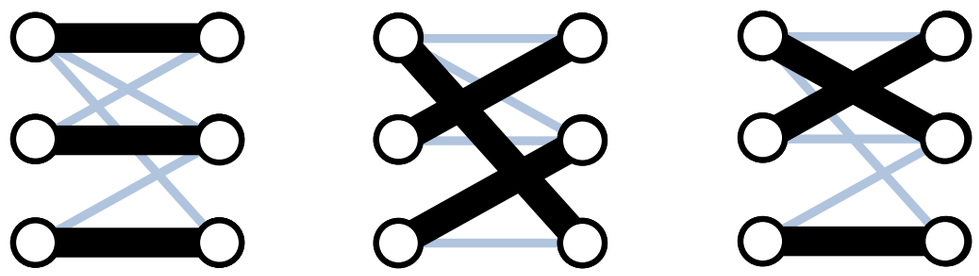
- test whether a graph has perfect matching;
- test isomorphism of rooted trees;
- distance property of Reed-Muller codes;
- proof of hardness vs randomness tradeoff;
- algebraic construction of probabilistically checkable proofs (PCP);
- ....

Bipartite Perfect Matching

bipartite graph



perfect matchings

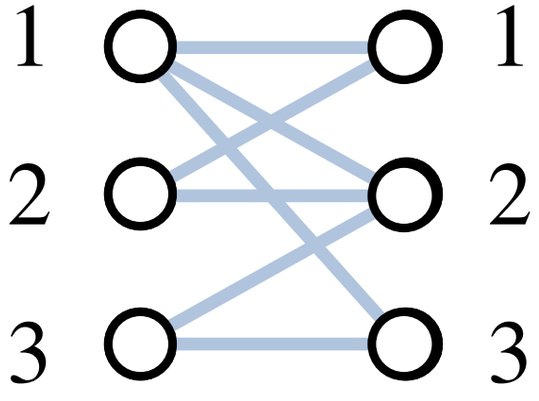


$G([n],[n],E)$

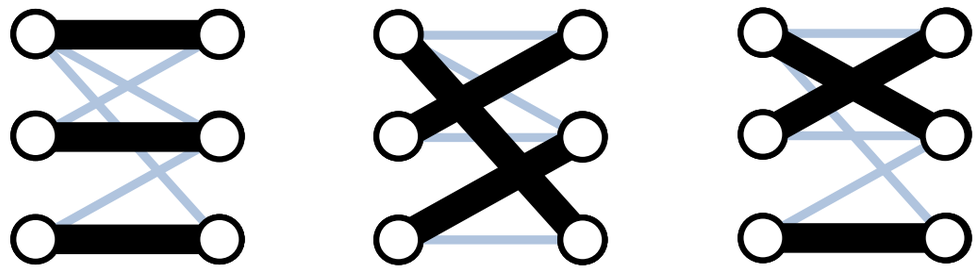
determine whether a bipartite graph has PM:

- Hall's theorem: enumerates all subset of $[n]$
- Hungarian method: $O(n^3)$
- Hopcroft-Karp algorithm: $O(m\sqrt{n})$
- efficient parallel algorithms?

bipartite graph



$G([n],[n],E)$



$$A = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & 0 \\ 0 & x_{32} & x_{33} \end{bmatrix}$$

$$\det(A) = x_{11}x_{22}x_{33} + x_{13}x_{21}x_{32} - x_{12}x_{21}x_{33}$$

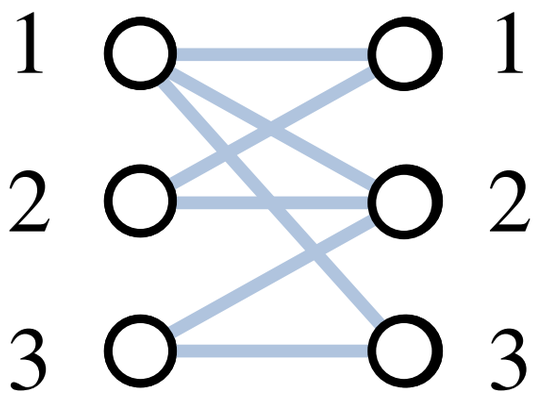
Edmond matrix:

$$A(i,j) = \begin{cases} x_{i,j} & \text{if } (i,j) \in E \\ 0 & \text{if } (i,j) \notin E \end{cases}$$

$$\det(A) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i \in [n]} A(i, \pi(i)) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i \in [n]} x_{i,\pi(i)}$$

$\not\equiv 0$ if and only if ∃ a perfect matching

bipartite graph



$G([n],[n],E)$

Edmond matrix:

$$A(i,j) = \begin{cases} x_{i,j} & \text{if } (i,j) \in E \\ 0 & \text{if } (i,j) \notin E \end{cases}$$

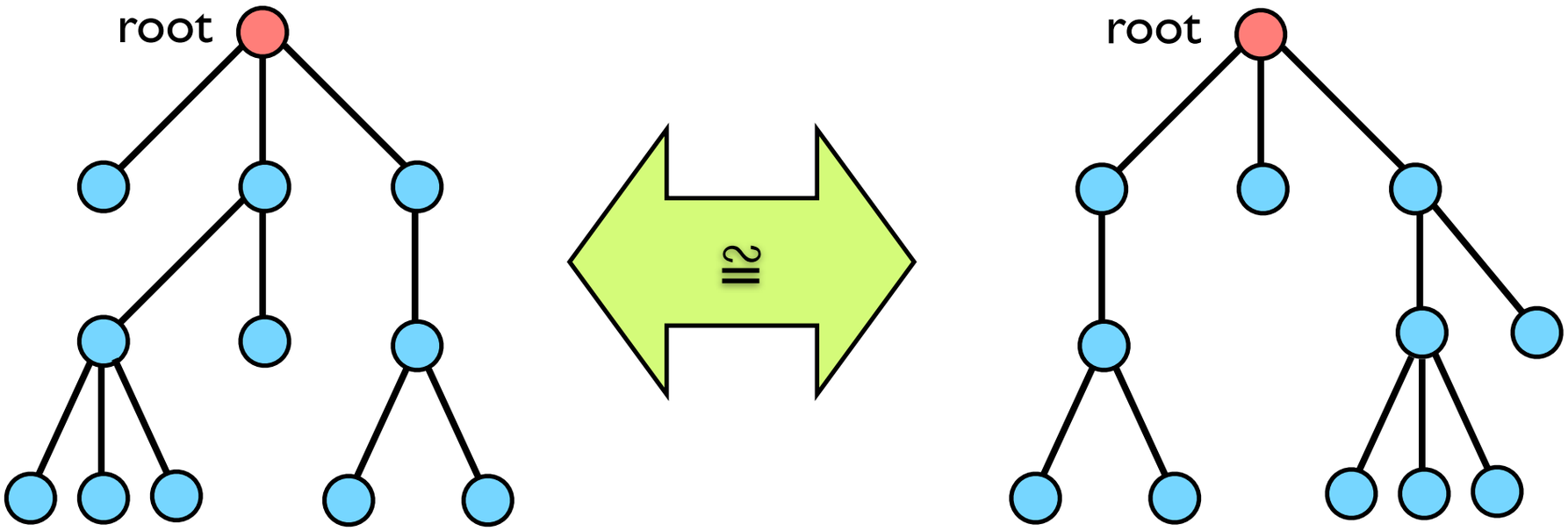
$\det(A) \not\equiv 0$

if and only if ∃ a perfect matching

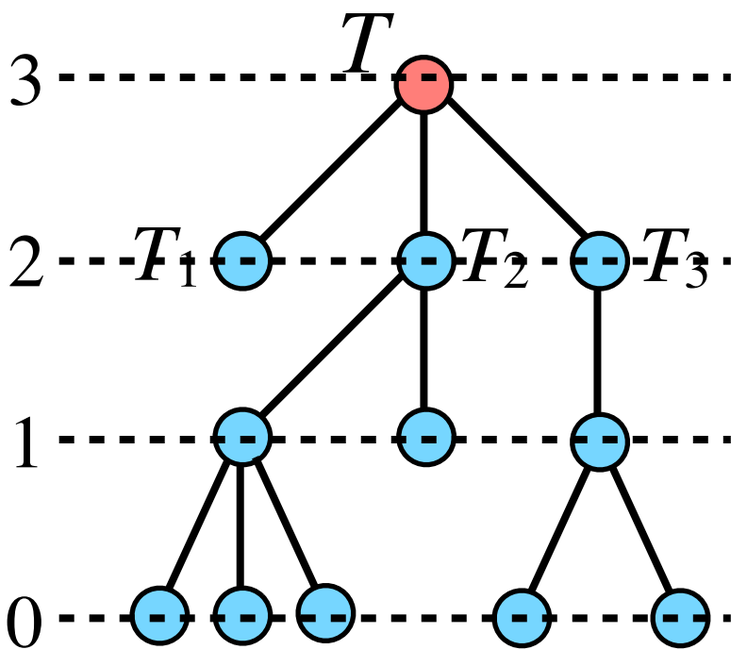
$\det(A)$ is an n-variate degree-n polynomial:

- use *Schwartz-Zippel* to check whether $\det(A) \not\equiv 0$
- there are fast parallel algorithms for computing determinants of numerical matrices (Chistov's algorithm)

Isomorphism of Rooted Trees



Isomorphism of Rooted Trees



associate each tree T a polynomial:

T is of height k

subtrees rooted by children of the root of T: $T_1, T_2, ..., T_m$

$$f_T = (x_k - f_{T_1})(x_k - f_{T_2}) \cdots (x_k - f_{T_m})$$

$$f_T = (x_3 - x_0)(x_3 - (x_2 - (x_1 - x_0)^3)(x_2 - x_0))(x_3 - (x_2 - (x_1 - x_0)^2))$$

by induction and the *uniqueness of polynomial factorization*:

$$f_T \equiv f_{T'} \iff T \cong T'$$

Fingerprinting

$$X \quad = \quad Y \quad ?$$

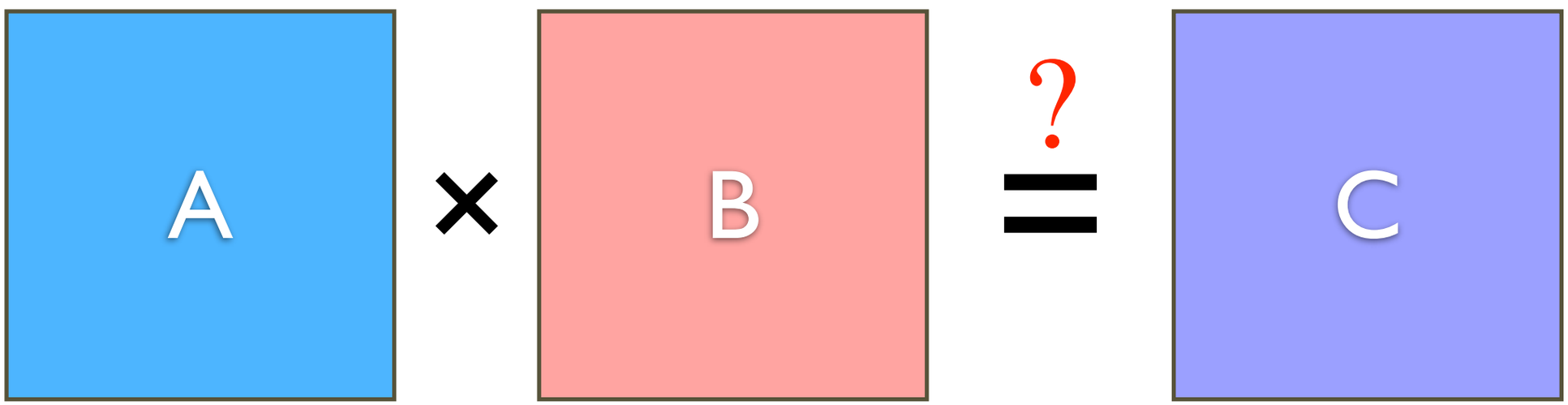
$$\downarrow \qquad\qquad \downarrow$$

$$\text{FING}(X) \quad = \quad \text{FING}(Y) \quad ?$$

- FING( ) is a function: $X{=}Y \Rightarrow \text{FING}(X) = \text{FING}(Y)$
- if $X \neq Y$, Pr[FING(X) = FING(Y)] is small.
- Fingerprints are easy to compute and compare.

Checking Matrix Multiplication

three $n \times n$ matrices A, B, C:



Freivald's Algorithm
pick a uniform random $r \in \{0,1\}^n$;
check whether $A(Br) = Cr$;

matrix M:

$\mathrm{FING}(M) = Mr$ for uniform random $r \in \{0,1\}^n$

Polynomial Identity Testing (PIT)

Input: $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d

Output: $f \equiv 0?$

fix an arbitrary $S \subseteq \mathbb{F}$

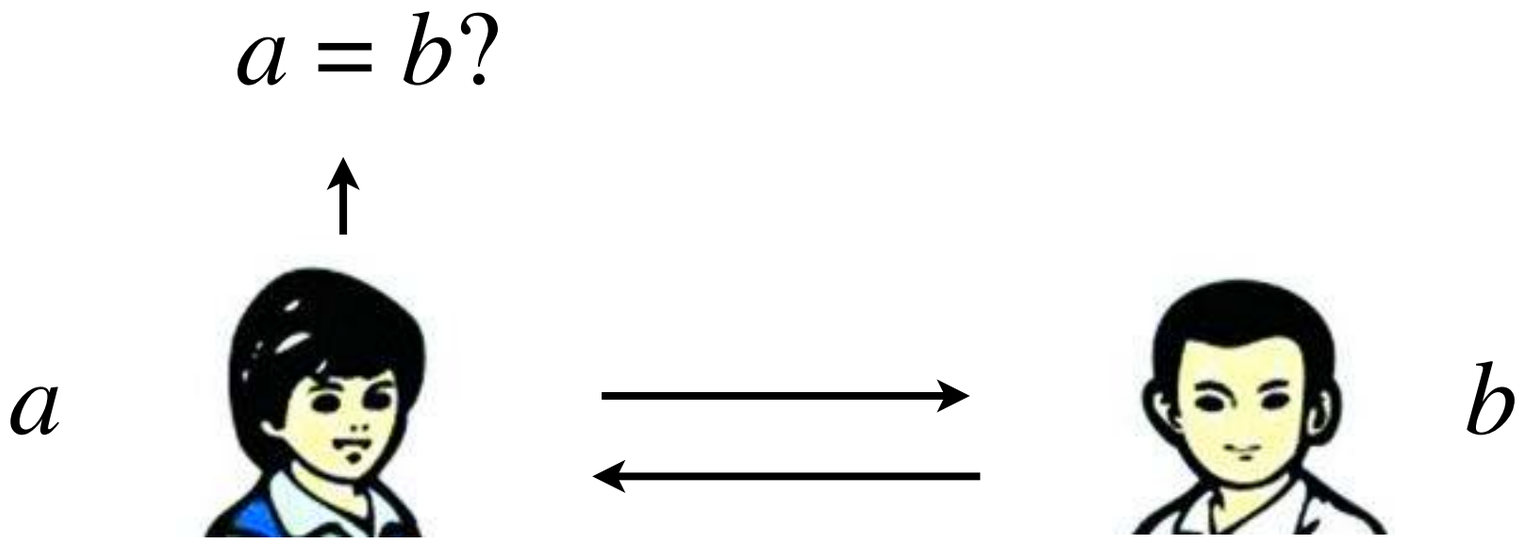
pick random $r_1, r_2, \ldots, r_n \in S$
uniformly and *independently* at random;

check whether $f(r_1, r_2, \ldots, r_n) = 0$;

polynomial f:

$\mathrm{FING}(f) = f(r_1, r_2, \ldots, r_n)$ for uniform&independent $r_1, \ldots, r_n \in S$

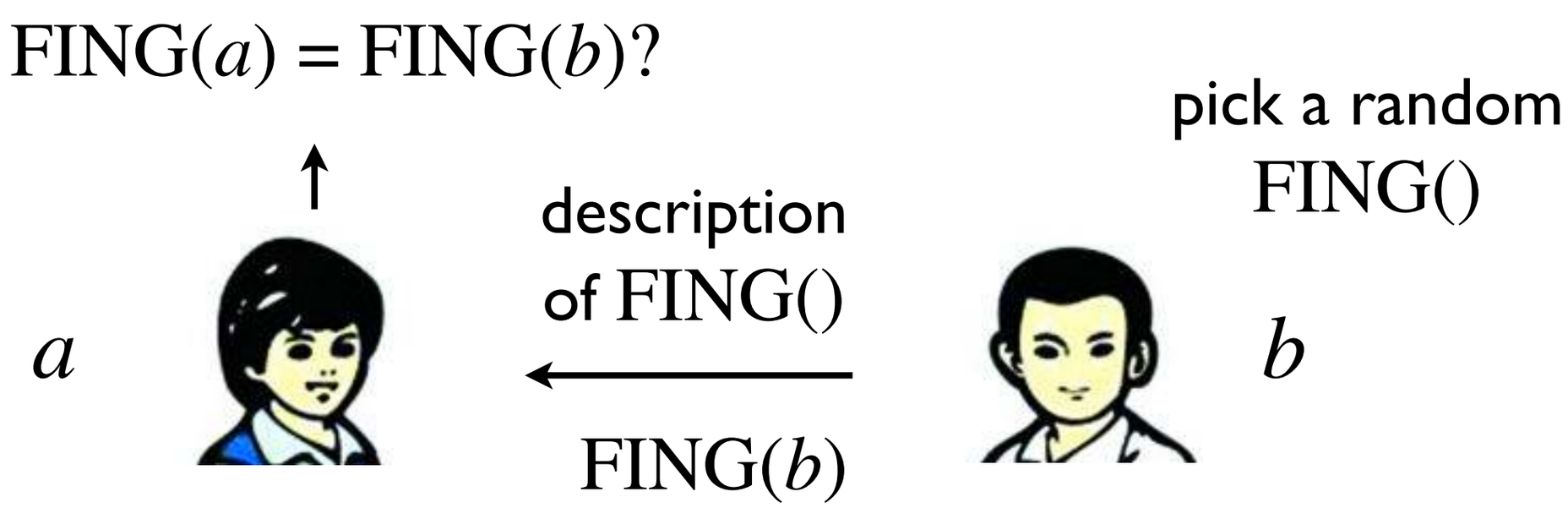
Communication Complexity



$$\mathrm{EQ} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$$

$$\mathrm{EQ}(a,b) = \begin{cases} 1 & a = b \\ 0 & a \neq b \end{cases}$$

Fingerprinting



- FING( ) is a function: $a=b \Rightarrow$ FING(a) = FING(b)
- if $a \neq b$, Pr[FING(a) = FING(b)] is small.
- Fingerprints are easy to compute and compare.

$$f = \sum_{i=0}^{n-1} a_i x^i$$

$a \in \{0,1\}^n$

$f(r)=g(r)$?

$r, g(r)$

$$g = \sum_{i=0}^{n-1} b_i x^i$$

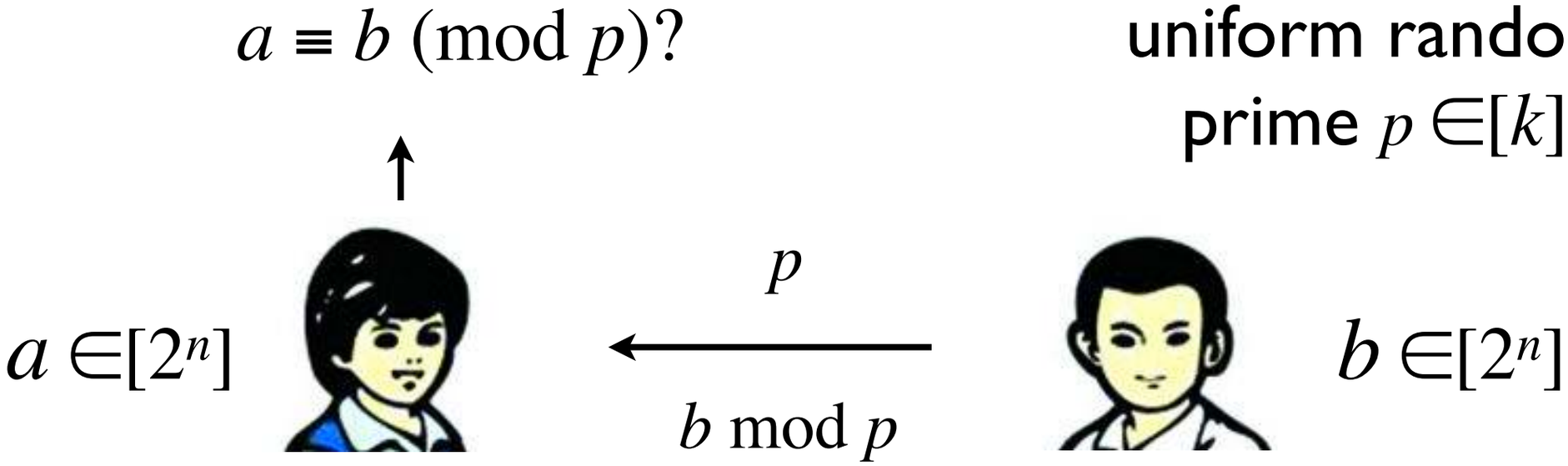
$b \in \{0,1\}^n$

pick uniform random $r \in [2n]$

$f, g \in \mathbb{Z}_p[x]$

prime $p \in [2^k, 2^{k+1}]$ for $k = \lceil \log_2(2n) \rceil$

$\mathrm{FING}(b) = \Sigma_i \, b_i \, r^i$ for random r



$\mathrm{FING}(x) = x \bmod p$ for uniform random prime $p \in [k]$

communication complexity: $O(\log k)$

if $a = b$ ⇨ $a \equiv b \pmod{p}$

if $a \neq b$: $\Pr[a \equiv b \pmod{p}] \leq ?$

for a $z = |a - b| \neq 0$: $\Pr[z \bmod p = 0] \leq ?$

uniform random prime $p \in [k]$

for a $z = |a - b| \neq 0$: $\Pr[z \bmod p = 0] \leq ?$

$\in [2^n]$

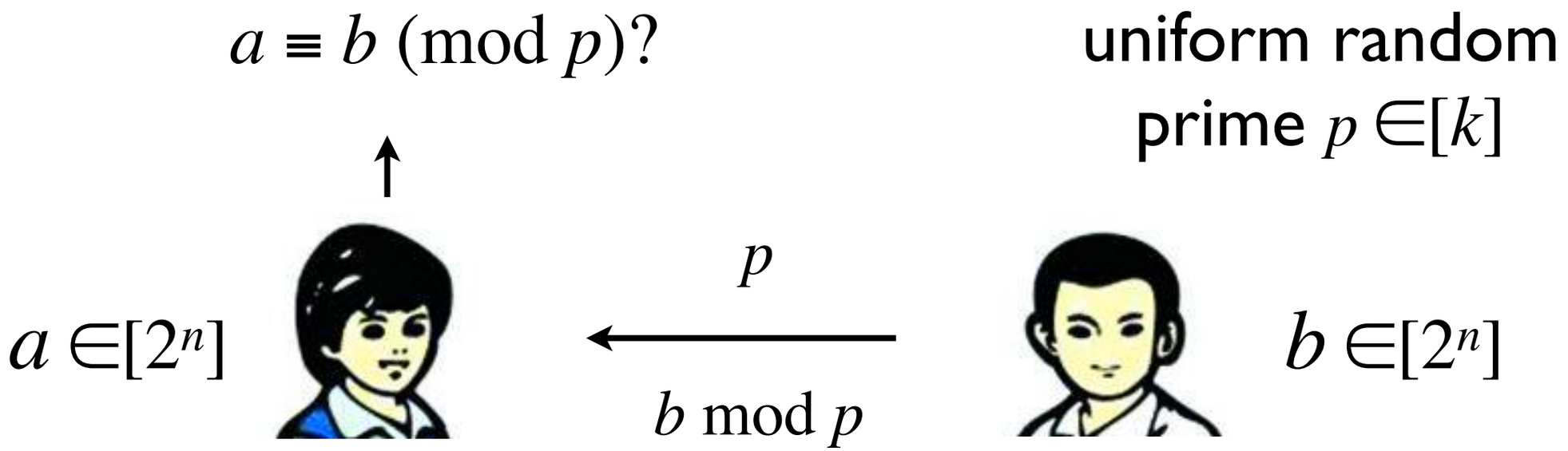
each prime divisor ≥ 2 } ⇒ # of prime divisors of $z \leq n$

$$\Pr[z \bmod p = 0] = \frac{\#\text{ of prime divisors of } z \quad \leq n}{\#\text{ of primes in } [k] \quad = \pi(k)}$$

$\pi(N)$: # of primes in $[N]$

Prime Number Theorem (PNT)

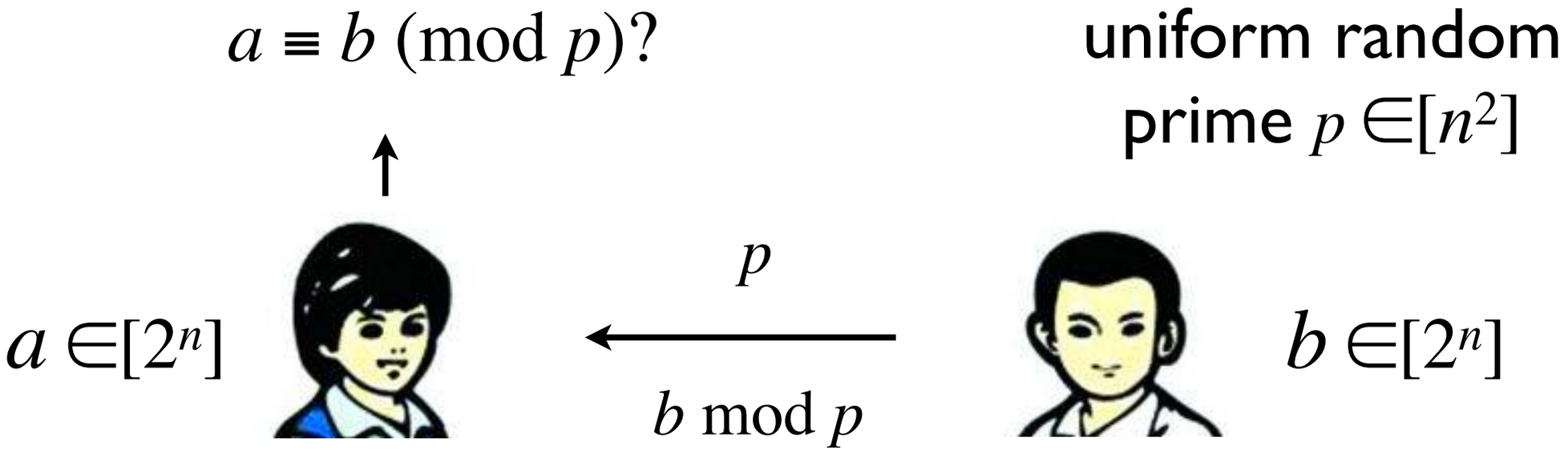
$$\pi(n) \sim \frac{N}{\ln N} \quad \text{as } N \longrightarrow \infty$$



for a $z = |a - b| \neq 0$: $\Pr[z \bmod p = 0] \leq ?$

$$\Pr[z \bmod p = 0] = \frac{\#\text{ of prime divisors of } z}{\#\text{ of primes in } [k]} \quad \begin{matrix} \leq n \\ = \pi(k) \end{matrix}$$

choose $k = n^2$ $\quad \leq \dfrac{n \ln k}{k} = \dfrac{2 \ln n}{n}$



FING(b) = $b \bmod p$ for uniform random prime $p \in [n^2]$

communication complexity: $O(\log n)$

if $a = b$ ⇨ $a \equiv b \pmod{p}$

if $a \neq b$ ⇨ $\Pr[a \equiv b \pmod{p}] \leq (2 \ln n) / n$

Pattern Matching

- **Input**: string $x \in \{0,1\}^n$, pattern $y \in \{0,1\}^m$
  - check wether y is a substring of x
  - naive algorithm: $O(mn)$ time
  - the *Knuth-Morris-Prat* algorithm: $O(m+n)$ time

Pattern Matching

$x(i) = y$?

y

$x(i)$

y : y_1 y_2 ····· y_m $\in\{0,1\}^m$

x : x_1 ····· x_i x_{i+1} ····· x_{i+m-1} ····· x_n $\in\{0,1\}^n$

$x(i)$

denote $x(i) = x_i x_{i+1} \ldots x_{i+m-1}$

$\mathrm{FING}(a) = a \bmod p$

pick a random FING();
for i=1, 2, ..., n-m+1;
    if FING($x(i)$) = FING(y) then return "match!";
return "not match!";

Pattern Matching

$x(i) = y$?

y

y : y_1 y_2 ····· y_m $\in\{0,1\}^m$

$x(i)$

x : x_1 ····· x_i x_{i+1} ····· x_{i+m-1} ····· x_n $\in\{0,1\}^n$

$x(i)$

denote $x(i) = x_i x_{i+1} \ldots x_{i+m-1}$

$\mathrm{FING}(a) = a \bmod p$

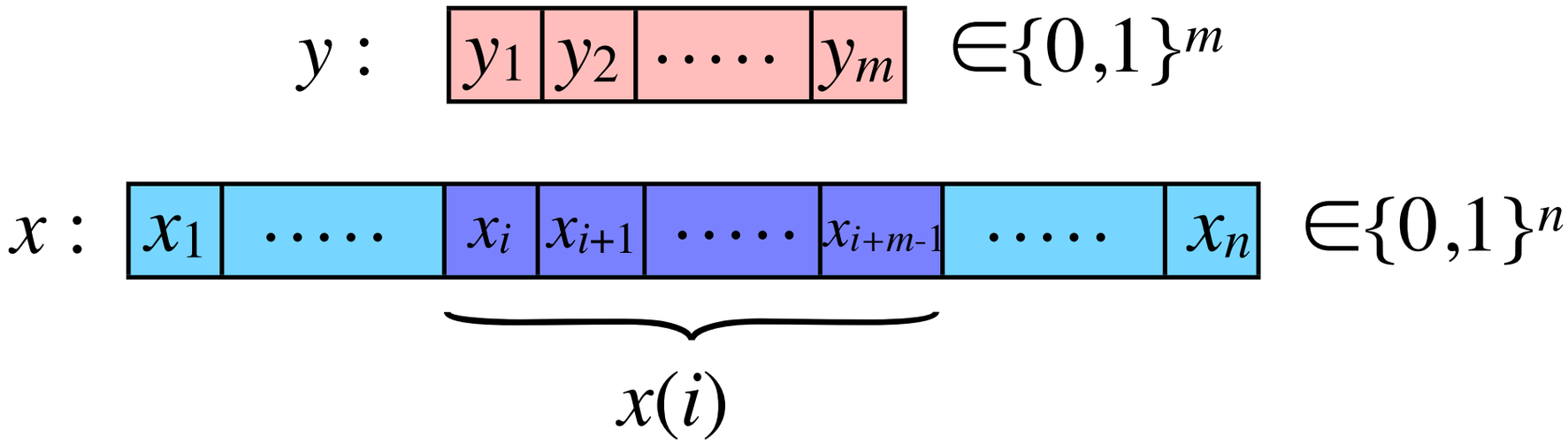
Karp-Rabin Algorithm:

pick a uniform random prime $p \in [mn^2]$;

for i=1, 2, ..., n-m+1;

if $x(i) \equiv y \pmod p$ then return "match!";

return "not match!";



Karp-Rabin Algorithm:

pick a uniform random prime $p \in [mn^2]$;

for i=1, 2, ..., n-m+1;

if $x(i) \equiv y \pmod p$ then return "match!";

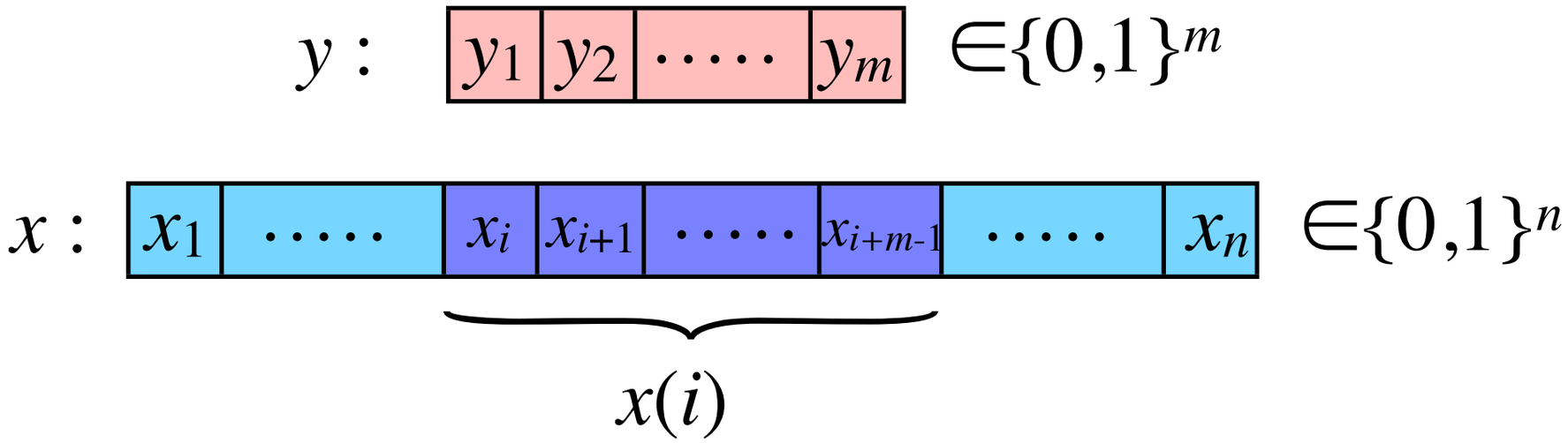
return "not match!";

for each i, if $x(i) \neq y$

$$\Pr[\, x(i) \equiv y \pmod p \,] = m \ln(mn^2) / mn^2 = o(1/n)$$

$$\Pr[\text{ mistake }] = \Pr[\, \exists\, i, x(i) \neq y \wedge x(i) \equiv y \pmod p \,] = o(1)$$

union bound



Karp-Rabin Algorithm:

pick a uniform random prime $p \in [mn^2]$;

for i=1, 2, ..., n-m+1;

if $x(i) \equiv y \pmod p$ then return "match!";

return "not match!";

$$\mathrm{FING}(a) = a \bmod p$$

$$x(i+1) = x_{i+m} + 2(x(i) - 2^{m-1}x_i)$$

$$\mathrm{FING}(x(i+1)) = \left(x_{i+m} + 2(\mathrm{FING}(x(i)) - 2^{m-1}x_i) \right) \bmod p$$

"$x(i) \equiv y \pmod p$" can be tested in constant time