# Constant-Cost Communication

Yuting Fang    Mika Göös    Nathan Harms    Pooya Hatami

**Alice:** $x \in \{0,1\}^n$

**Bob:** $y \in \{0,1\}^n$

**Q:** How many bits to communicate to decide $x = y$?

**Alice:** $x \in \{0,1\}^n$

**Bob:** $y \in \{0,1\}^n$

**Q:** How many bits to communicate to decide $x = y$?

**Deterministic: n bits**

**Alice:** $x \in \{0,1\}^n$

**Bob:** $y \in \{0,1\}^n$

**Q:** How many bits to communicate to decide $x = y$?

**Deterministic: n bits**

**Randomised: O(1)**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Equality Problem**

$2^n$

$2^n$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Equality Problem**

Equality Problem

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Shared randomness

| | 01 | 11 | 00 | 10 | 01 | 00 | 11 | 10 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 01 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 00 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Equality Problem

Shared randomness

| | 01 | 11 | 00 | 10 | 01 | 00 | 11 | 10 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 01 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 00 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Cost:** 3 bits

**Error:** 1/4

Equality Problem

Main question

# Which problems have constant cost?

Main question

# Which problems have constant cost?

TL;DR

- New perspective + tools
- Few examples known
- Many open problems!

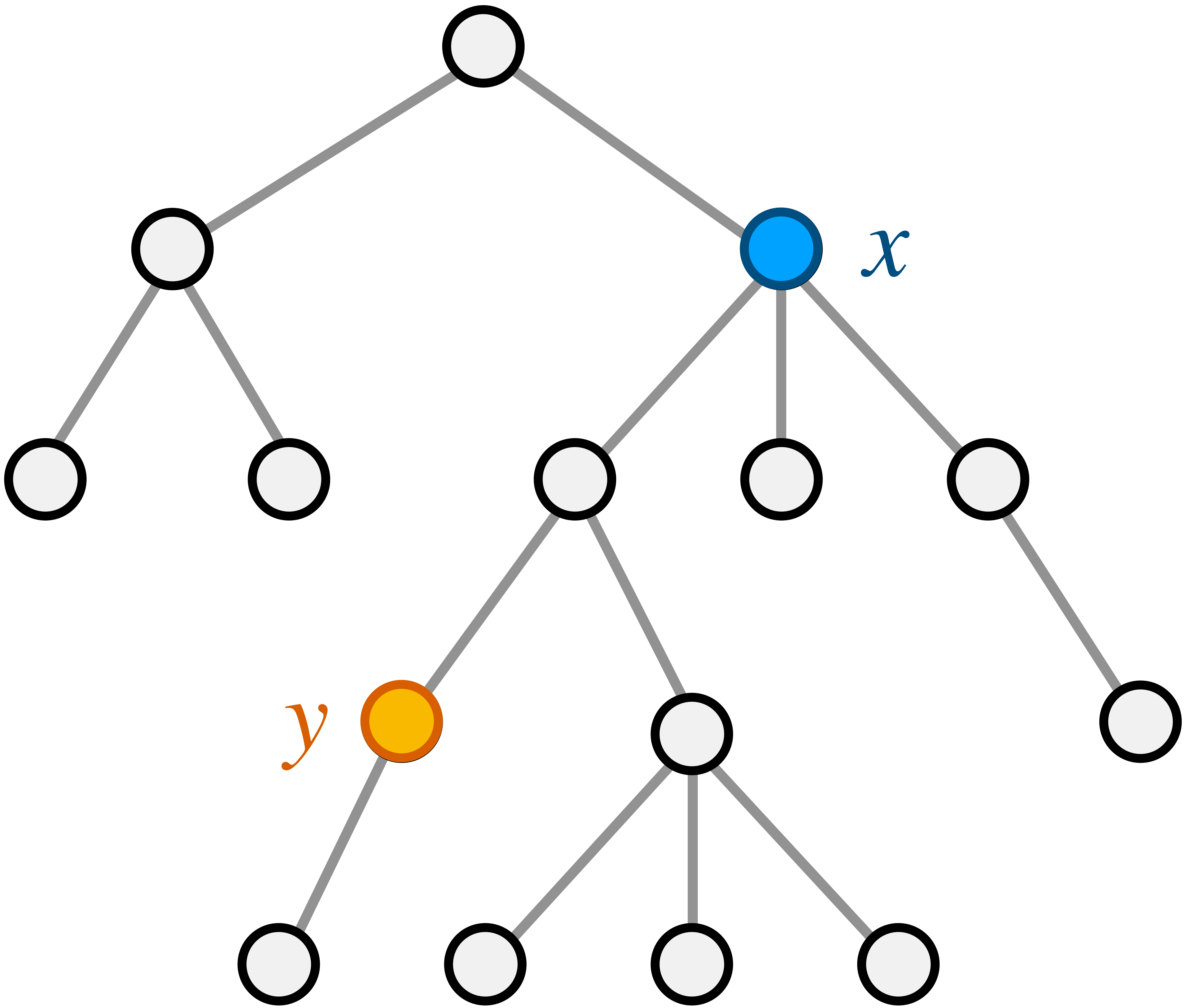
# Quiz

**Alice:** $x \in V$

**Bob:** $y \in V$

Does $x \sim y$?

**Tree Adjacency Problem**



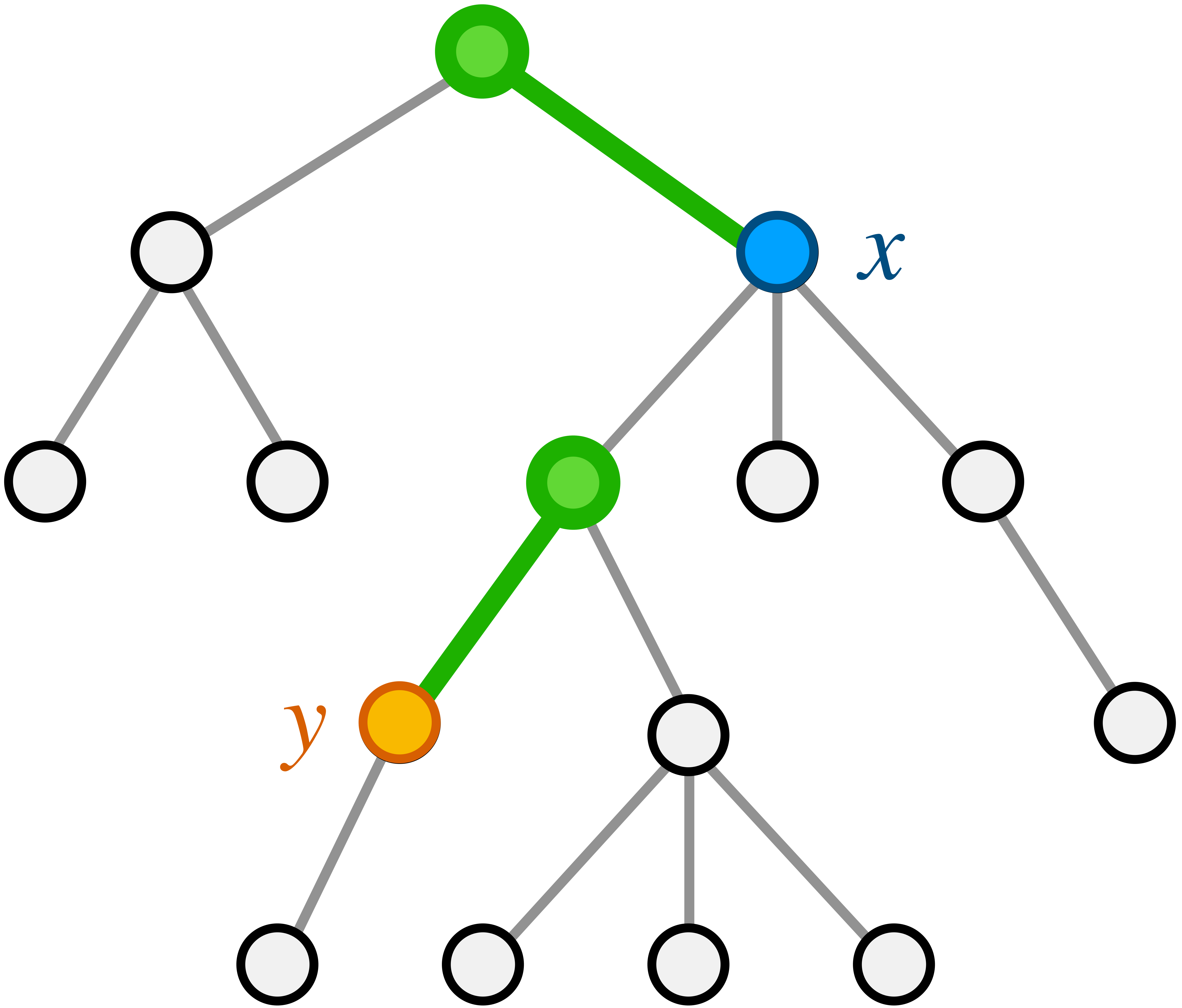
# Quiz

**Alice:** $x \in V$

**Bob:** $y \in V$

Does $x \sim y$?

**Tree Adjacency Problem**



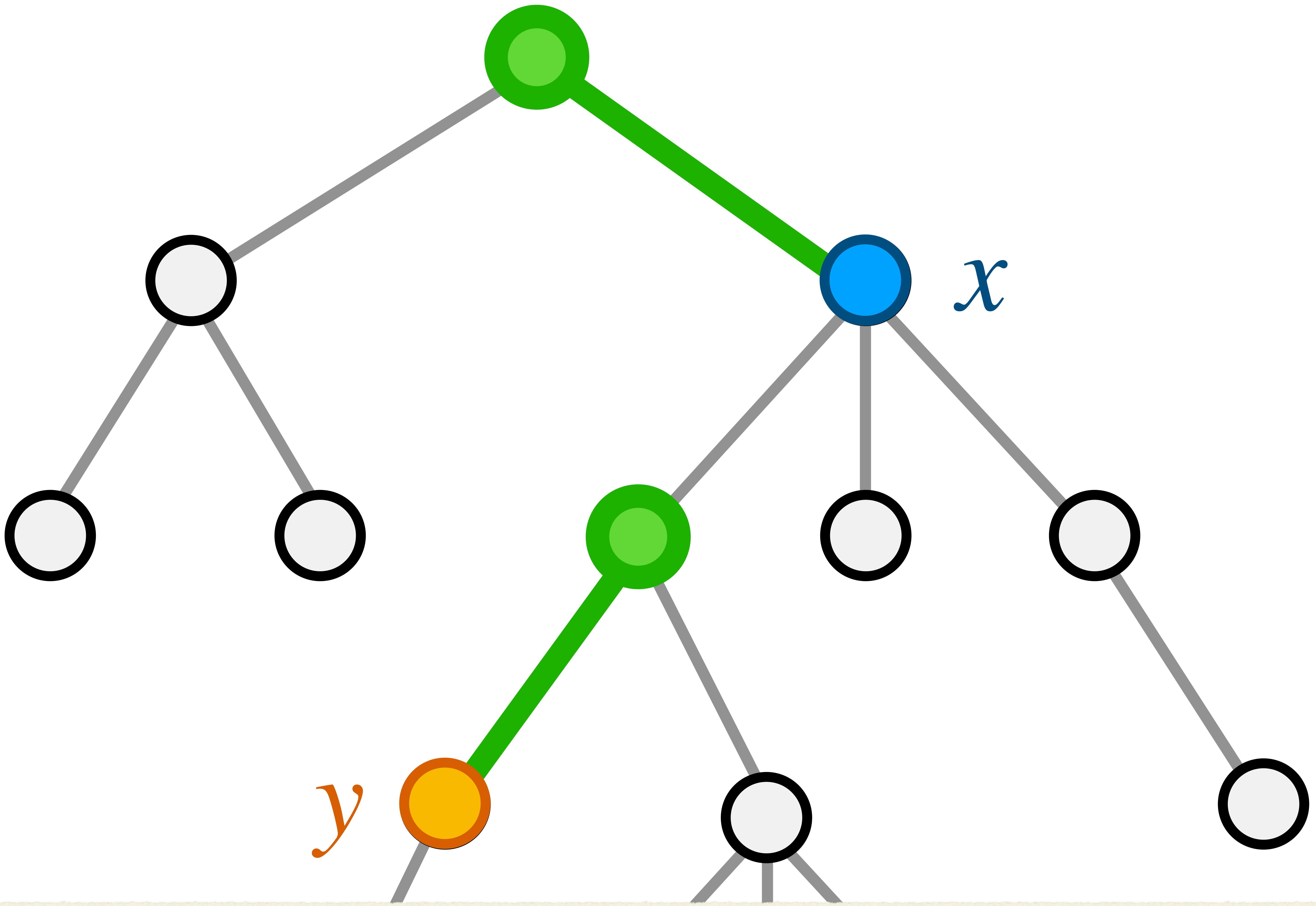
# Quiz

**Alice:** $x \in V$

**Bob:** $y \in V$

Does $x \sim y$?

**Tree Adjacency Problem**




**2 Equality tests!** = O(1) cost

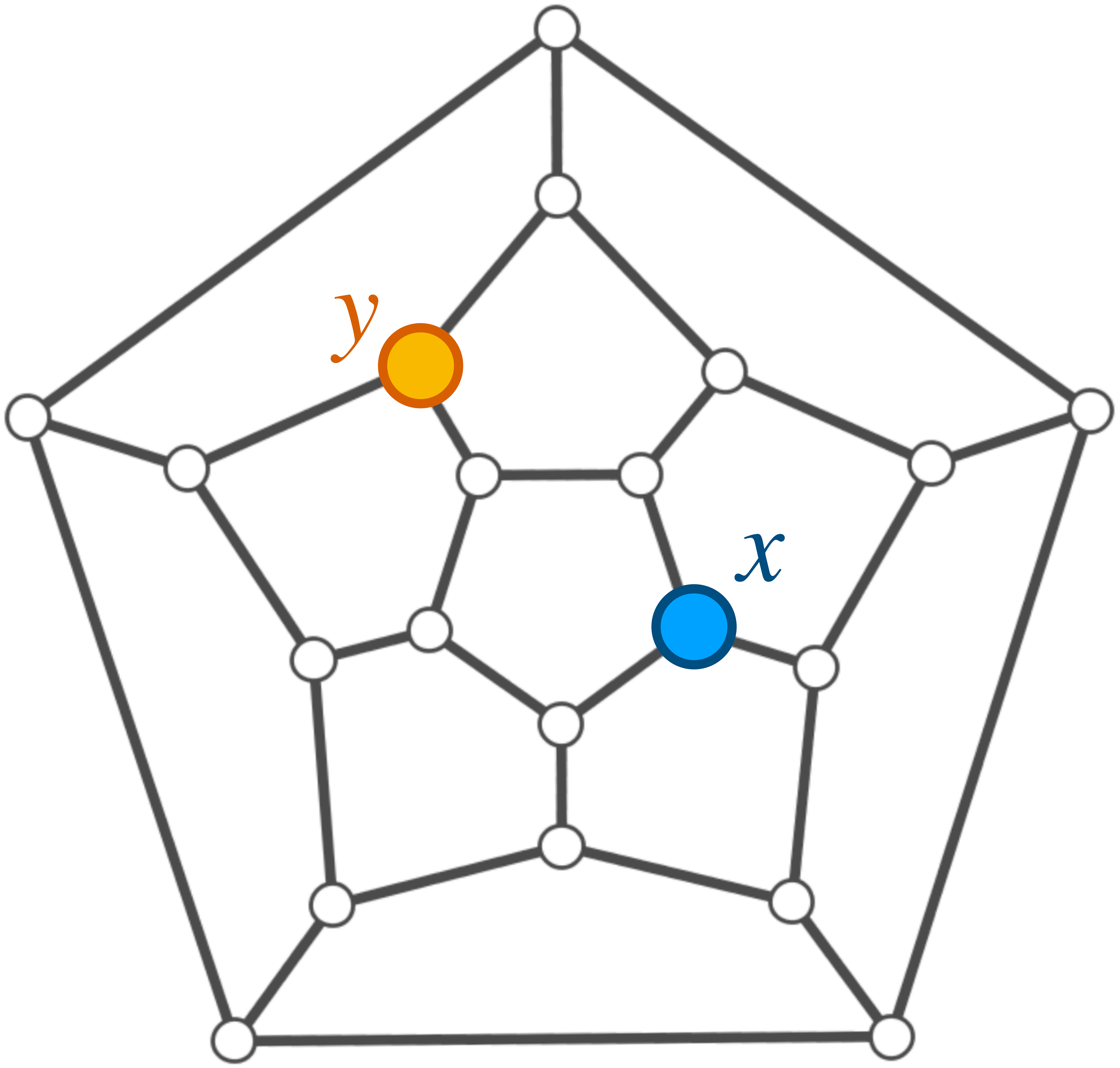
# Quiz 2

**Alice:** $x \in V$

**Bob:** $y \in V$

Does $x \sim y$?

**Planar Adjacency Problem**

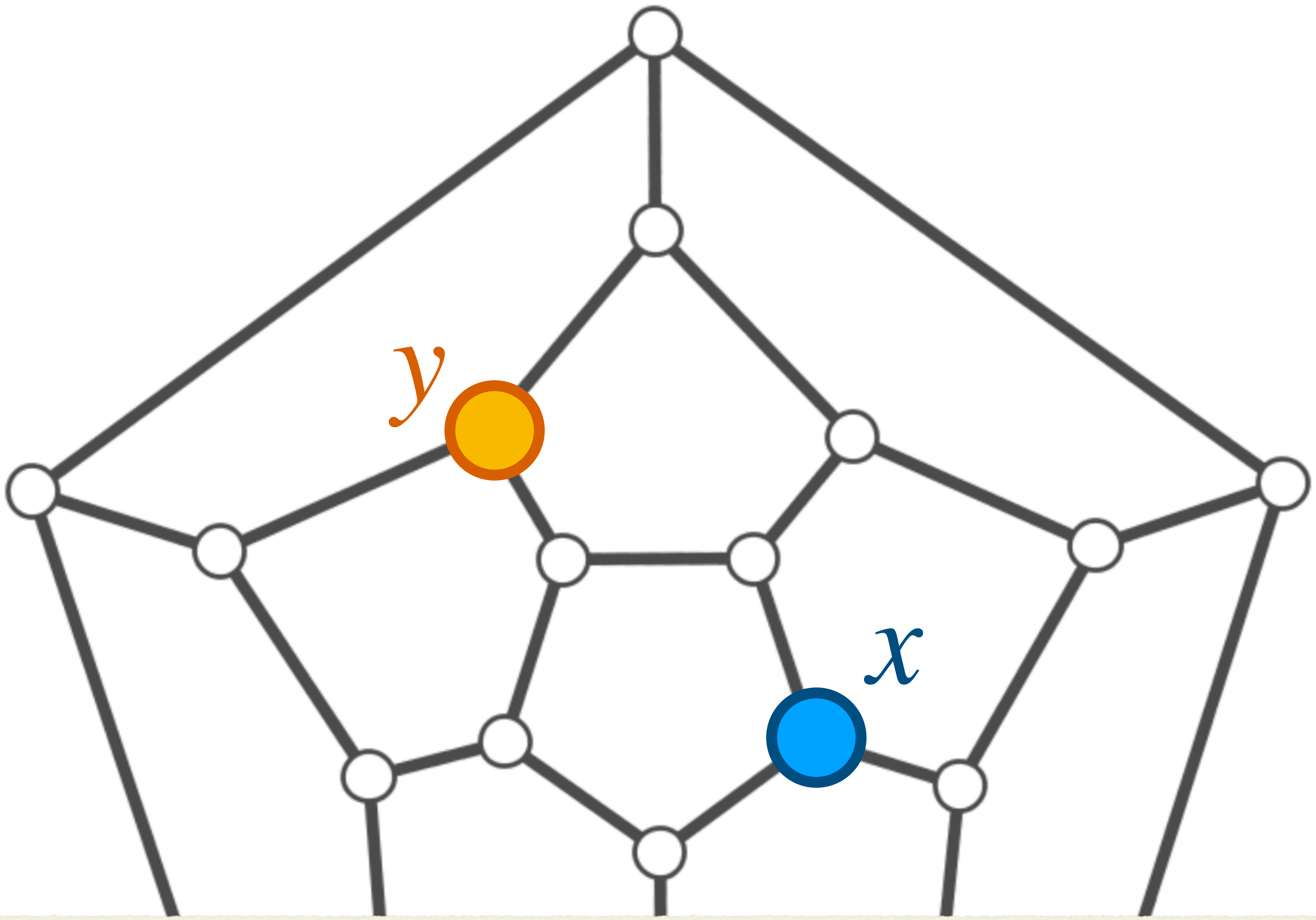


# Quiz 2

**Alice:** $x \in V$

**Bob:** $y \in V$

Does $x \sim y$?



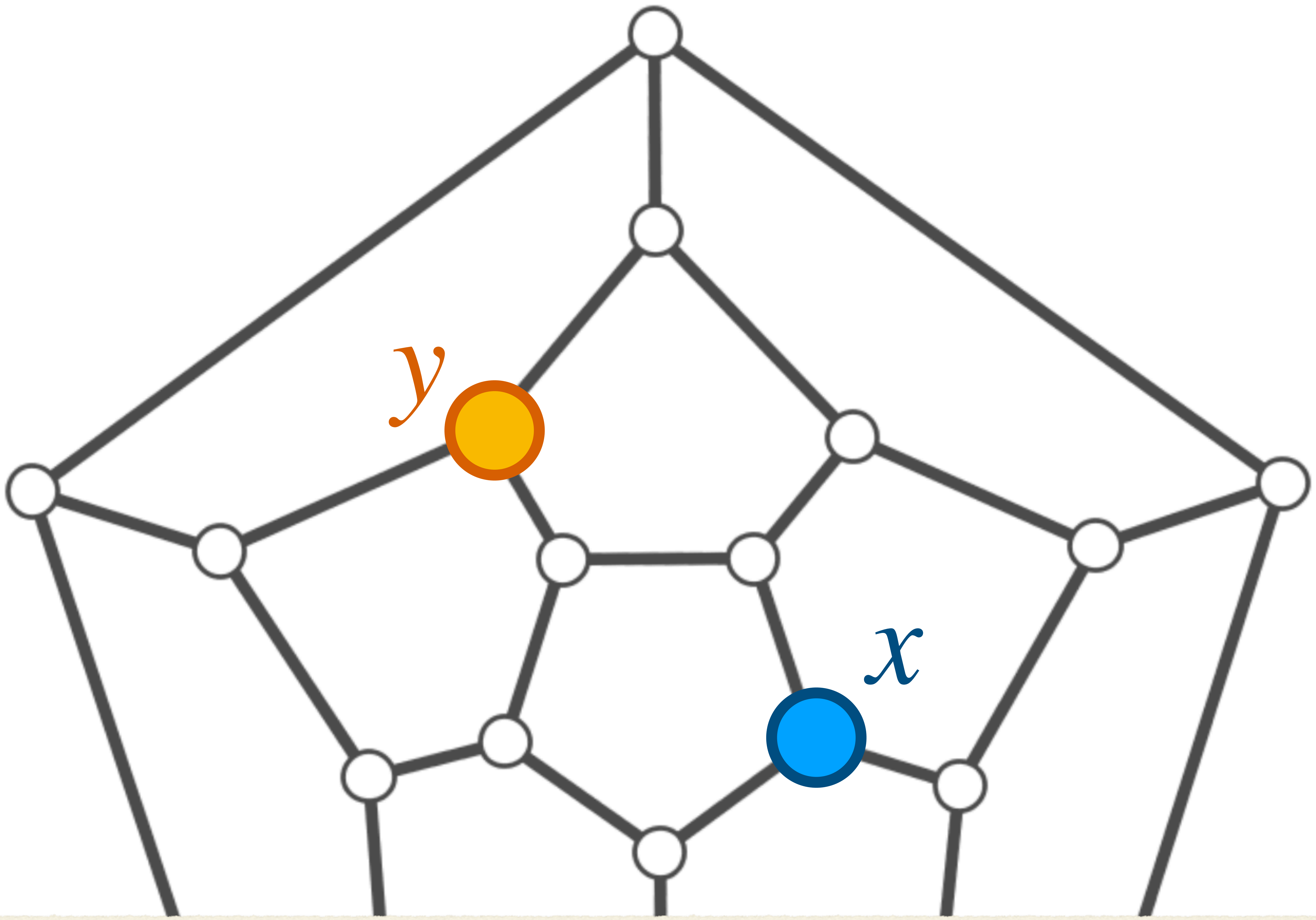

**Planar Adjacency Problem**

Planar graph is union of **3 forests**

# Quiz 2

**Alice:** $x \in V$
**Bob:** $y \in V$

Does $x \sim y$?




**Planar Adjacency Problem**

Planar graph is union of **3 forests**
→ Run **Tree Adjacency** thrice

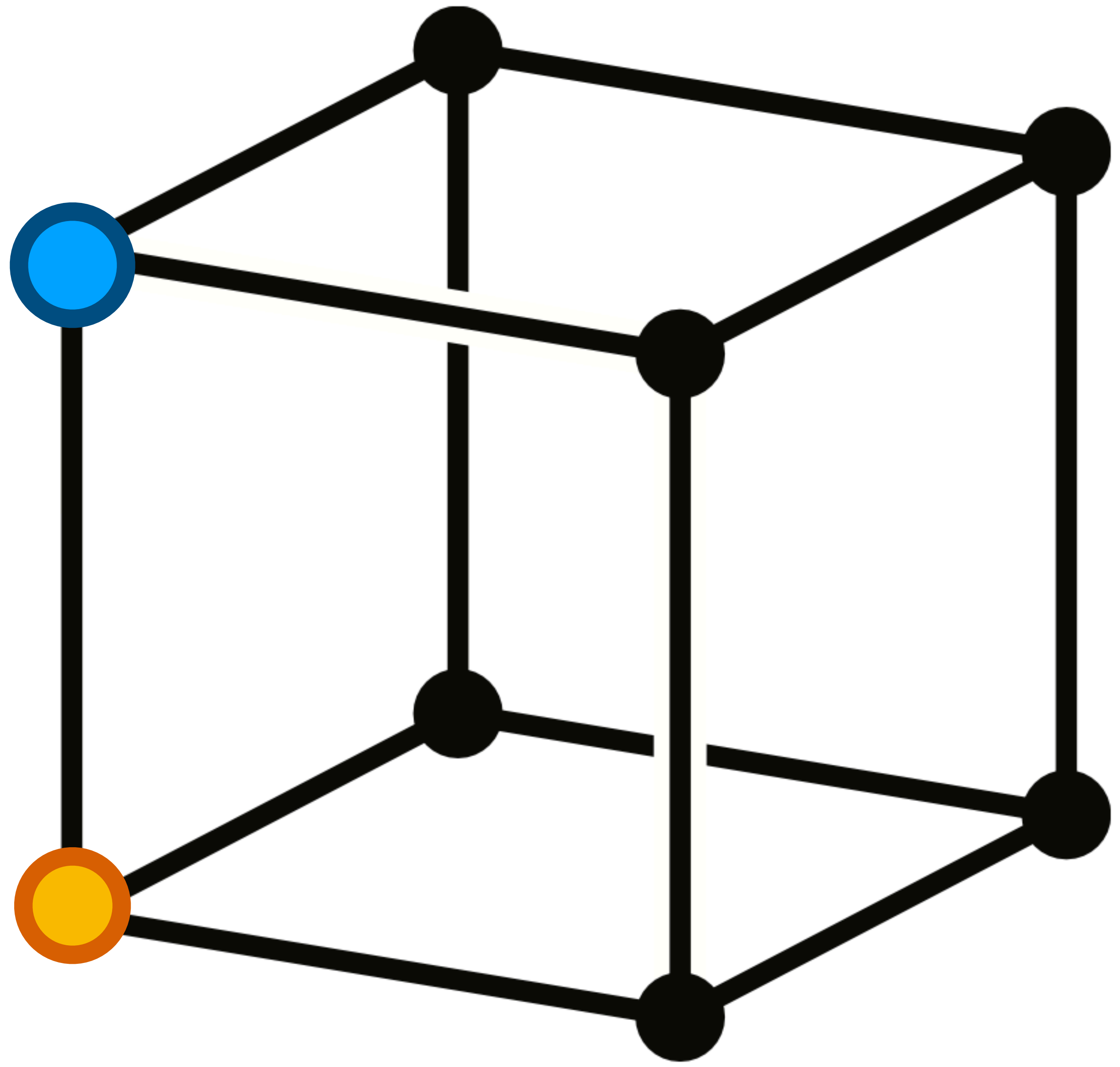
# Quiz 3

**Alice:** 001110**0**01010100010010101010010

**Bob:** 001110**1**01010100010010101010010

↑

Differ in **one** coordinate?



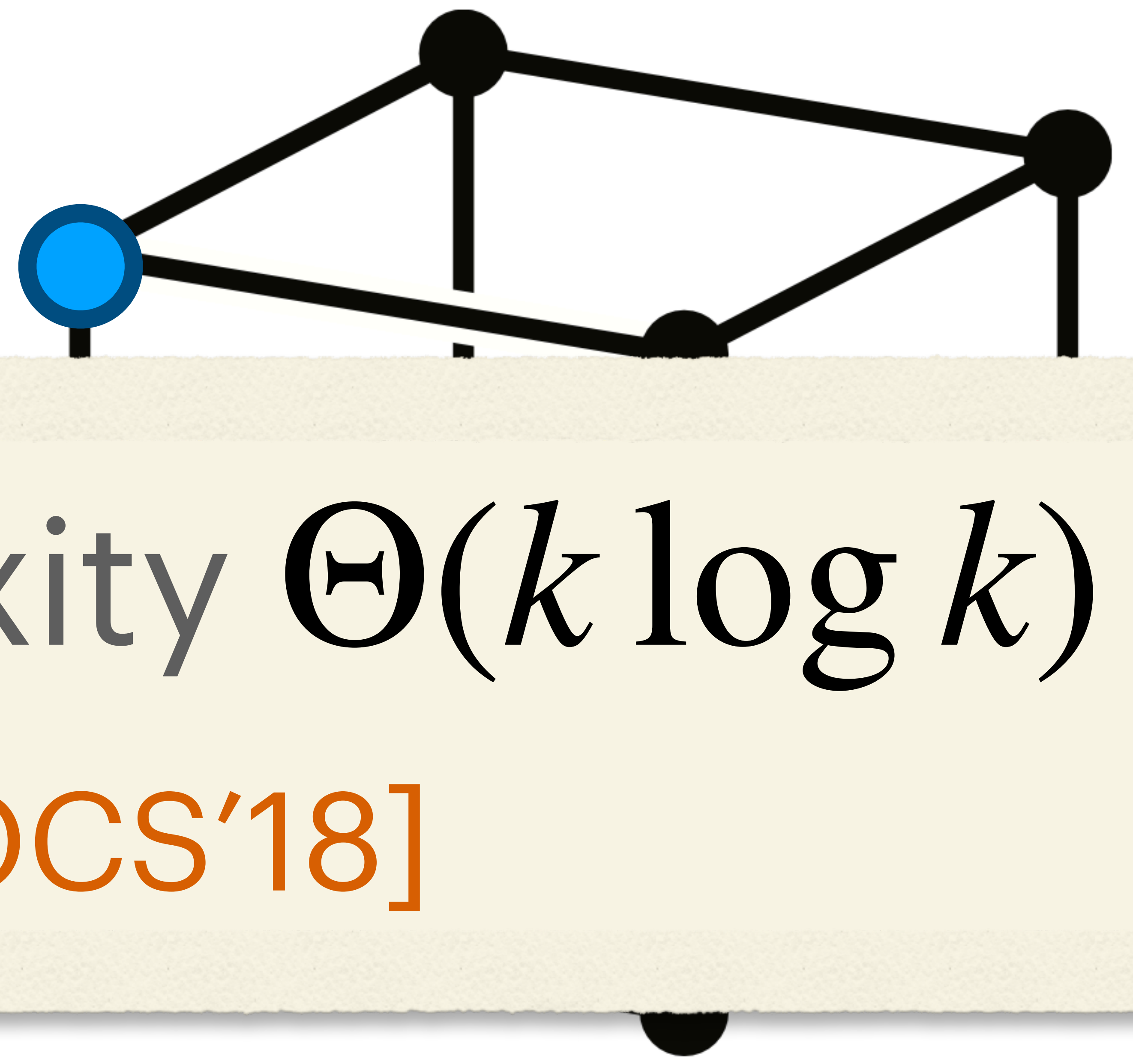
**1-Hamming Distance Problem**

# Quiz 3

**Alice:** 0011100**0**01010100010010101010010

**Bob:** 0011100**1**01010100010010101010010

Differ in **one** coordinate?



**1-Hamming Distance Problem**

k-HD has complexity $\Theta(k \log k)$

[Saglam, FOCS'18]

# Large alphabet?

**Alice:** `ATGCGATA`

**Bob:** `ATGTGATA`

**Large Alphabet 1-HD Problem**

# Large alphabet?

**Alice:** ATGCGATA

→ 1000 0100 0010 0001 0010 1000 0100 1000

**Bob:** ATGTGATA

→ 1000 0100 0010 0100 0010 1000 0100 1000

Large Alphabet
1-HD Problem

# Large alphabet?

**Alice:** ATGCGATA

→ 1000 0100 0010 0001 0010 1000 0100 1000

**Bob:** ATGTGATA

→ 1000 0100 0010 0100 0010 1000 0100 1000

Differ in **two** coordinates?

**Large Alphabet 1-HD Problem**

# Large alphabet?

**Alice:** ATGCGATA

→ 1000 0100 0010 0001 0010 1000 0100 1000

**Bob:** ATGTGATA

→ 1000 0100 0010 0100 0010 1000 0100 1000

Differ in **two** coordinates?

→ Run **2-HD** protocol

**Large Alphabet 1-HD Problem**

# Non-example

**Alice:** $x \in [n]$

**Bob:** $y \in [n]$

Does $x \geq y$?

**Greater-Than Problem**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

# Non-example

**Alice:** $x \in [n]$

**Bob:** $y \in [n]$

Does $x \geq y$?

**Greater-Than Problem**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

Cost $\Theta(\log \log n)$ [BW15, Vio15]

Note VC = O(1) and $\text{rk}_{\pm}$ = O(1)

Enough examples… Next:

# Structure theory

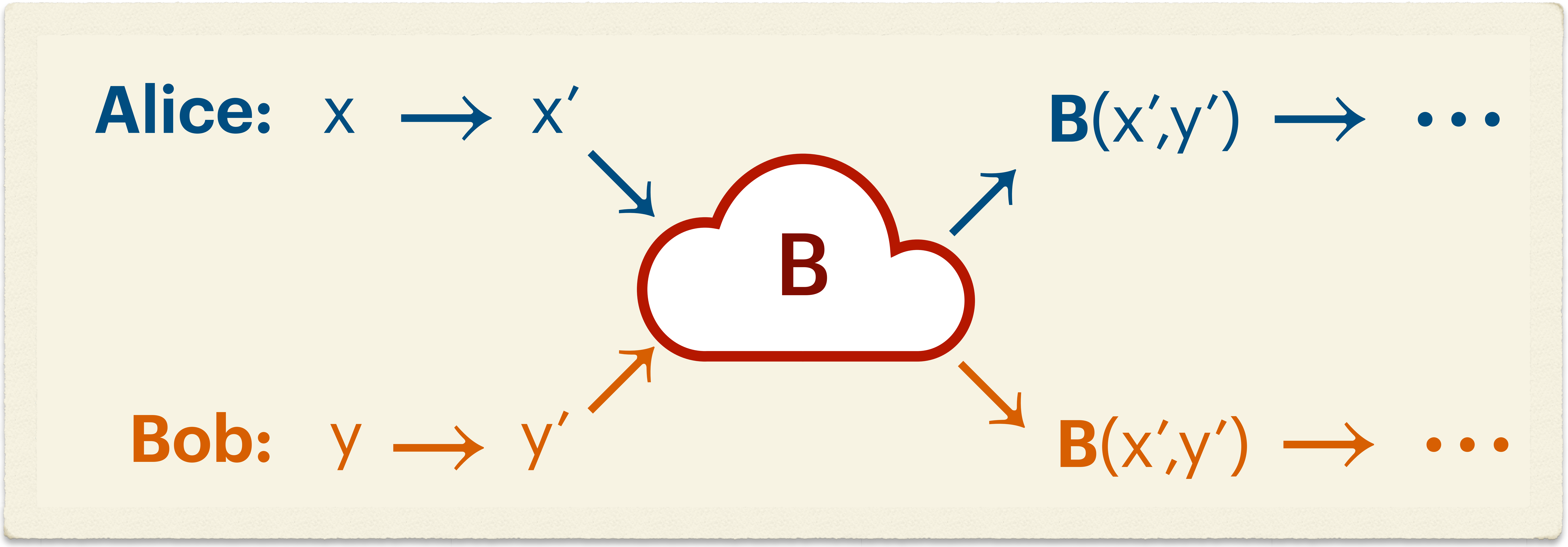
# Reductions

Planar Adjacency $\leq$ Tree Adjacency $\leq$ Equality

# Reductions

Planar Adjacency $\leq$ Tree Adjacency $\leq$ Equality

**A $\leq$ B :** **A** can be solved deterministically by making **O(1) oracle** calls to **B**




**A ≤ B :** **A** can be solved deterministically by making **O(1) oracle** calls to **B**

# Reductions

Planar Adjacency $\leq$ Tree Adjacency $\leq$ Equality

**A** $\leq$ **B** : **A** can be solved deterministically by making **O(1) oracle** calls to **B**

# Reductions

Planar Adjacency ≡ Tree Adjacency ≡ Equality

**A** $\leq$ **B** : **A** can be solved deterministically by making **O(1) oracle** calls to **B**

# Reductions

**Equality** $\leq$ **1-Hamming Distance**

**A ≤ B :** **A** can be solved deterministically by making **O(1) oracle** calls to **B**

# Reductions

Equality $<$ 1-Hamming Distance

[HHH'22, HWZ'22]

**A ≤ B :** **A** can be solved deterministically by making **O(1) oracle** calls to **B**

# Infinite hierarchy

[FHHH, STOC'24]

Equality $<$ 1-HD $<$ 2-HD $<$ 3-HD $< \cdots$

# Infinite hierarchy

[FHHH, STOC'24]

Equality < 1-HD < 2-HD < 3-HD < ⋯

(+ no single complete problem)

# Infinite hierarchy

[FHHH, STOC'24]

Equality $<$ 1-HD $<$ 2-HD $<$ 3-HD $<$ $\cdots$

(+ no single complete problem)

## Is this everything?

*Does every O(1)-cost problem reduce to k-HD?*

[HHH22b, HWZ22, HHH22a, EHK22, HHP+22, HZ24, HH24, FHHH24]

# Main result

**New O(1)-cost problem**

that does not reduce to k-HD

# Main result

**New O(1)-cost problem**

that does not reduce to k-HD

**Bonus:** Fun coding theory lemma

# New problem

# {4,4}-Hamming Distance

```
01100110111110
10001001111111
01000010101011
01010111000010
10011100101111
10010111001001
11100000000000
```

**Alice:** $X \in \{0,1\}^{n \times n}$

```
01100110111110
10001001111111
00001010100111
01010111000010
10011100101111
10110101101000
11100000000000
```

**Bob:** $Y \in \{0,1\}^{n \times n}$

# {4,4}-Hamming Distance

| | | |
|---|---|---|
| 01100110111110 | = | 01100110111110 |
| 10001001111111 | = | 10001001111111 |
| 0**1**00**0**01010**10**11 | **dist: 4** | 0**0**00**1**01010**01**11 |
| 01010111000010 | = | 01010111000010 |
| 10011100101111 | = | 10011100101111 |
| 10**0**101**1**1**0**0100**1** | **dist: 4** | 10**1**101**0**1**1**0100**0** |
| 11100000000000 | = | 11100000000000 |

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

# {4,4}-Hamming Distance

**O(1)-cost protocol:**

Check ∃two unequal rows

Choose random $A \subseteq [n]$

Check $\mathrm{dist}(X_A, Y_A) = 4$

Check $\mathrm{dist}(X_{\bar{A}}, Y_{\bar{A}}) = 4$

| | | |
|---|---|---|
| 0110011011111 0 | = | |
| 1000100111111 1 | = | |
| 0**1**00**0**0101**10**11 | dist: 4 | |
| 0101011100001 0 | = | 0101011100001 0 |
| 1001110010111 1 | = | 1001110010111 1 |
| 10**0**101**11**0**0**100**1** | dist: 4 | 10**1**101**0**1**1**01000**0** |
| 1110000000000 0 | = | 1110000000000 0 |

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

## {4,4}-Hamming Distance

```
0110011011111 0
1000100111111 1
0100001010101 1
0101011100001 0
1001110010111 1
1001011100100 1
1110000000000 0
```

=
=
dist: 4
=
=
dist: 4
=

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

## O(1)-cost protocol:

Check $\exists$two unequal rows

Choose random $A \subseteq [n]$

Check $\mathrm{dist}(X_A, Y_A) = 4$

Check $\mathrm{dist}(X_{\bar{A}}, Y_{\bar{A}}) = 4$

## Main result:

{4,4}-HD $\not\leq$ k-HD

# Why {4,4}?

# {1,1}-Hamming Distance

| | | |
|---|---|---|
| 01100110111110 | = | 01100110111110 |
| 10001001111111 | = | 10001001111111 |
| 0000**0**010100111 | **dist: 1** | 0000**1**010100111 |
| 01010111000010 | = | 01010111000010 |
| 10011100101111 | = | 10011100101111 |
| 1001011100100**1** | **dist: 1** | 1001011100100**0** |
| 11100000000000 | = | 11100000000000 |

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

# {1,1}-Hamming Distance

**Oracle protocol:**

Check ∃ two unequal rows

Check $\mathrm{dist}(X, Y) = 2$

| Alice's matrix | | Bob's matrix |
|---|---|---|
| 01100110111110 | = | |
| 10001001111111 | = | 10001001111111 |
| 00000010100111 | dist: 1 | 00001010100111 |
| 01010111000010 | = | 01010111000010 |
| 10011100101111 | = | 10011100101111 |
| 10010111001001 | dist: 1 | 10010111001000 |
| 11100000000000 | = | 11100000000000 |

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

# {2,2}-Hamming Distance

| Alice's matrix | | Bob's matrix |
|---|---|---|
| 01100110111110 | = | 01100110111110 |
| 10001001111111 | = | 10001001111111 |
| 00000010100111 | **dist: 2** | 00001010000111 |
| 01010111000010 | = | 01010111000010 |
| 10011100101111 | = | 10011100101111 |
| 10010111001001 | **dist: 2** | 11010111001000 |
| 11100000000000 | = | 11100000000000 |

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

# {2,2}-Hamming Distance

**Oracle protocol:**

Check ∃ two unequal rows

Check $\mathrm{dist}(X, Y) = 4$

# Must be {2,2} or {1,3}

Let $(x, y)$ be row parities

Check $x = y$

```
011001101111110     =
100010011111111     =
000000101001111     dist:
010101110000010     =
100111001011111     =                 100111001011111
100101110010001     dist: 2           110101110010000
111000000000000     =                 111000000000000
```

**Alice:** $X \in \{0,1\}^{n \times n}$

**Output "YES"**

**Bob:** $Y \in \{0,1\}^{n \times n}$

# Coding lemma

Towards $\{4,4\}$-HD $\not\preceq$ k-HD

**Definition**

Let $f\colon \{2,4,6\} \to \mathbb{N}$

Call $E\colon \{0,1\}^n \to \{0,1\}^m$ an ***f*-code** iff

$$\mathrm{dist}(E(x), E(y)) = f(\mathrm{dist}(x, y))$$

whenever $f$ is defined

**Definition**

Let $f\colon \{2,4,6\} \to \mathbb{N}$

Call $E\colon \{0,1\}^n \to \{0,1\}^m$ an **$f$-code** iff

$$\mathrm{dist}(E(x), E(y)) = f(\mathrm{dist}(x, y))$$

whenever $f$ is defined

**Lemma**

If there exists $f$-codes for infinitely many $n$, then

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Definition**

Let $f\colon \{2,4,6\} \to \mathbb{N}$

Call $E\colon \{0,1\}^n \to \{0,1\}^m$ an ***f*-code** iff

$$\text{dist}(E(x), E(y)) = f(\text{dist}(x, y))$$

whenever $f$ is defined

**Lemma**

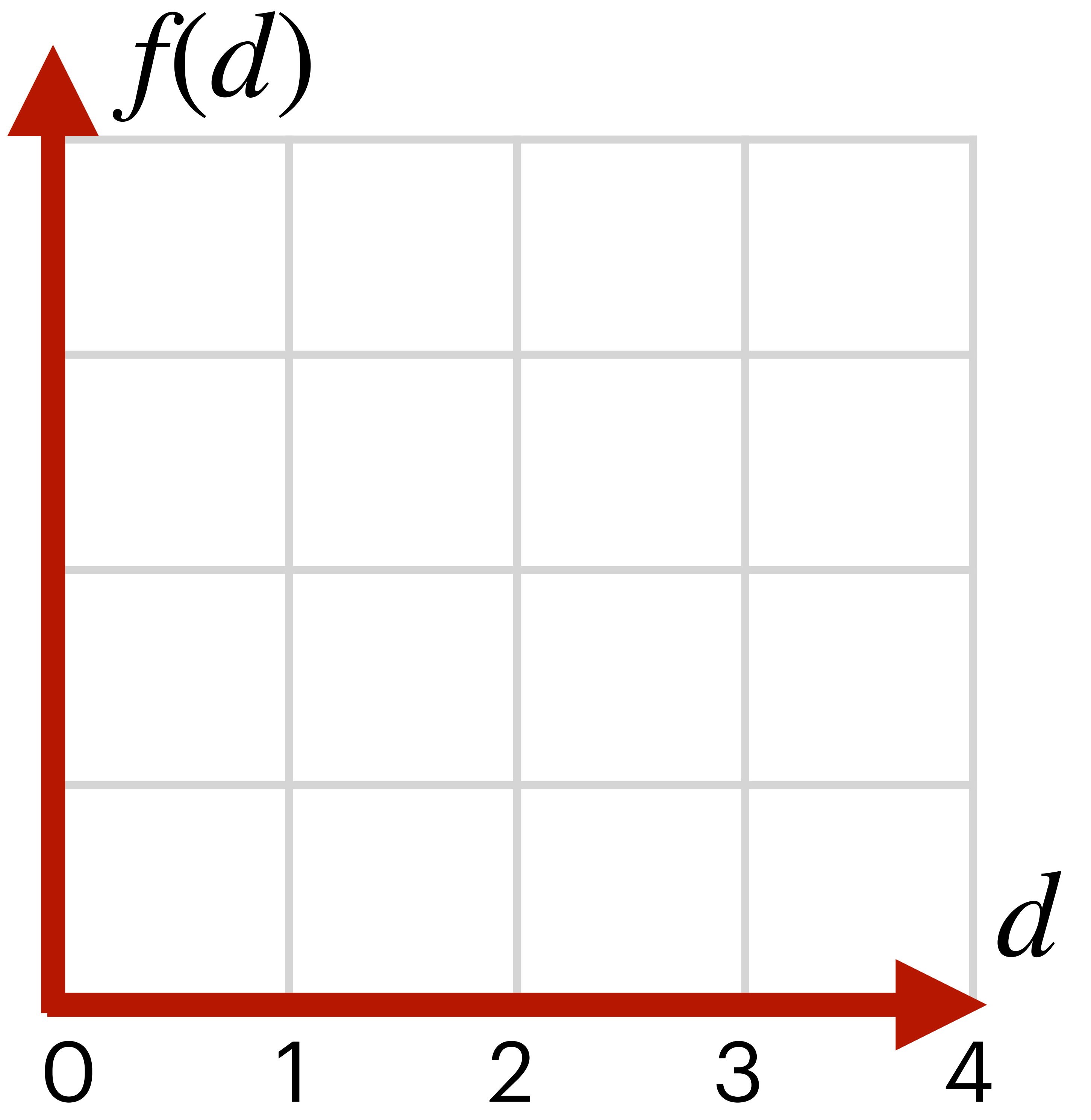
If there exists $f$-codes for infinitely many $n$, then

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Examples**

$E(x) = xx$

$m = 2n$



**Definition**

Let $f\colon \{2,4,6\} \to \mathbb{N}$

Call $E\colon \{0,1\}^n \to \{0,1\}^m$ an **$f$-code** iff

$$\mathrm{dist}(E(x), E(y)) = f(\mathrm{dist}(x, y))$$

whenever $f$ is defined

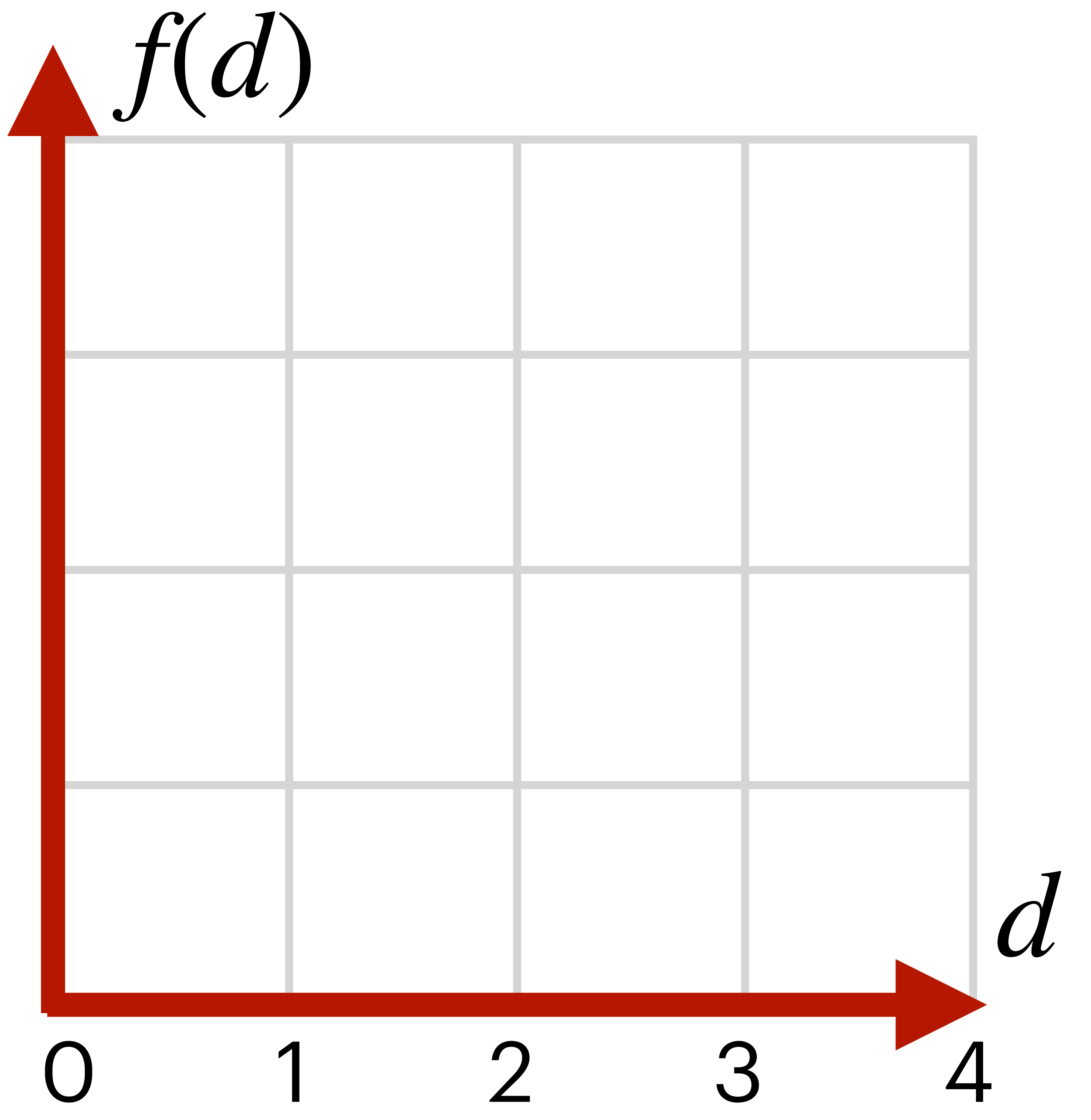
**Lemma**

If there exists $f$-codes for infinitely many $n$, then

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Examples**

$$E(x) = e_x$$
$$m = 2^n$$



**Definition**

Let $f\colon \{2,4,6\} \to \mathbb{N}$

Call $E\colon \{0,1\}^n \to \{0,1\}^m$ an **$f$-code** iff

$$\mathrm{dist}(E(x), E(y)) = f(\mathrm{dist}(x, y))$$

whenever $f$ is defined

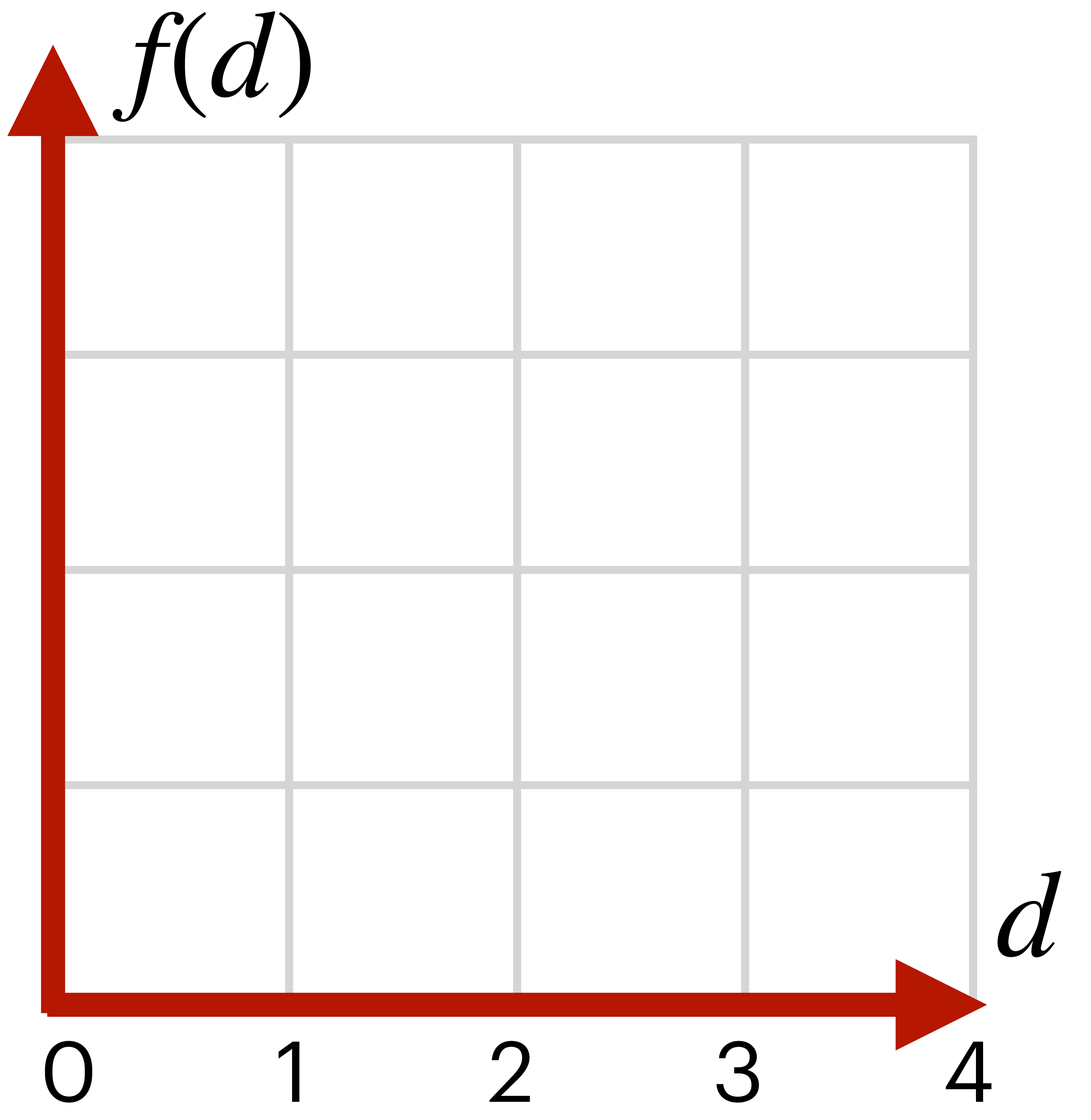
**Lemma**

If there exists $f$-codes for infinitely many $n$, then

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Examples**

$$E(x) = \bigoplus_i x_i$$
$$m = 1$$



**Main result**

**{4,4}-HD $\not\leq$ k-HD**

**Coding lemma**

Infinite $f$-code family has

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Main result**

**{4,4}-HD $\not\leq$ k-HD**

$\Downarrow \quad \Uparrow$

**Coding lemma**

Infinite $f$-code family has

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

**Main result**

$$\{4,4\}\text{-HD} \not\leq \text{k-HD}$$

*Easy* $\Downarrow$ $\Uparrow$ *Ramsey theory*

**Coding lemma**

Infinite $f$-code family has

$$f(4) = \frac{1}{2}(f(2) + f(6))$$

$\exists$ non-affine $f$-code $E \implies$ {4,4}-HD $\leq$ k-HD

# $\exists$ non-affine $f$-code $E \implies$ {4,4}-HD $\leq$ k-HD

**Alice**

$X \in \{0,1\}^{n \times n}$

**Bob**

$Y \in \{0,1\}^{n \times n}$

**Oracle protocol:**

Check $\exists$ two unequal rows

Check $\mathrm{dist}(X, Y) = 8$

# {1,7}, {2,6}, {3,5}, or {4,4}

Check row parities

# {2,6} or {4,4}

Encode rows by $E$, check:

$\mathrm{dist}(E(X), E(Y)) = 2 \cdot f(4)$

# $\exists$ non-affine $f$-code $E \implies$ {4,4}-HD $\leq$ k-HD

**Alice**

$X \in \{0,1\}^{n\times n}$

$X$:

| $X_1$ |
| --- |
| $X_2$ |
| $X_3$ |
| $X_4$ |
| $X_5$ |

**Bob**

$Y \in \{0,1\}^{n\times n}$

$Y$:

| $Y_1$ |
| --- |
| $Y_2$ |
| $Y_3$ |
| $Y_4$ |
| $Y_5$ |

**Oracle protocol:**

Check $\exists$ two unequal rows

Check $\mathrm{dist}(X, Y) = 8$

# {1,7}, {2,6}, {3,5}, or {4,4}

Check row parities

# {2,6} or {4,4}

Encode rows by $E$, check:

$\mathrm{dist}(E(X), E(Y)) = 2 \cdot f(4)$

# $\exists$ non-affine $f$-code $E \implies$ {4,4}-HD $\leq$ k-HD

**Alice**

$X \in \{0,1\}^{n \times n}$

$E(X)$:

$E(X_1)$
$E(X_2)$
$E(X_3)$
$E(X_4)$
$E(X_5)$

**Bob**

$Y \in \{0,1\}^{n \times n}$

$E(Y)$:

$E(Y_1)$
$E(Y_2)$
$E(Y_3)$
$E(Y_4)$
$E(Y_5)$

**Oracle protocol:**

Check $\exists$ two unequal rows

Check $\mathrm{dist}(X, Y) = 8$

# {1,7}, {2,6}, {3,5}, or {4,4}

Check row parities

# {2,6} or {4,4}

Encode rows by $E$, check:

$\mathrm{dist}(E(X), E(Y)) = 2 \cdot f(4)$

**Main result**

**{4,4}-HD $\not\leq$ k-HD**

$\Updownarrow$

**Coding lemma**

Infinite $f$-code family has

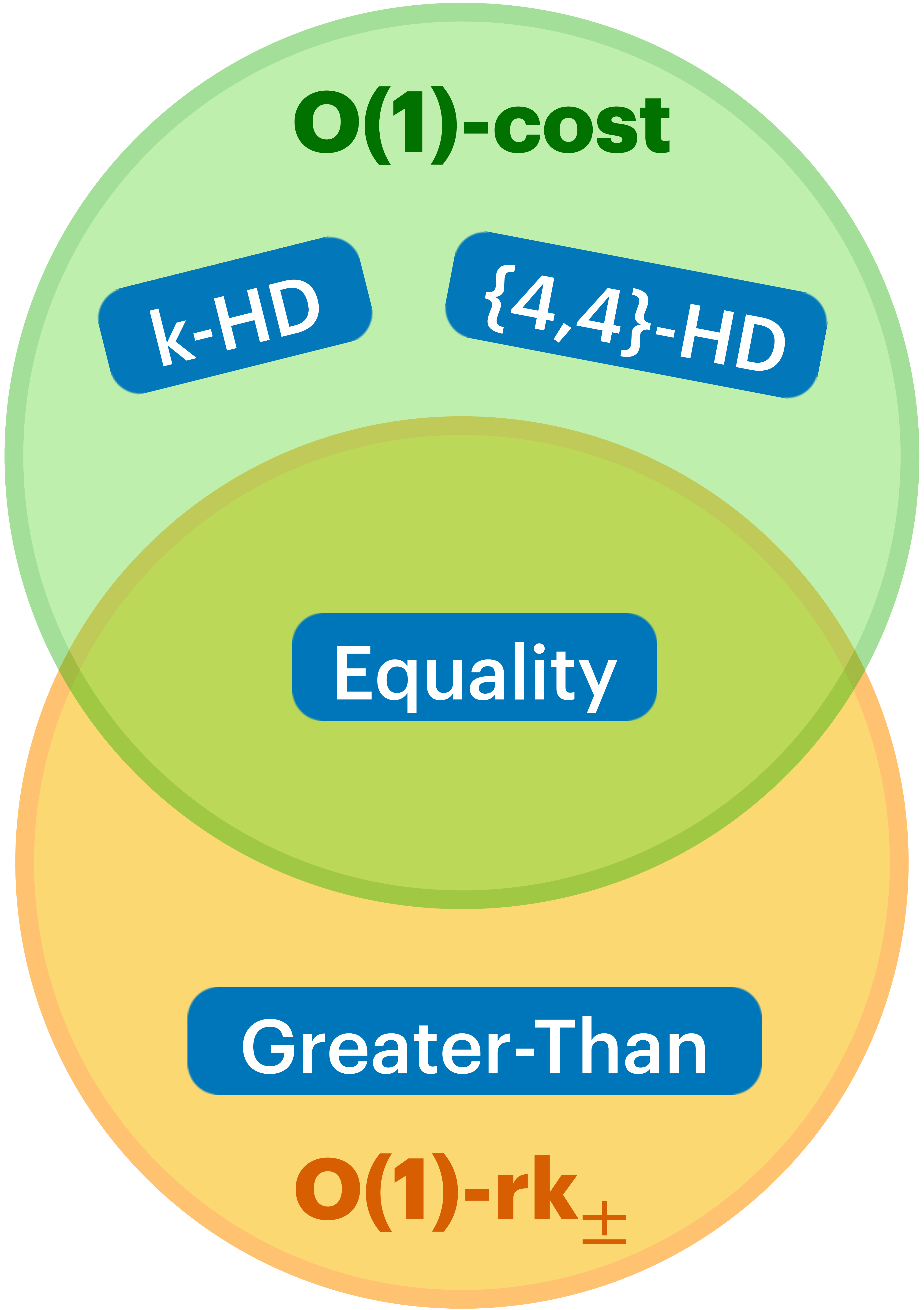
$$f(4) = \frac{1}{2}(f(2) + f(6))$$

Next for O(1)-cost?

# Open problems

# Open problems

- More examples!
  - Complete hierarchy?
  - Characterisation?
- Structure of O(1)-cost
  - Monochrome rectangles?
  - One-sided error?
- Intersection classes
- O(1)-$rk_{\pm}$ closed under OR?



SIGACT News Complexity Theory Column, March 2024

# Structure in Communication Complexity and Constant-Cost Complexity Classes

Hamed Hatami[1] Pooya Hatami[2]

**Abstract**

Several theorems and conjectures in communication complexity state or speculate that the complexity of a matrix in a given communication model is controlled by a related analytic or algebraic matrix parameter, e.g., rank, sign-rank, discrepancy, etc. The forward direction is typically easy as the structural implications of small complexity often imply a bound on some matrix parameter. The challenge lies in establishing the reverse direction, which requires understanding the structure of Boolean matrices for which a given matrix parameter is small or large. We will discuss several research directions that align with this overarching theme.

## 1 Introduction

In 1979, Yao [Yao79] introduced an abstract model for analyzing communication. It quickly became apparent that the applications of this elegant paradigm go far beyond the concept of communication. Many results in communication complexity have equivalent formulations in other fields that are equally natural, and the techniques developed within this framework have proven to be powerful tools applicable across various domains. Today, communication complexity is a vibrant research area with many connections across theoretical computer science and mathematics: in learning theory, circuit design, pseudorandomness, data streaming, data structures, computational complexity, computer networks, time-space trade-offs, discrepancy theory, and property testing.

In this article, we focus on the most standard framework where a communication problem is simply a *Boolean* matrix. Formally, there are two parties, often called Alice and Bob, and a communication problem is defined by a matrix $F \in \{0,1\}^{\mathcal{X}\times\mathcal{Y}}$. Alice receives a row index $x \in \mathcal{X}$, and Bob receives a column index $y \in \mathcal{Y}$. Together, they should compute the entry $F(x,y)$ by exchanging bits of information according to a previously agreed-on protocol tailored to $F$. There is

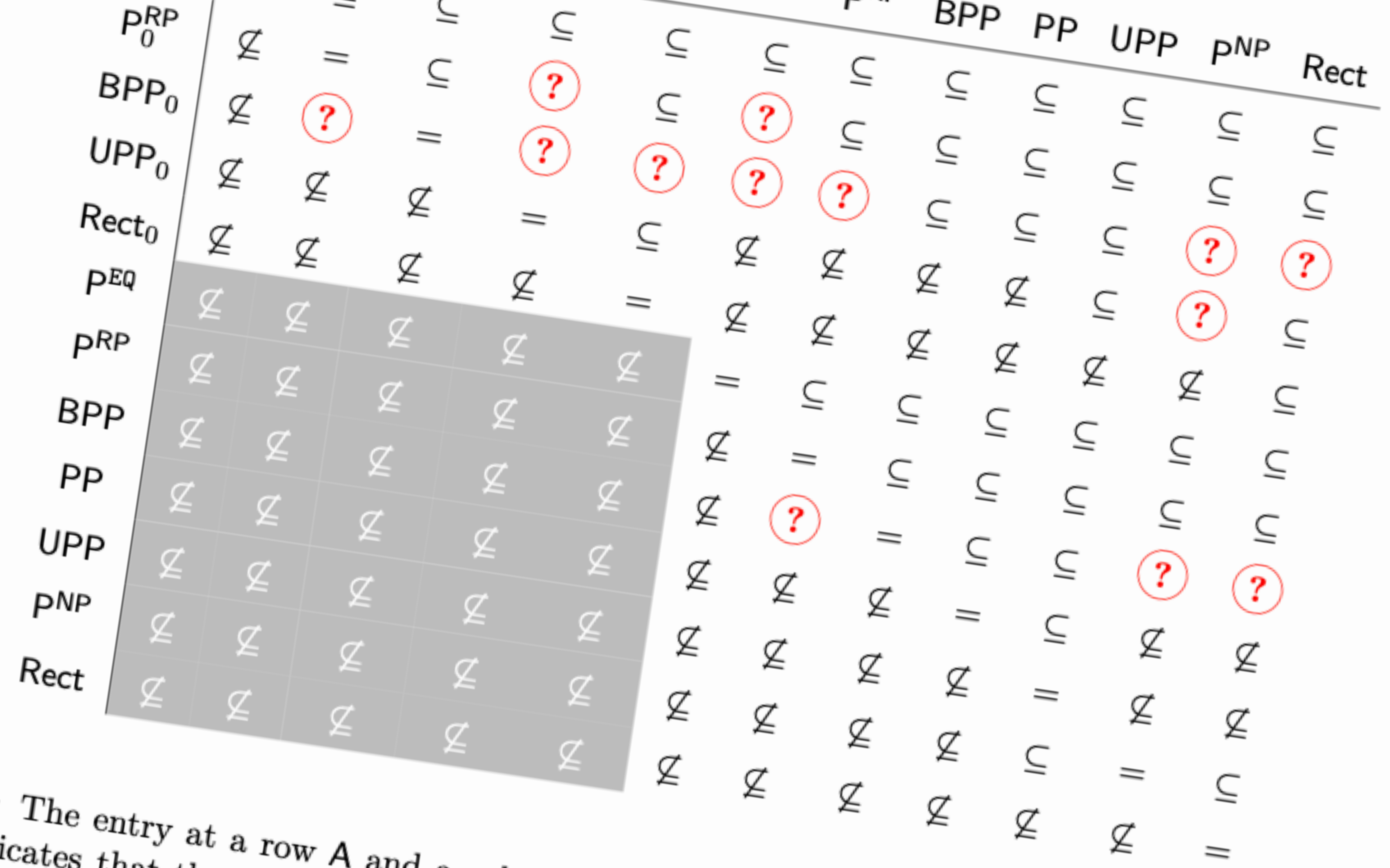


Figure 1: The entry at a row A and a column B indicates whether A ⊆ B or A ⊄ B. A question mark indicates that the relationship is unknown. The separations in grey entries follow trivially via padding.

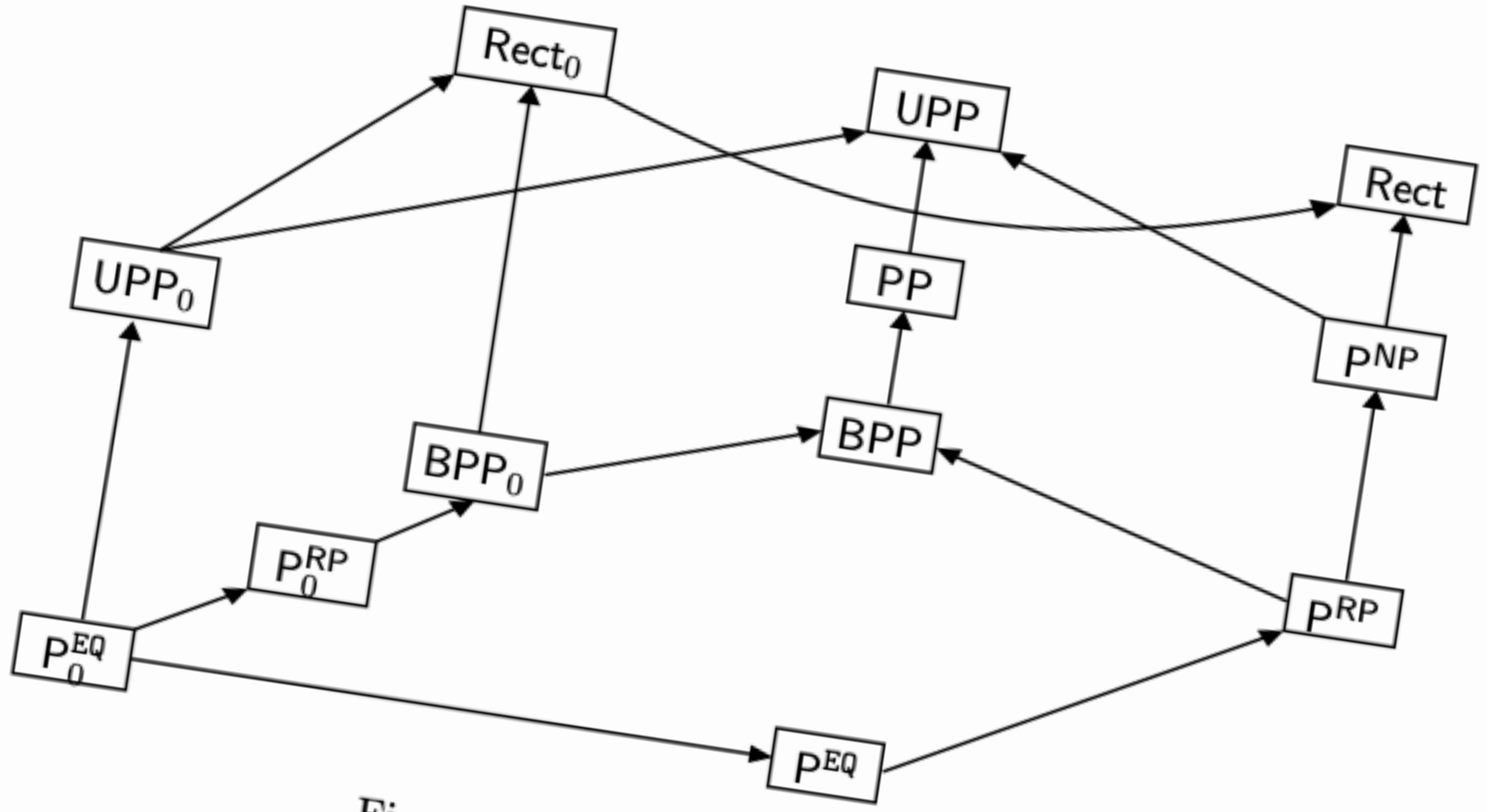



Figure 2: A → B indicates A ⊆ B.

# Thank you!