VANDERBILT UNIVERSITY School of Engineering

Discrete Structures

CS 2212

(Fall 2020)

12 – Binary Relations

Chapter 5:

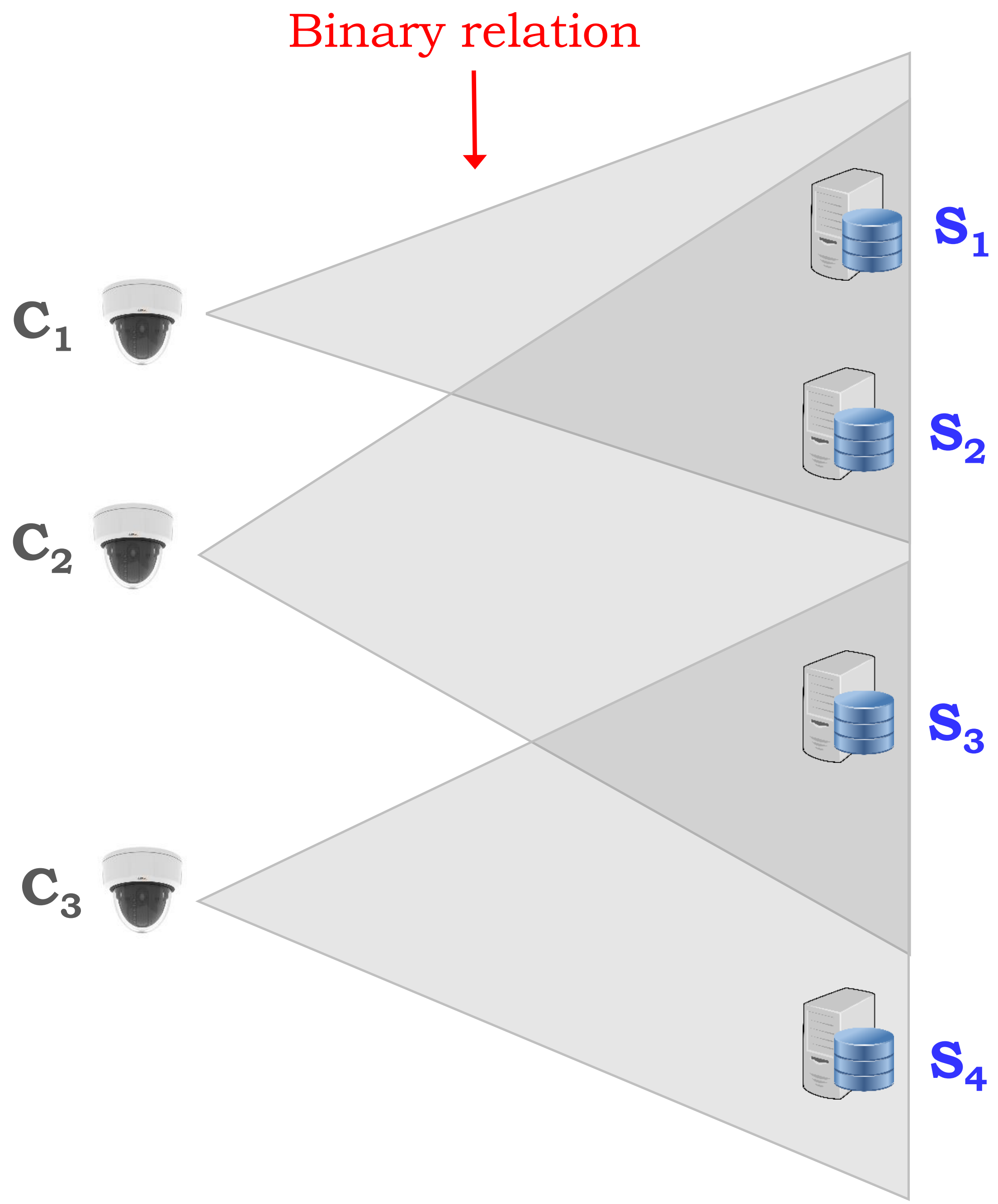
Relations

Cameras = { C_1, C_2, C_3 }

Servers = { S_1, S_2, S_3, S_4 }

We are interested in Camera-Server **“Relationship”**

Which camera **“monitors”** which server?



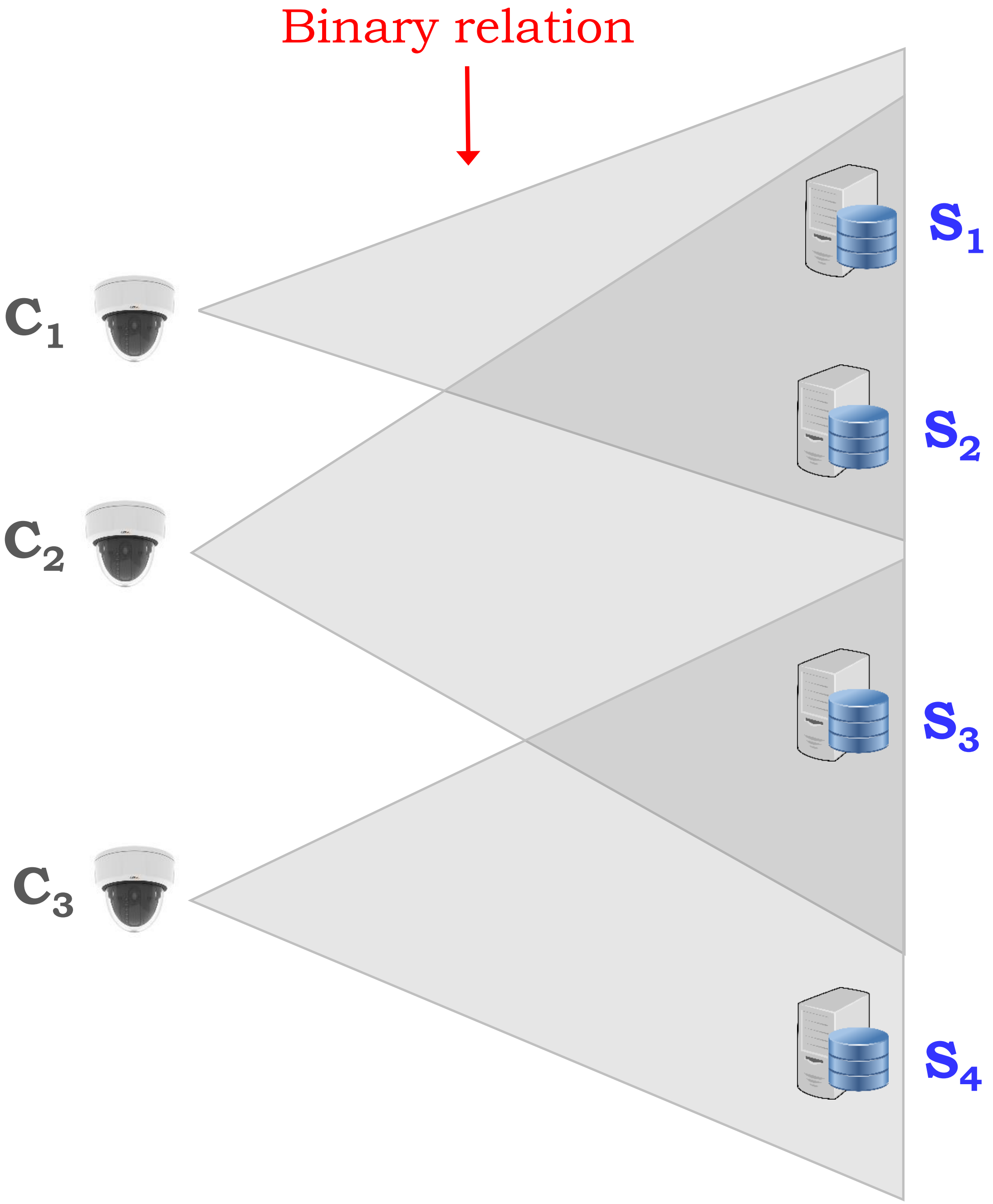
Cameras = { C_1, C_2, C_3 }

Servers = { S_1, S_2, S_3, S_4 }

We can use **(Camera, Server)** pair to denote this binary relationship.

Cameras – Servers "Relationship":

- C_1 *monitors* S_1 — (C_1, S_1)
- C_1 *monitors* S_2 — (C_1, S_2)
- C_2 *monitors* S_1 — (C_2, S_1)
- C_2 *monitors* S_2 — (C_2, S_2)
- C_2 *monitors* S_3 — (C_2, S_3)
- C_3 *monitors* S_3 — (C_3, S_3)
- C_3 *monitors* S_4 — (C_3, S_4)

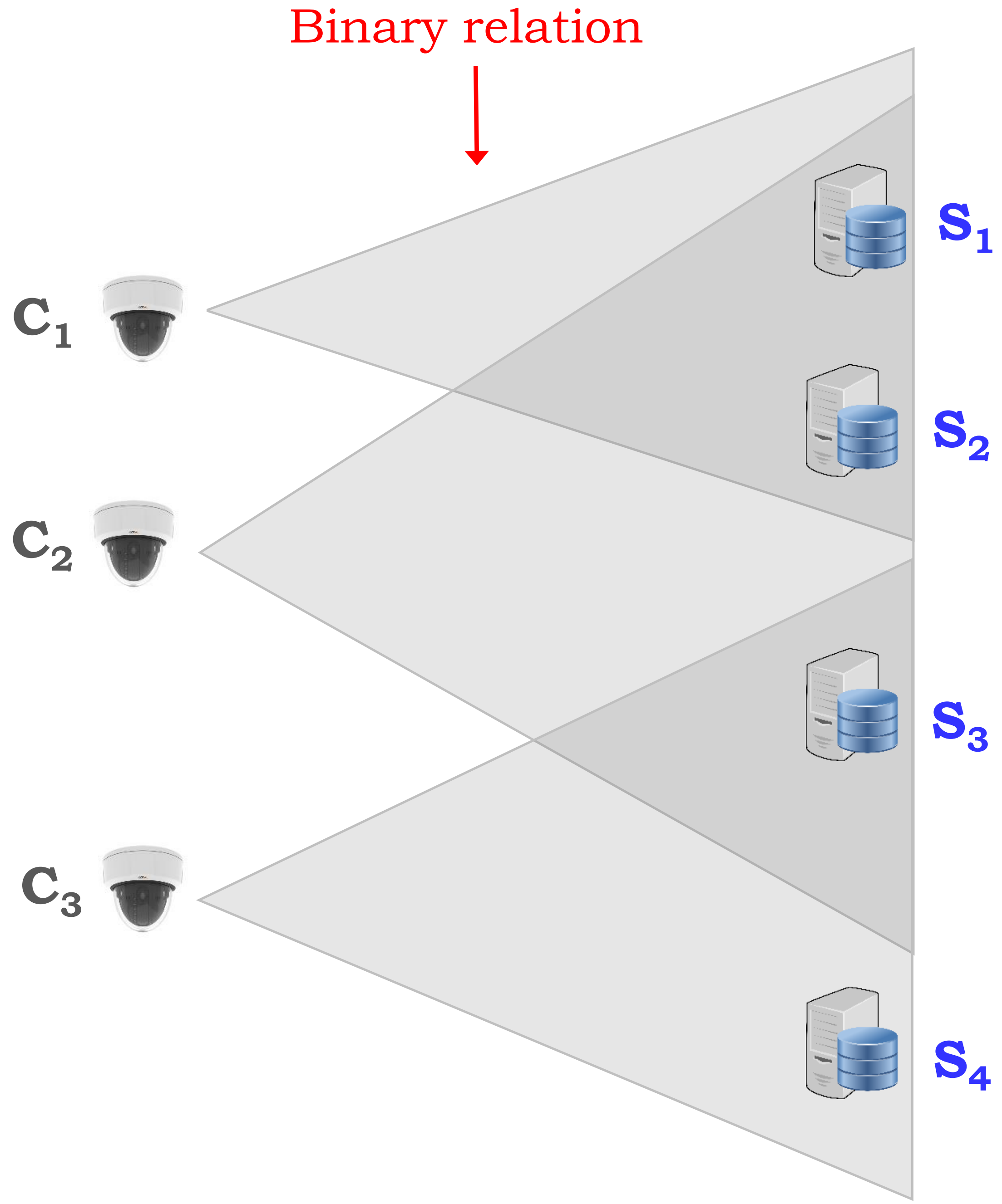


Cameras = { C_1, C_2, C_3 }

Servers = { S_1, S_2, S_3, S_4 }

So, we can describe the Camera-Server relationship by a **set of (C_i, S_j) pairs**.

$$\left\{ \begin{array}{c} (C_1, S_1) , (C_1, S_2) , \\ (C_2, S_1) , (C_2, S_2) , (C_2, S_3), \\ (C_3, S_3) , (C_3, S_4) \end{array} \right\}$$



Cameras = $\{ C_1, C_2, C_3 \}$

Servers = $\{ S_1, S_2, S_3, S_4 \}$

This (relationship) set is a subset of the **Cartesian Product** of **Cameras × Servers**, which consists of all possible pairs between these two sets.

Cameras × Servers =

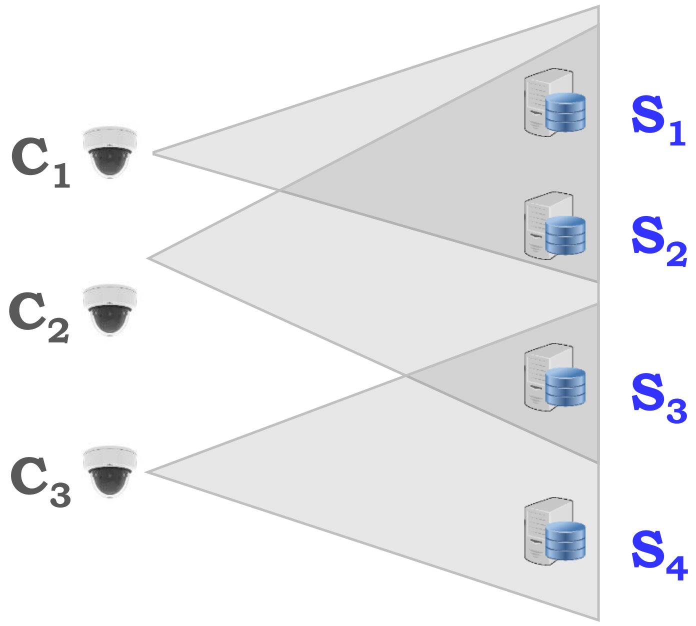
$$\left\{ \begin{array}{llll} (C_1, S_1), & (C_1, S_2), & (C_1, S_3), & (C_1, S_4), \\ (C_2, S_1), & (C_2, S_2), & (C_2, S_3), & (C_2, S_4), \\ (C_3, S_1), & (C_3, S_2), & (C_3, S_3), & (C_3, S_4) \end{array} \right\}$$

Binary Relations

$\{ (C_1, S_1), (C_1, S_2), (C_2, S_1), (C_2, S_2), (C_2, S_3), (C_3, S_3), (C_3, S_4) \}$ $\subseteq$ $\{ (C_1, S_1), (C_1, S_2), (C_1, S_3), (C_1, S_4), (C_2, S_1), (C_2, S_2), (C_2, S_3), (C_2, S_4), (C_3, S_1), (C_3, S_2), (C_3, S_3), (C_3, S_4) \}$

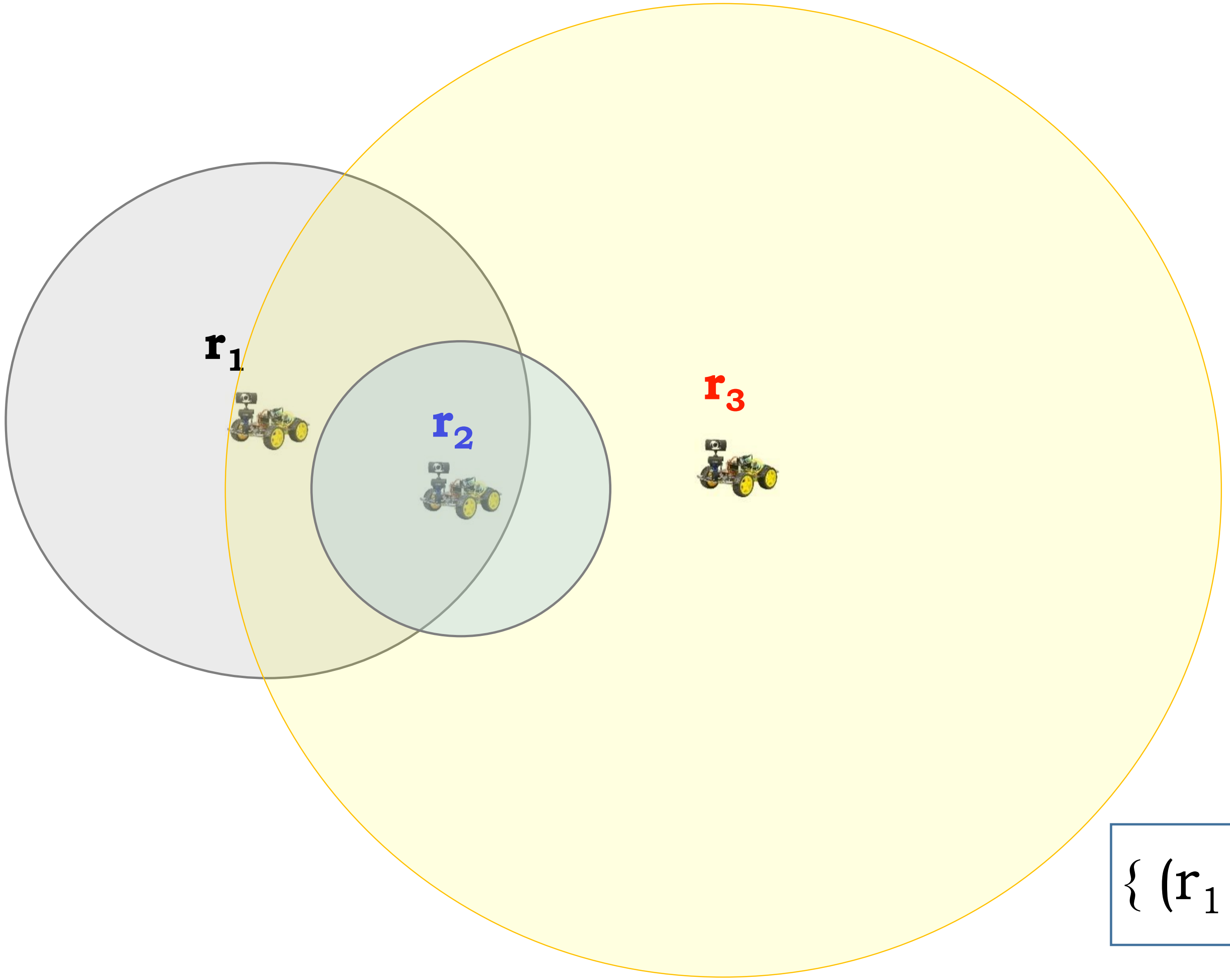
Cameras – Servers (relations) | Cameras × Servers (all possibilities)

Binary relation $\subseteq$ **Cartesian product**



A **binary relation** R between two sets A and B is a subset of A × B (the Cartesian product) where there is a relationship between the elements of A and B.

Binary Relations



Set of robots $M = \{ r_1 , r_2 , r_3 \}$

Cartesian product = $M \times M$

Relation: Who can see who?

Each robot has a sensing circle and can see all robots that lie in its sensing circle.

(r_1 , r_2) = r_1 can see r_2

(r_3 , r_1) = r_3 can see r_1

(r_3 , r_2) = r_3 can see r_2

$\{ (r_1 , r_2) , (r_3 , r_1) , (r_3 , r_2) \} \subseteq M \times M$

Binary Relations

- If **(x, y) ∈ R** (where R is the relationship), we can also write **x R y** indicating that there exists a binary relationship between x and y.
- Relations can be defined on both **finite and infinite sets.**

Example:

Some binary relations defined over the set A where A= {0, 1} that are finite include:

- Cartesian product: $A \times A$
- Equality relationship = {(0, 0), (1, 1)}
- Less than relationship = {(0, 1)}

Binary Relations

Example:

Some binary relations defined between the set of reals and the set of integers that are infinite include:

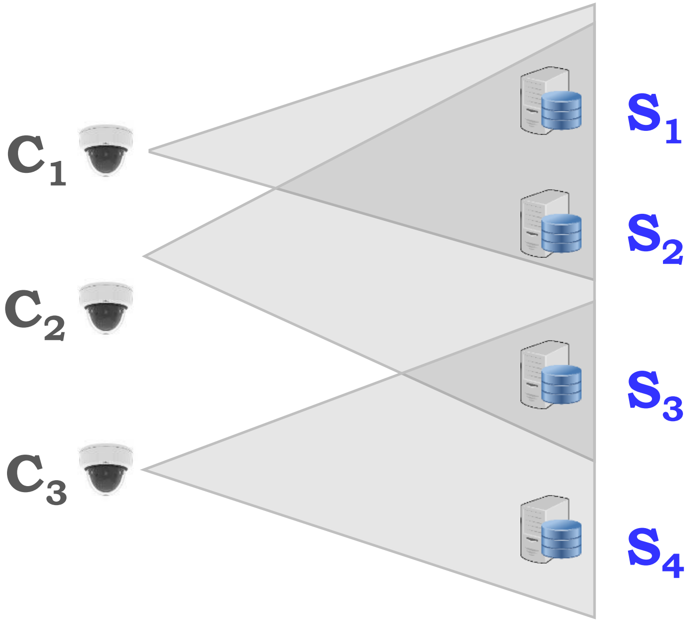
- xRy if $|x - y| \leq 1$.

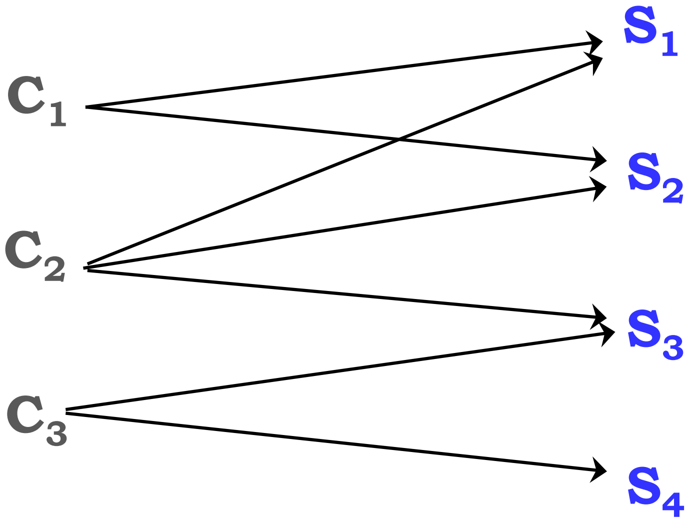
In other words, xRy (x is related to y) if the distance between real number x and integer y is at most 1.

Binary Relations – Arrow Diagram Representation

There are couple of common ways to represent binary relationships.

Arrow diagram The elements of A are listed on the left (domain) and the elements of B are listed on the right (co-domain). There is an arrow from a ∈ A to b ∈ B if aRb.

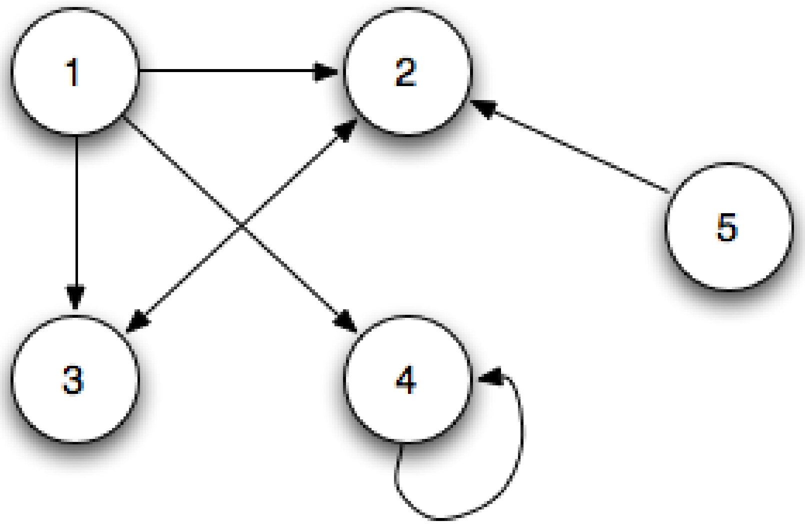




Binary Relations – Matrix Representation

A **matrix representation** of relation R between A and B is a rectangular array of numbers with |A| rows and |B| columns.

- Each row corresponds to an element of A
- Each column corresponds to an element of B.
- For a ∈ A and b ∈ B, there is a 1 in row a, column b, if aRb. Otherwise, there is a 0.



	1	2	3	4	5
1	0	1	1	1	0
2	0	0	1	0	0
3	0	1	0	0	0
4	0	0	0	1	0
5	0	1	0	0	0

Binary Relations - Practice

Exercise:

A = {r, o, t, p, c}

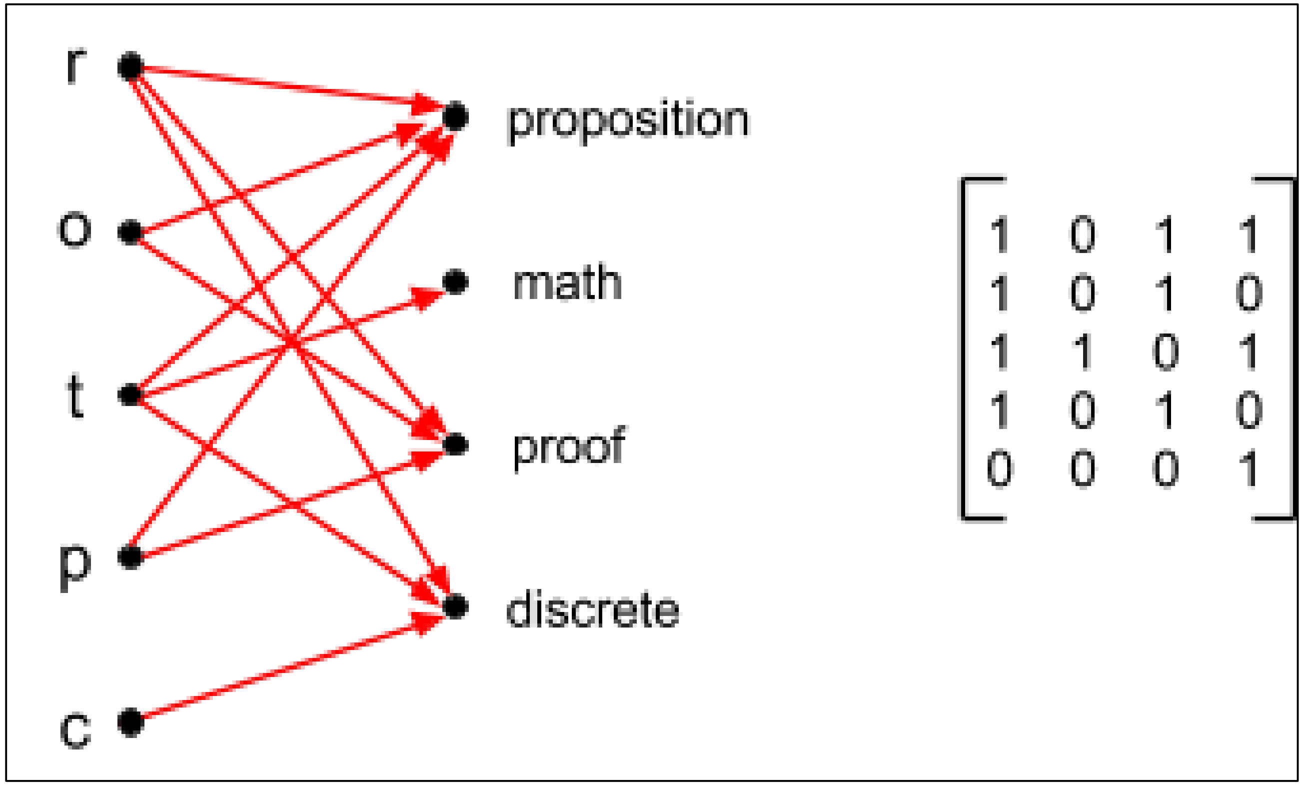
B = {discrete, math, proof, proposition}

Relation: $R \subseteq A \times B$ such that (letter, word) is in the relation if that letter occurs somewhere in the word draw the following:

1. The **arrow diagram** representation of the relation.
2. The **matrix representation** of the relation.

Binary Relations - Practice

A = {r, o, t, p, c} and B = {discrete, math, proof, proposition},
R ⊆ A × B such that (letter, word) is in the relation if that letter occurs somewhere in the word

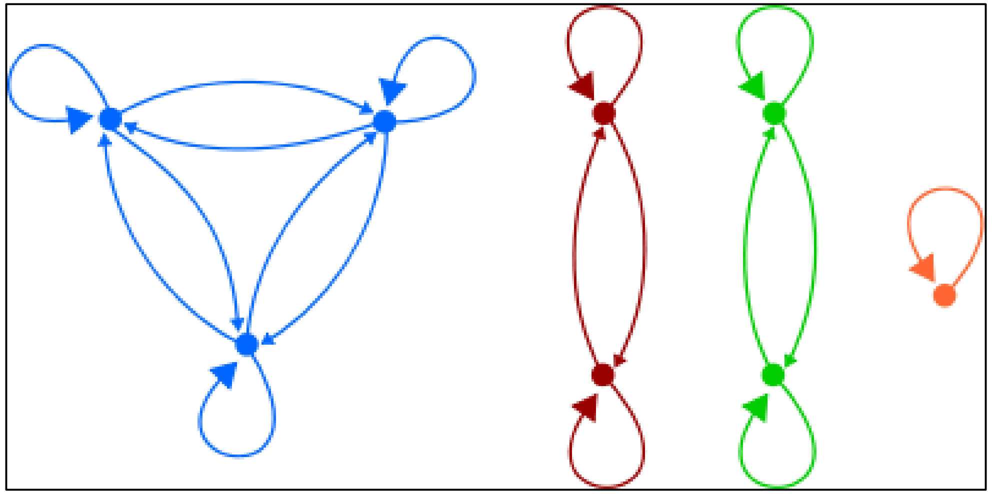


Properties of Binary Relations

The relation R is **reflexive** if and only if

for **every** $x \in A$, xRx

In other words, every x is related to itself.

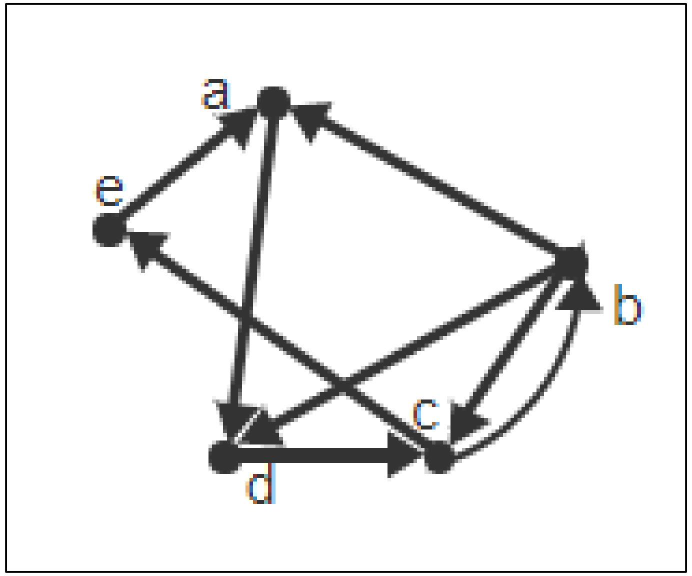


Reflexive

Properties of Binary Relations

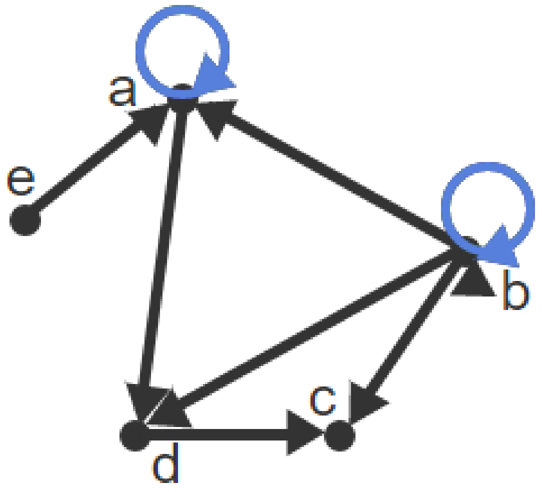
The relation R is **anti-reflexive** if and only if

for **every** $x \in A$, it is not true that xRx



Anti-reflexive

Properties of Binary Relations



A = { a, b, c, d, e }

Reflexive ? ✗

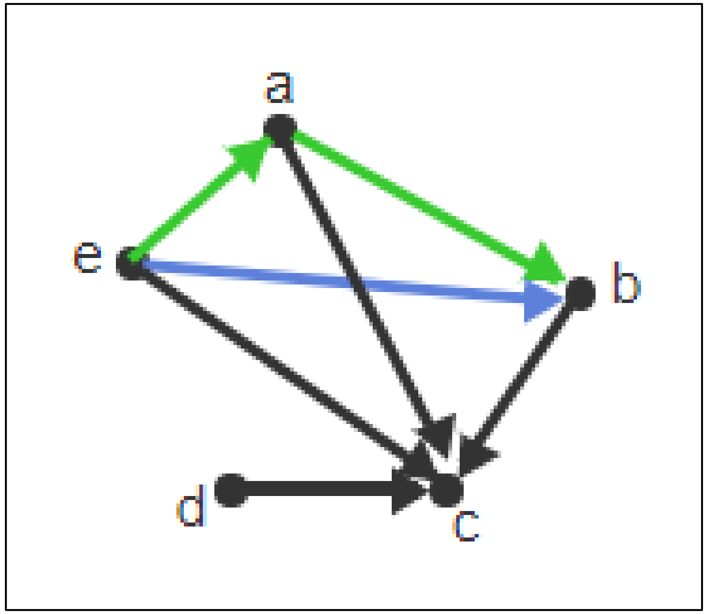
Anti-reflexive ? ✗

It is possible that a relation R is neither reflexive, nor anti-reflexive.

Properties of Binary Relations

The relation R is **transitive** if and only if for **every** x, y, $z \in$ A,

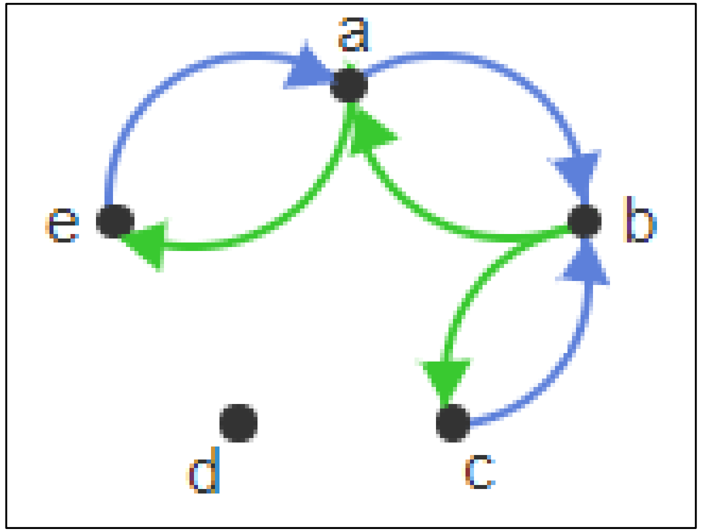
xRy and yRz implies that xRz.



Properties of Binary Relations

The relation R is **symmetric** if and only if for **every** $x, y \in A$,

$$xRy \text{ implies that } yRx.$$



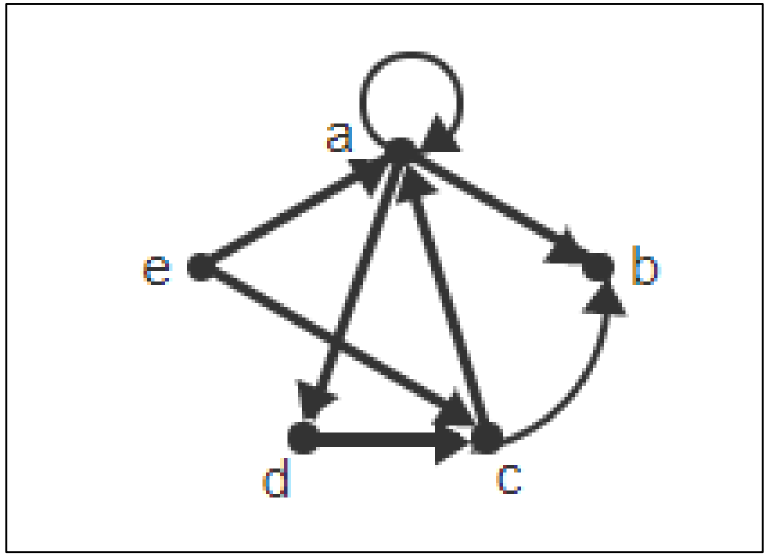
Properties of Binary Relations

The relation R is **anti-symmetric** if and only if for **every** x, $y \in$ A,

xRy and yRx imply that $x = y$.

In other words,

if $x \neq y$, then both xRy and yRx cannot be true at the same time.



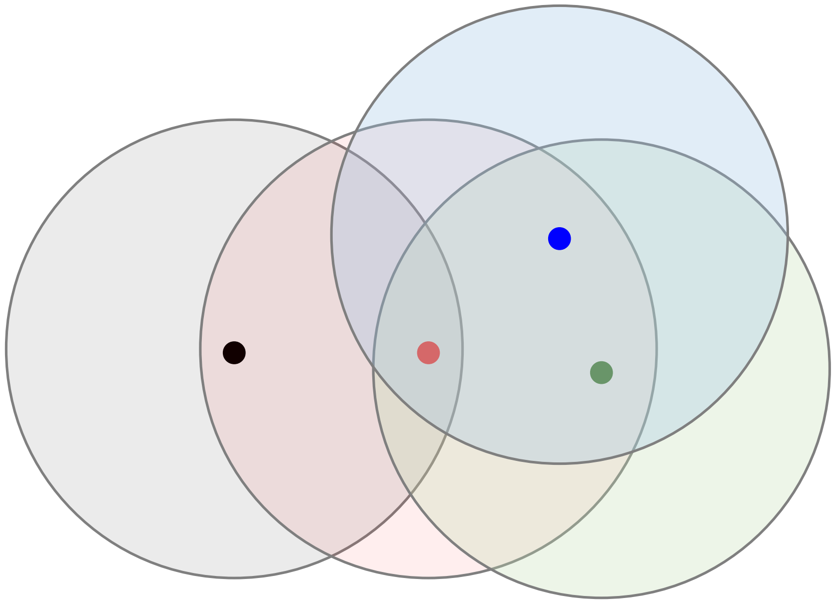
Anti-symmetric

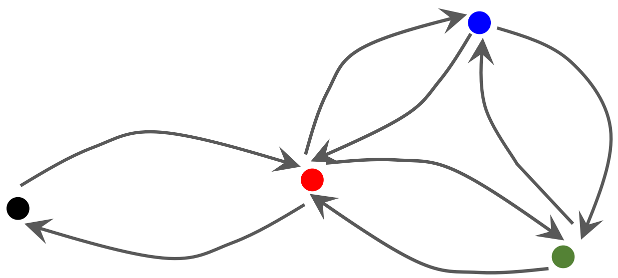
Properties of Binary Relations

If all robots in our previous example have exactly the **same sensing circle**, then we get a **symmetric** relation.

Why? Lets see …

If robot *i* can see robot *j*, then robot *j* can also see robot *i* as their sensing circles are same and the distance between *i* and *j* is smaller than the radius of sensing circle for both *i* and *j*.





Properties of Binary Relations – Practice

Consider the relation,

R: less than or equal over the set of **real numbers**.

Explain your reasoning.

1. Is R reflexive?
2. Is R symmetric?
3. Is R transitive?
4. Is R irreflexive?
5. Is R anti-symmetric?

Properties of Binary Relations – Practice

Is it **reflexive**?

(xRx for all $x \in A$)

Yes.

If $x \in A$ then $x \leq x$ is also true since they are equal. So the relationship is reflexive (consider $5 \leq 5$).

Properties of Binary Relations – Practice

Is it **symmetric**?
(xRy implies yRx for all x, $y \in$ A)

No.

Counterexample:

- Note that (5, 6) $\in$ R because $5 \leq 6$.
- However (6, 5) $\notin$ R since $6 > 5$.

Properties of Binary Relations – Practice

Is it **transitive**?

($\forall$ x, y, $z \in$ A; xRy and yRz implies xRz)

Yes.

- If $x \leq y$ and $y \leq z$, then $x \leq z$ for all reals.
- So the relationship is transitive.
- For example, $5 \leq 6$ and $6 \leq 8$ implies $5 \leq 8$.

Properties of Binary Relations – Practice

Is it **anti-reflexive**?
(xRx is not true for all $x \in$ A)

No.

Counterexample:

- 5 $\in$ A, and 5 $\leq$ 5. So, (5, 5) $\in$ R.
- Thus, by definition R is not anti-reflexive.

Properties of Binary Relations – Practice

Is it **anti-symmetric**?

(xRy and yRx implies $x = y$ for all $x, y \in$ A)

Yes.

- The only way $x \leq y$ and at the same time $y \leq x$ is when $x = y$.

Symmetric Closure

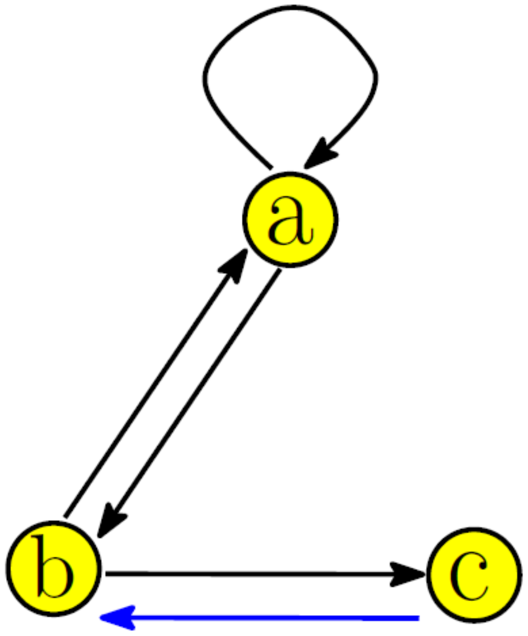
Consider a set: $A = \{ a, b, c \}$
Binary relation: $R = \{ (a, a) , (a, b) , (b, a) , (b, c) \} \subseteq A \times A$

Is R **symmetric**? **No**

What is the minimum set of pairs that we need to add to R to make it reflexive?

$\mathbf{s(R)} = R \cup \{ (c, b) \}$

Symmetric closure

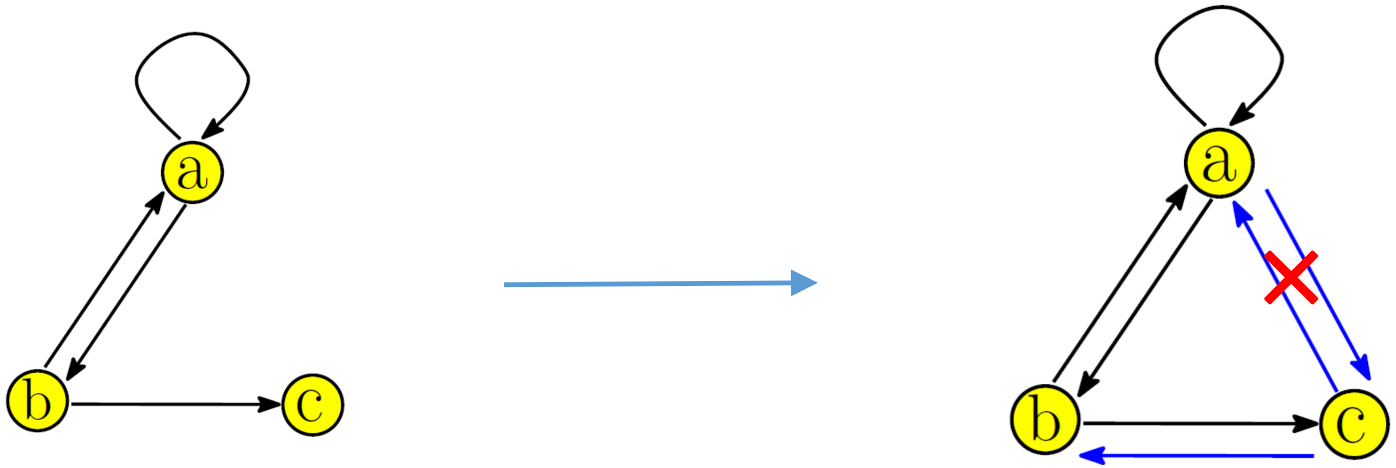


Symmetric Closure

To obtain a symmetric closure, if there is a directed arc in one direction, add a directed arc in **opposite direction** (except self loops), if it is **missing.**

Question?

R



s(R) ?

No

Question?

If R is a symmetric, then what will be its symmetric closure?

Question?

In terms of matrices, it means matrix and its transpose are same.

Transitive Closure

Consider a set: $A = \{ a, b, c \}$

Binary relation: $R = \{ (a, a) , (a, b) , (b, a) , (b, c) \} \subseteq A \times A$

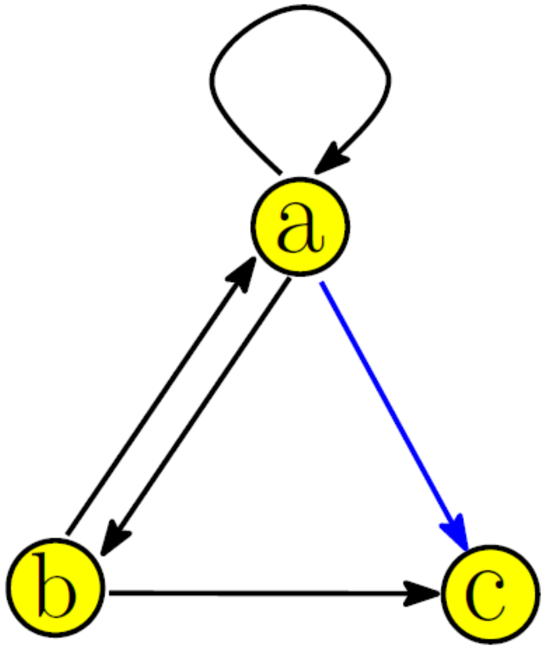
Is R **transitive**? **No**

What is the minimum set of pairs that we need to add to R to make it transitive?

$\mathbf{t(R)} = R \cup \{ (a, c) \}$

Is there anything missing?

bRa and aRb $\rightarrow$ bRb **??**



Transitive Closure

Consider a set: $A = \{ a, b, c \}$

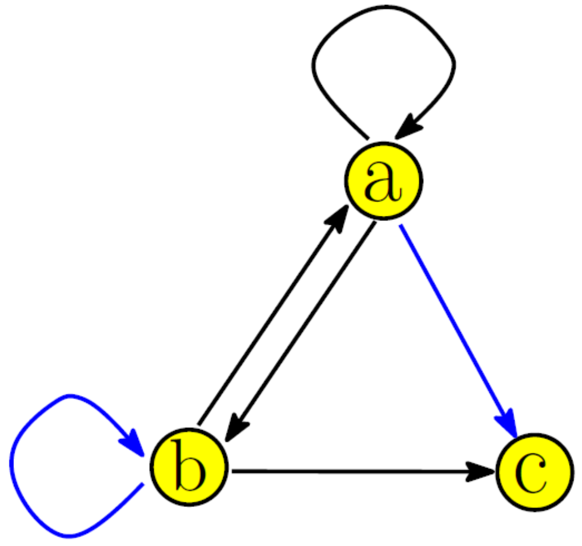
Binary relation: $R = \{ (a, a) , (a, b) , (b, a) , (b, c) \} \subseteq A \times A$

So, the transitive closure should include (b,b).

$\mathbf{t(R)} = R \cup \{ (a, c) , (b, b) \}$

↑

Transitive closure



A Practice Question

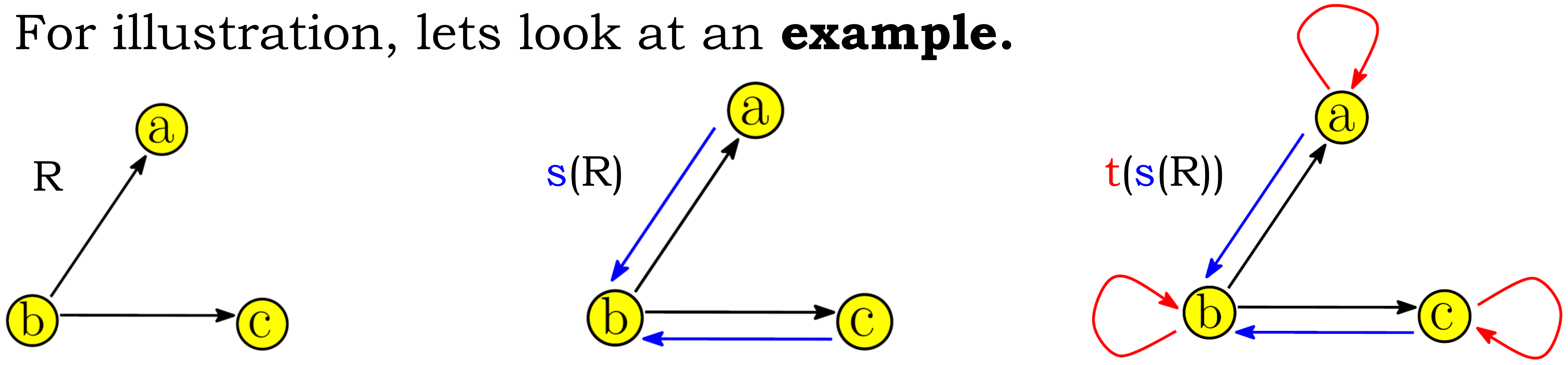
Let **R** $\subseteq$ **A** × **A** be a relation such that for every element a ∈ A, there is some b ∈ B such that either aRb or bRa.

Is it true or false?

The transitive closure of the symmetric closure of R is also reflexive, that is, **t(s(R)) is reflexive**.

Yes. Can you show how?

For illustration, lets look at an **example.**



A Practice Question

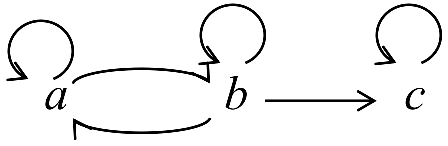
Consider a set: A = { a, b, c }

Binary relation: R = { (a, b) , (b, a) , (b, c) } $\subseteq$ A × A

Compute the three closures of R and draw graph representations.

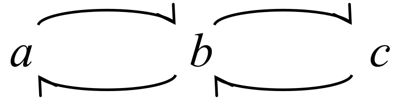
r(R) = {(a, b), (b, a), (b, c), (a, a), (b, b), (c, c)}

r(R):



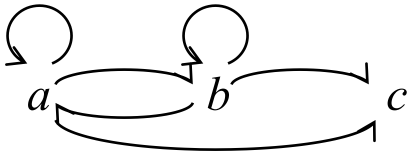
s(R) = {(a, b), (b, a), (b, c), (c, b)}

s(R):



t(R) = {(a, b), (b, a), (b, c), (a, a), (b, b), (a, c)}

t(R):



Relations and Orders

Binary relations can be used to formalize the notion of (partial) ordering.

What does it mean when items are **"ordered"**? Intuitively, we think that one item has to go before another.

Example:

1. 

2. 

3. 

4. 

(Coke < Pepsi)

(Pepsi < Orange juice)

(Orange juice < Apple juice)

Relations and Orders

(Coke < Pepsi)

(Orange juice < Apple juice)

Can you (completely) list drinks in the order of their preference?

Sometimes, it is very difficult to establish a **totally ranked list** (a total order of elements), for instance, where notion of precedence between some but not all pairs is present.

The notion of **partial order** is extremely useful here.

First, lets see what does it mean to **compare elements pairwise**.

Partial Order

List movies in the order of liking.

	Rachel	Jason
Wizard of Oz	**1**	**3**
Godfather	**3**	**1**
Forest Gump	**4**	**2**
Jurassic Park	**2**	**4**

Set of movies: { **G**, **F**, **W**, **J** }

Rachel's ordering: **G < F < W < J**

Jason's ordering: **W < J < G < F**

"***x*** < ***y***" symbol means here that x is preferred over b, or x must come before b.

Partial Order

List movies in the order of your liking.

	Rachel	Jason
Wizard of Oz	1	3
God		
Fore		
Jura		

How can we order movies such that preferences of both Rachel and Jason can be realized at the same time?

For Rachel: G < F < W < J

For Jason: W < J < G < F

Partial Order

Instead of a totally ranked list, **compare pairwise elements** (movies).

Then, there will be some pairwise comparisons that will represent preferences of **both** Rachel and Jason.

For Rachel: **G < F < W < J**

(G < F) , (G < W) , (G < J) ,
(F < W) , (F < J) , (W < J)

For Jason: **W < J < G < F**

(W < J) , (W < G) , (W < F) ,
(J < G) , (J < F) , (G < F)

So, for both persons, we know (G < F) and (W < J).

Partial Order

So, instead of **“completely”** ordering the elements of a set, we have **“partially”** ordered them.

$$\{ (G < F) , (W < J) \}$$

Note that its also a **relation**
(as we have been studying)

Partial Order

So, instead of **“completely”** ordering the elements of a set, we have **“partially”** ordered them.

Binary relations can be used to represent partial order.

{ (G < F) , (W < J) }

Note that its also a **relation**
(as we have been studying)