Discrete Algebraic Structures

WiSe 2025/2026

Prof. Dr. Antoine Wiehe
Research Group for Theoretical Computer Science

- If you are registered for a tutorial but are not attending, please unregister!
- 16 points not necessary for students registered in WiSe 2023
- Use of LLMs for the weekly quiz is strictly forbidden

  Be careful of the blackboxing/*vibe coding* phenomenon!

$$\begin{aligned} \text{Studienleistungen} &\subseteq \text{official examinations} \\ \text{Use of LLM} &\subseteq \text{cheating} \\ \Longrightarrow \quad \text{Use of LLM during quiz} &\subseteq \text{cheating during official examination} \end{aligned}$$

$$\begin{aligned} &\text{SL} \subseteq \text{Exam} \\ &\text{LLM} \subseteq \text{Cheat} \\ &\Rightarrow \text{SL} \cap \text{LLM} \subseteq \text{Exam} \cap \text{Cheat} \end{aligned}$$

Allgemeine Studien- und Prüfungsordnung:

§ 25b Deception and breach of order

(1) [1]If the student attempts to influence the result of the examination or coursework by deception, in particular by using unauthorized aids, the examination or coursework in question will be graded as "fail" (5.0).

Sets

- Definition of $\cap, \cup, \times, \setminus, \Delta, \mathcal{P}$
- How to prove $S \subseteq T$
- How to prove $S = T$
  (prove $S \subseteq T$ and $T \subseteq S$)

Functions

- Definition; be able to determine if a given set $f \subseteq A \times B$ is a function or not
- Properties of functions: injectivity, surjectivity, bijectivity
- Composition of functions
- Identity function
- Inverses

Logic

- Recognize legal formulas of propositional/predicate logic
- How to compute truth table of a propositional formula
- From a truth table, compute a corresponding formula
- Be able to determine logical equivalences between formulas
- Know the valid ways to prove an implication $\varphi \Rightarrow \psi$
- Know how to compute $\neg\varphi$

Part 5: Algebra

- Cryptography
- Error correction
- Quantum Computing

Part 3: Combinatorics and counting

- Runtime analysis of algorithms
- Basics of probabilities

Part 4: basic number theory

Part 2: Graphs and relations

- Representing data and relationships between things
- Important basis for the rest of your studies
  (in particular "Algorithms and Data Structures")

Part 1: introduction to mathematics

- Sets, functions
- Basics in logic and proofs

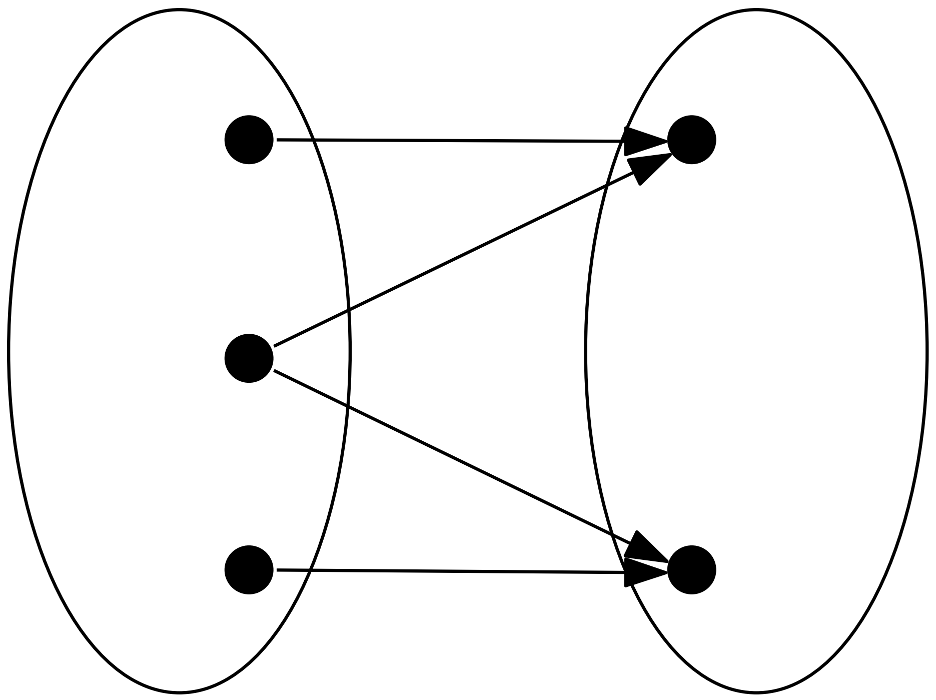
- We know functions $A \to B$

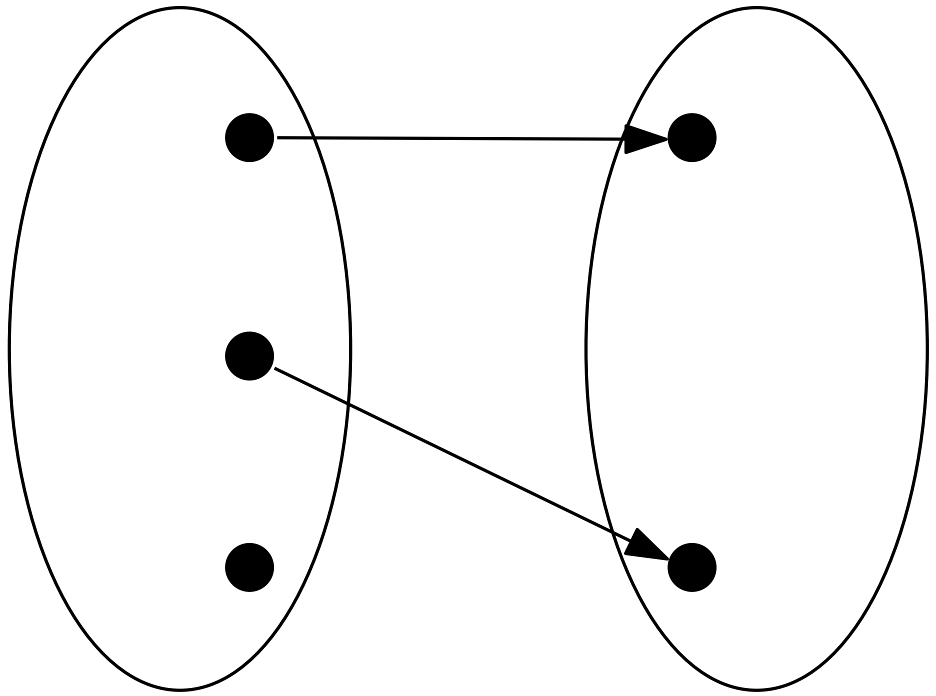
  1 input (from A) $\longrightarrow$ 1 output (from B)

- In many applications, we want to express a relationship between things that is *not* a function

1 input		multiple outputs
movie	$\longrightarrow$	directors
book	$\longrightarrow$	authors
x	$\longrightarrow$	a square root of x

1 input		0 or 1 output
element of an array	$\longrightarrow$	next element
x	$\longrightarrow$	$1/x$



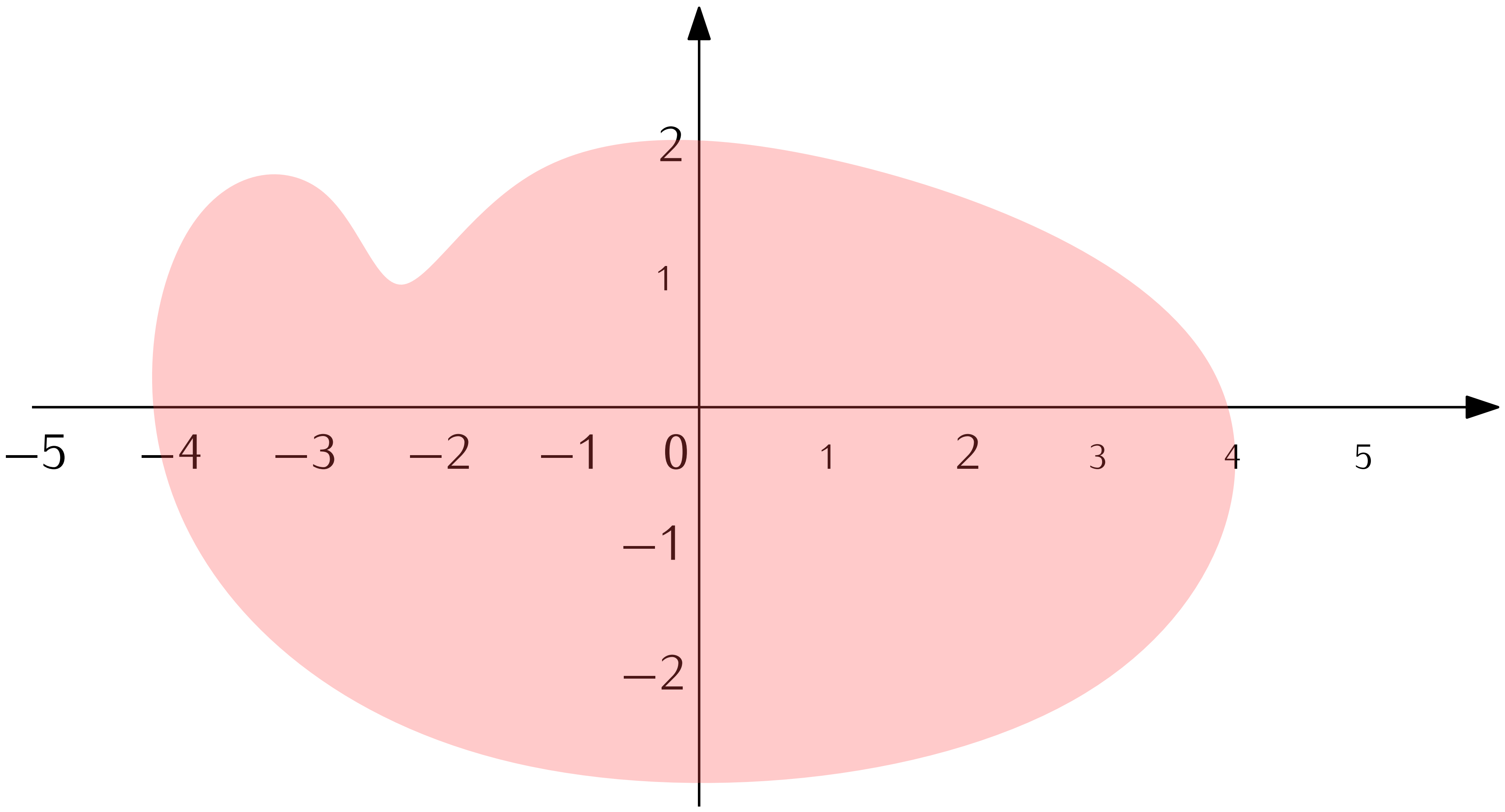


Definition. A function $f\colon A \to B$ is a set $f \subseteq A \times B$ such that for every $a \in A$, there exists a unique $b \in B$ such that $(a, b) \in f$.

Definition. A function $f\colon A \to B$ is a set $f \subseteq A \times B$ ~~such that for every $a \in A$, there exists a unique $b \in B$ such that $(a, b) \in f$~~.

Definition. A ~~function~~ binary relation R is a set $R \subseteq A \times B$ ~~such that for every $a \in A$, there exists a unique $b \in B$ such that $(a, b) \in f$~~.

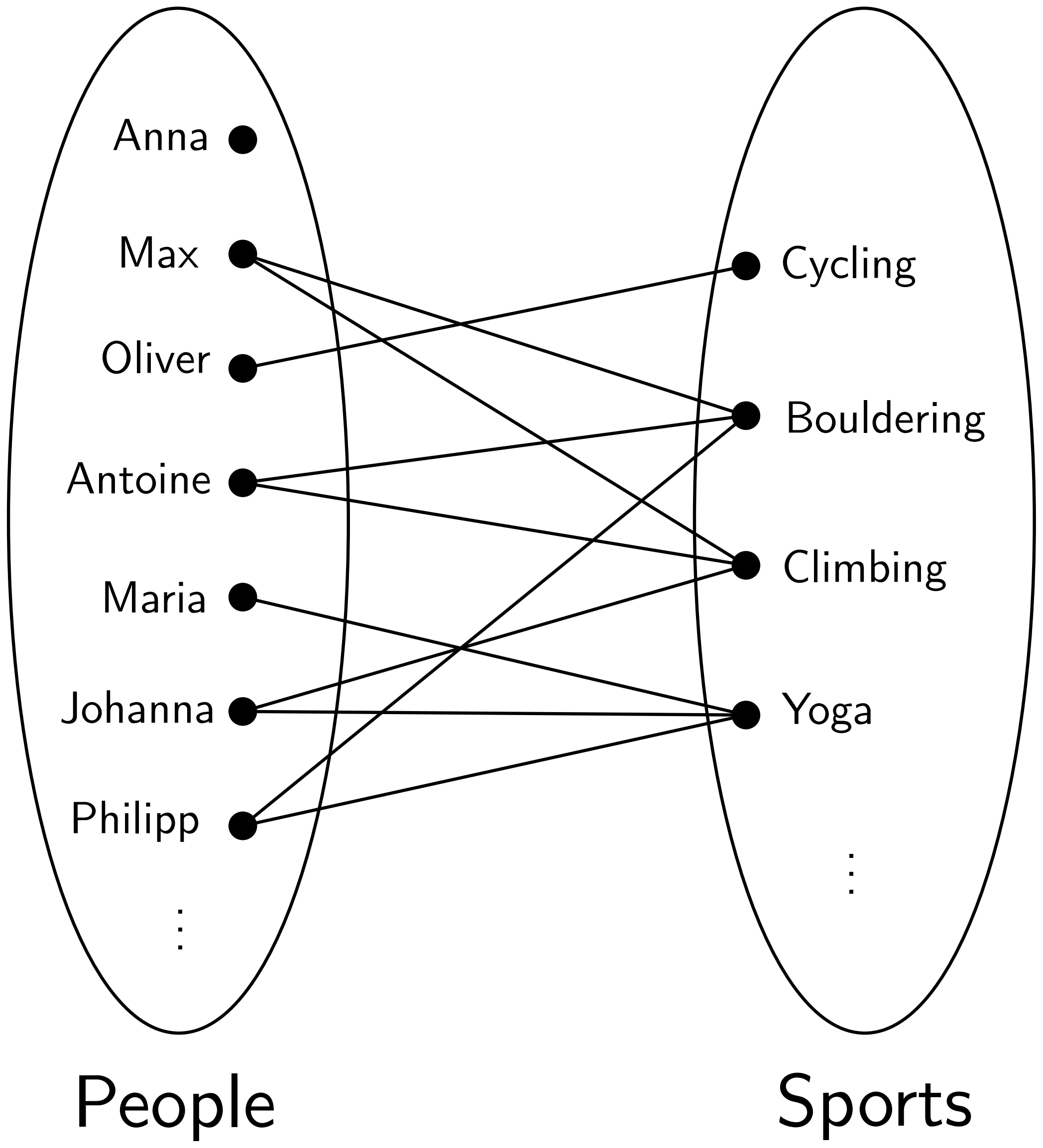
Definition. A binary relation R is a set $R \subseteq A \times B$.



$R \subseteq \mathbb{R} \times \mathbb{R}$

a relation that is not a function

Definition. A binary relation R is a set $R \subseteq A \times B$.



$\{(\text{Max}, \text{Climbing}), (\text{Maria}, \text{Yoga}), (\text{Philipp}, \text{Bouldering}), \ldots\}$

Definition. A binary relation R is a set $R \subseteq A \times B$.

- Most common way of representing structured information
- Every database probably contains a binary relation somewhere

Subscriptions $\subseteq$ Accounts $\times$ Channels Birthdate $\subseteq$ Accounts $\times\ \mathbb{N}$ Teaches $\subseteq$ Professors $\times$ Courses

One can also put more data together:

Definition. A relation on $A_1, \ldots, A_n$ is a set $R \subseteq A_1 \times A_2 \times \cdots \times A_n$.

- In data science: the sets $A_1, \ldots, A_n$ are the features, $R \subseteq A_1 \times \cdots \times A_n$ is the dataset

$$\text{Iris data set} \subseteq \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \{\text{I. setosa}, \text{I. virginica}, \text{I. versicolor}\}$$

$$(5.1, 3.5, 1.4, 0.2, \text{I. setosa})$$

sepal length sepal width petal length petal width species

Definition. A binary relation R is a set $R \subseteq A \times B$.

- Most common way of representing structured information
- Every database probably contains a binary relation somewhere

$$\text{Subscriptions} \subseteq \text{Accounts} \times \text{Channels} \qquad \text{Birthdate} \subseteq \text{Accounts} \times \mathbb{N} \qquad \text{Teaches} \subseteq \text{Professors} \times \text{Courses}$$

One can also put more data together:

Definition. A relation on $A_1, \ldots, A_n$ is a set $R \subseteq A_1 \times A_2 \times \cdots \times A_n$.

- In data science: the sets $A_1, \ldots, A_n$ are the features, $R \subseteq A_1 \times \cdots \times A_n$ is the dataset

$$\text{Iris data set} \subseteq \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \{\text{I. setosa}, \text{I. virginica}, \text{I. versicolor}\}$$

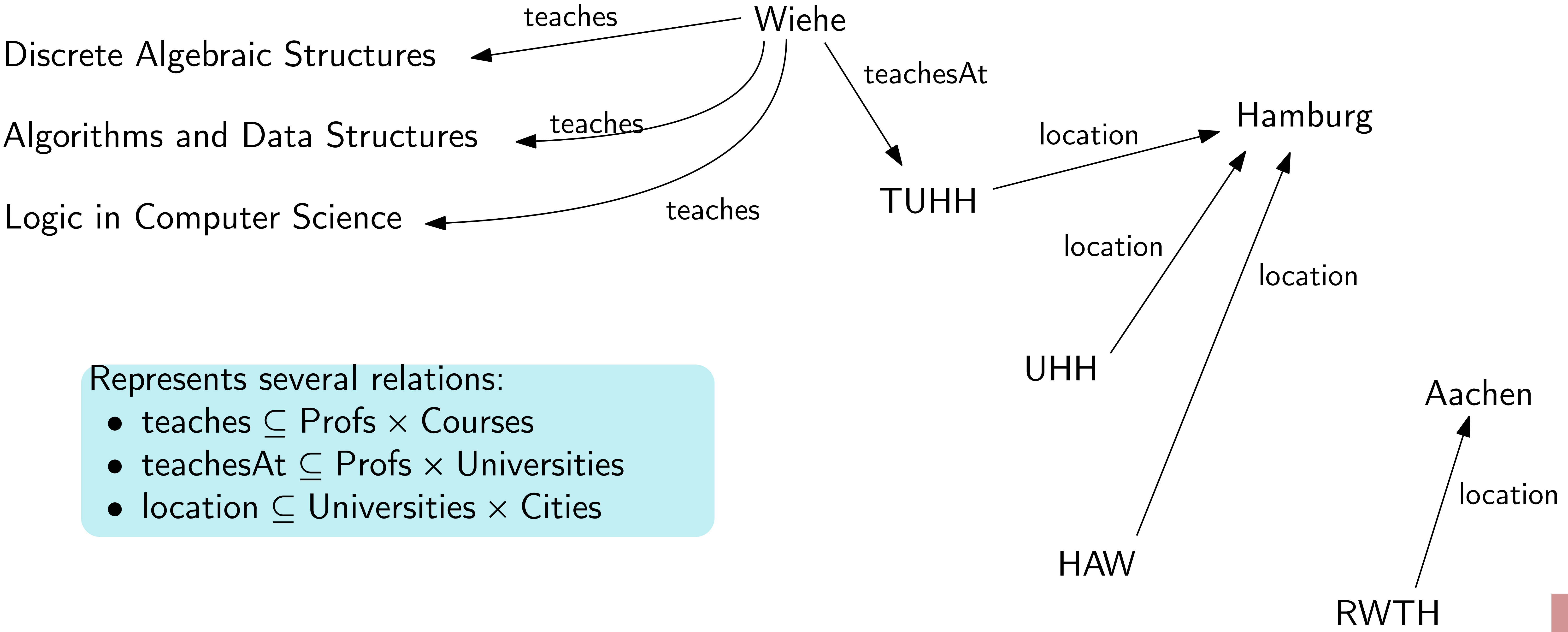
$$\text{YellowCab} \subseteq \text{DateTime} \times \text{DateTime} \times \mathbb{N} \times \mathbb{Q}$$

- In computer science: many relations together = relational database

$$\begin{aligned} \text{Movies} &\subseteq \mathbb{N} \times \text{String} \times \mathbb{N} \times \ldots \\ \text{Directors} &\subseteq \mathbb{N} \times \text{String} \times \ldots \\ \text{Directed} &\subseteq \mathbb{N} \times \mathbb{N} \end{aligned}$$

Knowledge graphs:

- semantic Web: representing information on the Web in a structured way
- `SPARQL` query language

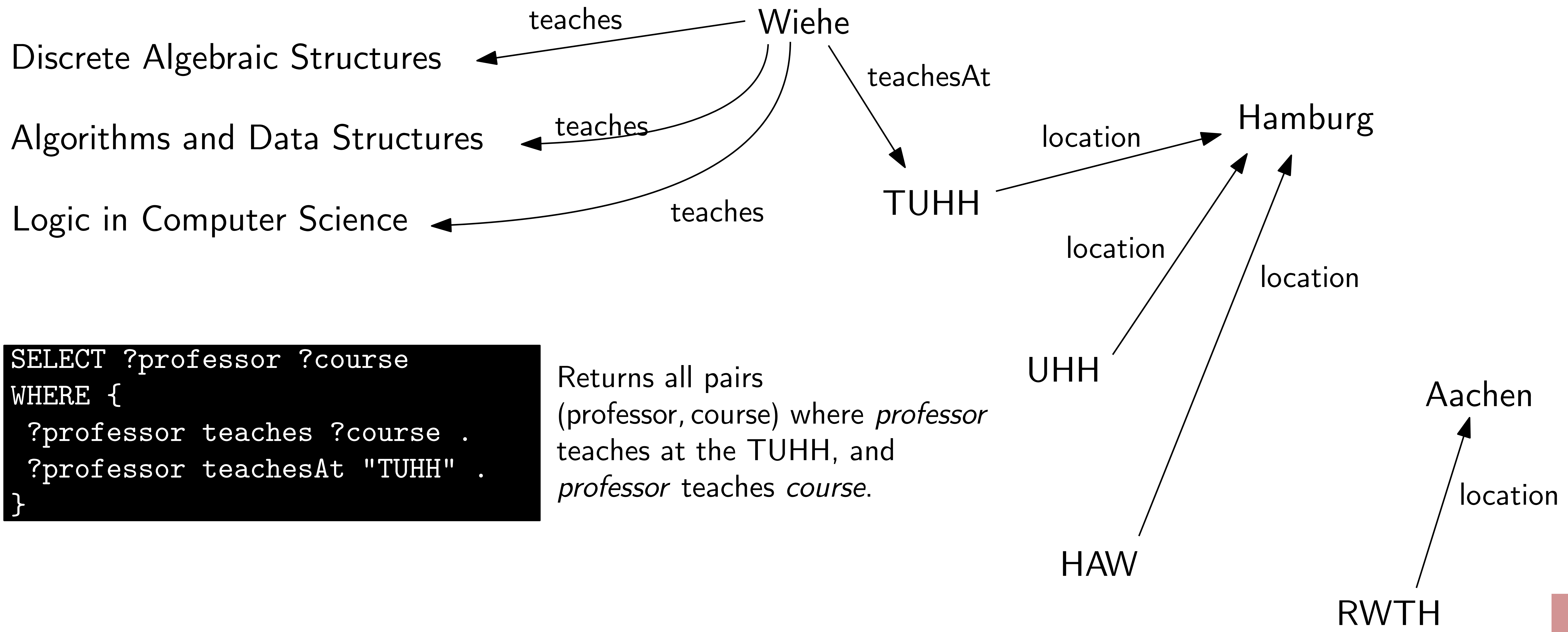


Represents several relations:

- teaches $\subseteq$ Profs $\times$ Courses
- teachesAt $\subseteq$ Profs $\times$ Universities
- location $\subseteq$ Universities $\times$ Cities

Knowledge graphs:

- semantic Web: representing information on the Web in a structured way
- SPARQL query language

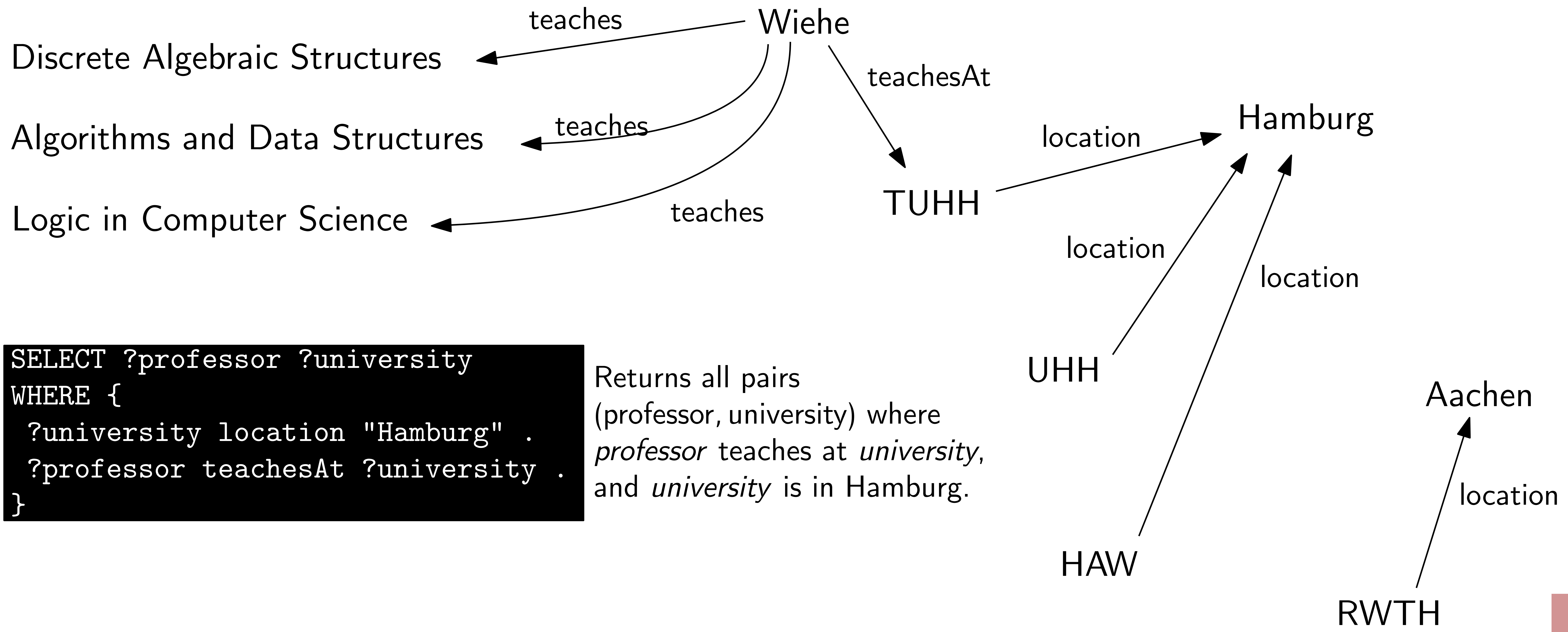



```
SELECT ?professor ?course
WHERE {
 ?professor teaches ?course .
 ?professor teachesAt "TUHH" .
}
```

Returns all pairs (professor, course) where *professor* teaches at the TUHH, and *professor* teaches *course*.

Knowledge graphs:

- semantic Web: representing information on the Web in a structured way
- SPARQL query language



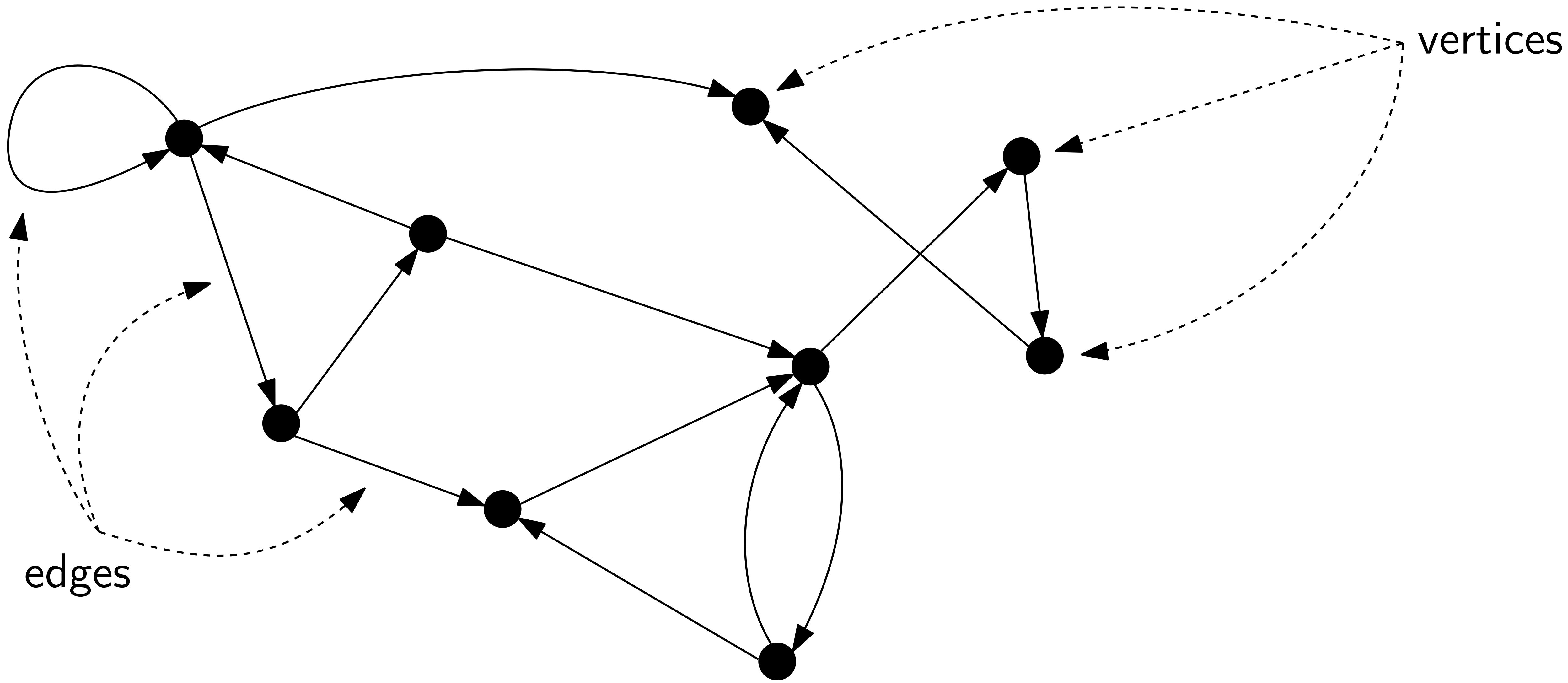

```
SELECT ?professor ?university
WHERE {
 ?university location "Hamburg" .
 ?professor teachesAt ?university .
}
```

Returns all pairs (professor, university) where *professor* teaches at *university*, and *university* is in Hamburg.

Very often:

- relations are binary $R \subseteq A \times B$
- $A = B$

Definition. A relation $R \subseteq A \times A$ is called a directed graph.
The elements of A are the vertices and the elements of R are the edges.



Algebraic properties of relations

Algebra: study of **operations** and **equations** on *stuff*

Number theory

Boolean algebra

Numbers: $+, \times, 1/x, 1, 0$

Matrices: $+, \times, M^{-1}, I_n, 0$

Sets: $\cap, \cup, \times, \Delta, \emptyset, \ldots$

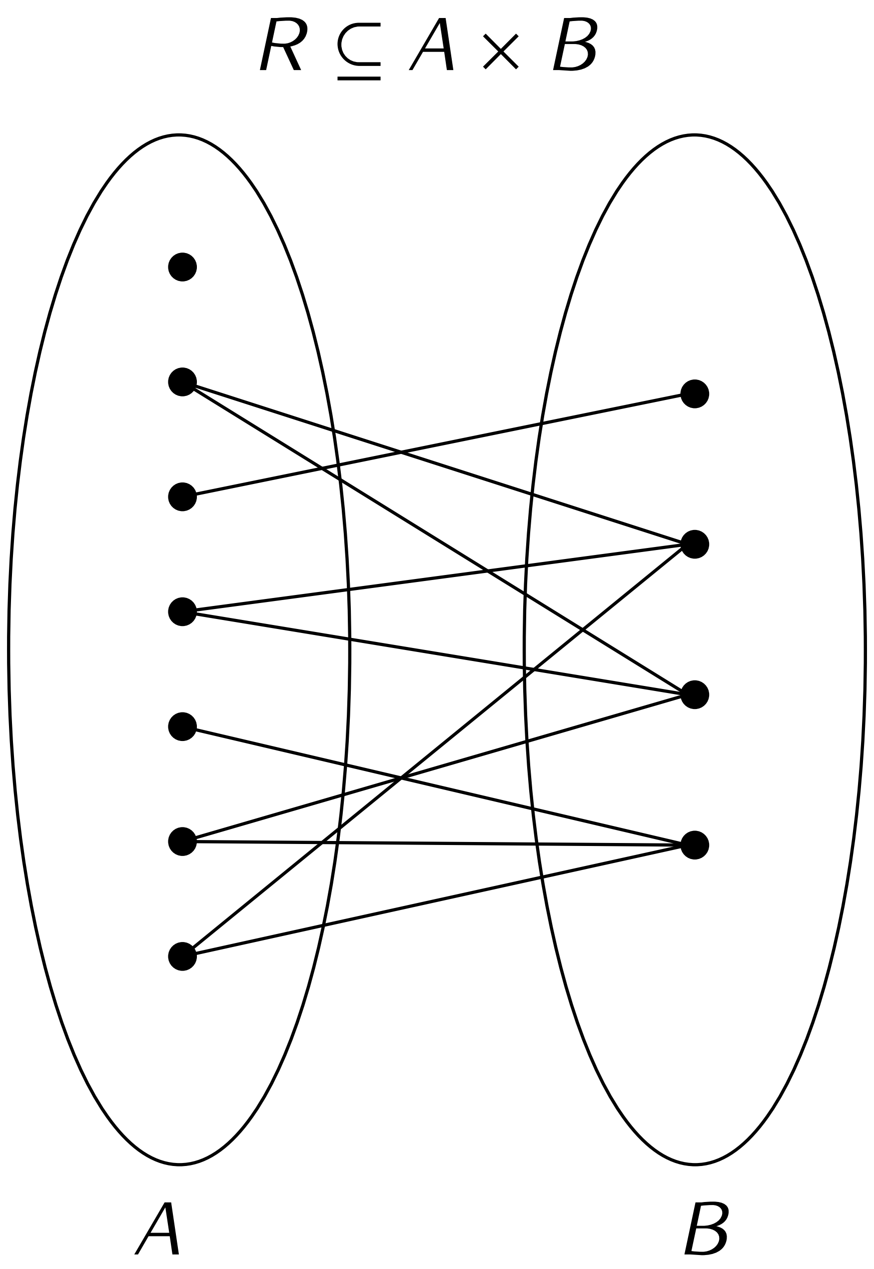
Functions: $\circ, f^{-1}, \mathrm{Id}_A$

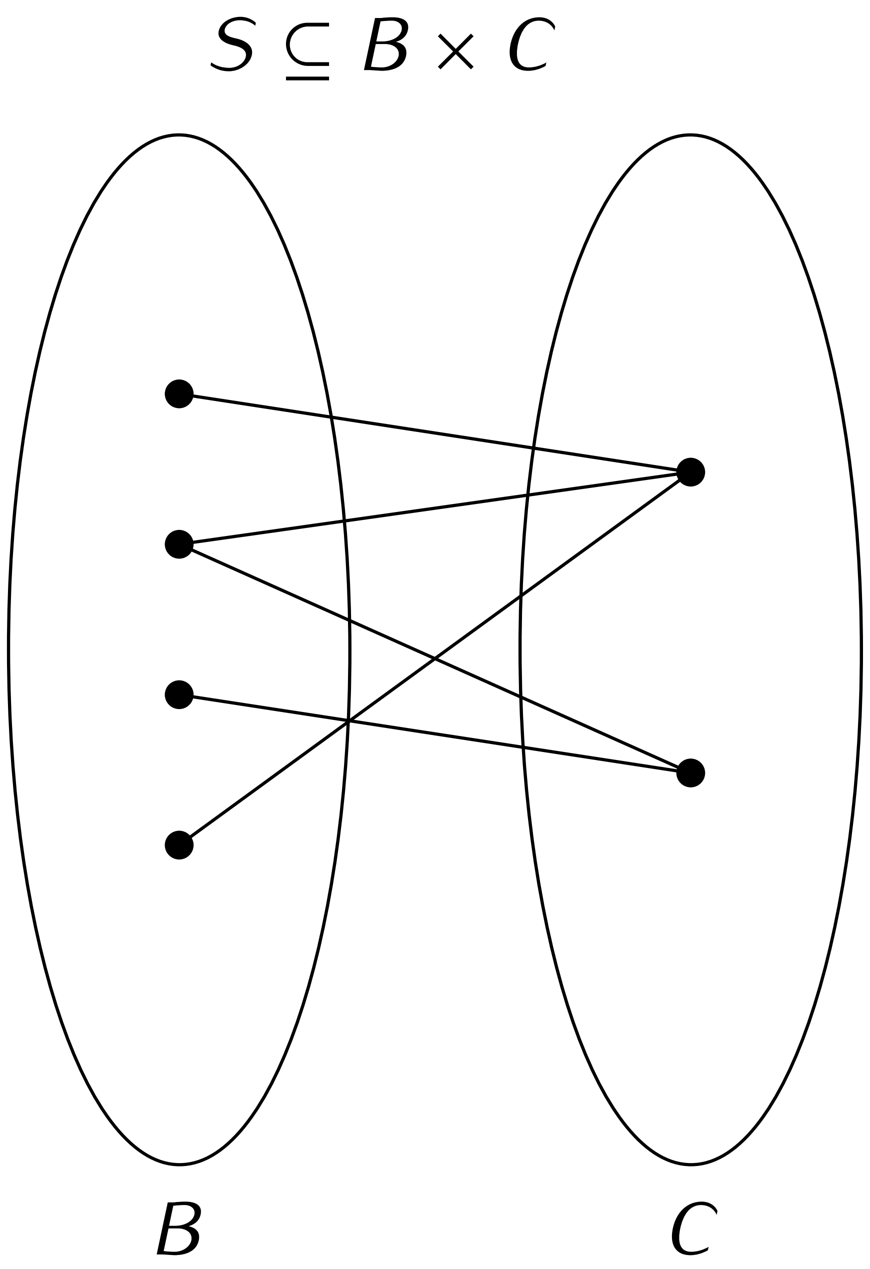
Booleans: $\wedge, \vee, \Rightarrow, \neg, \top, \bot$

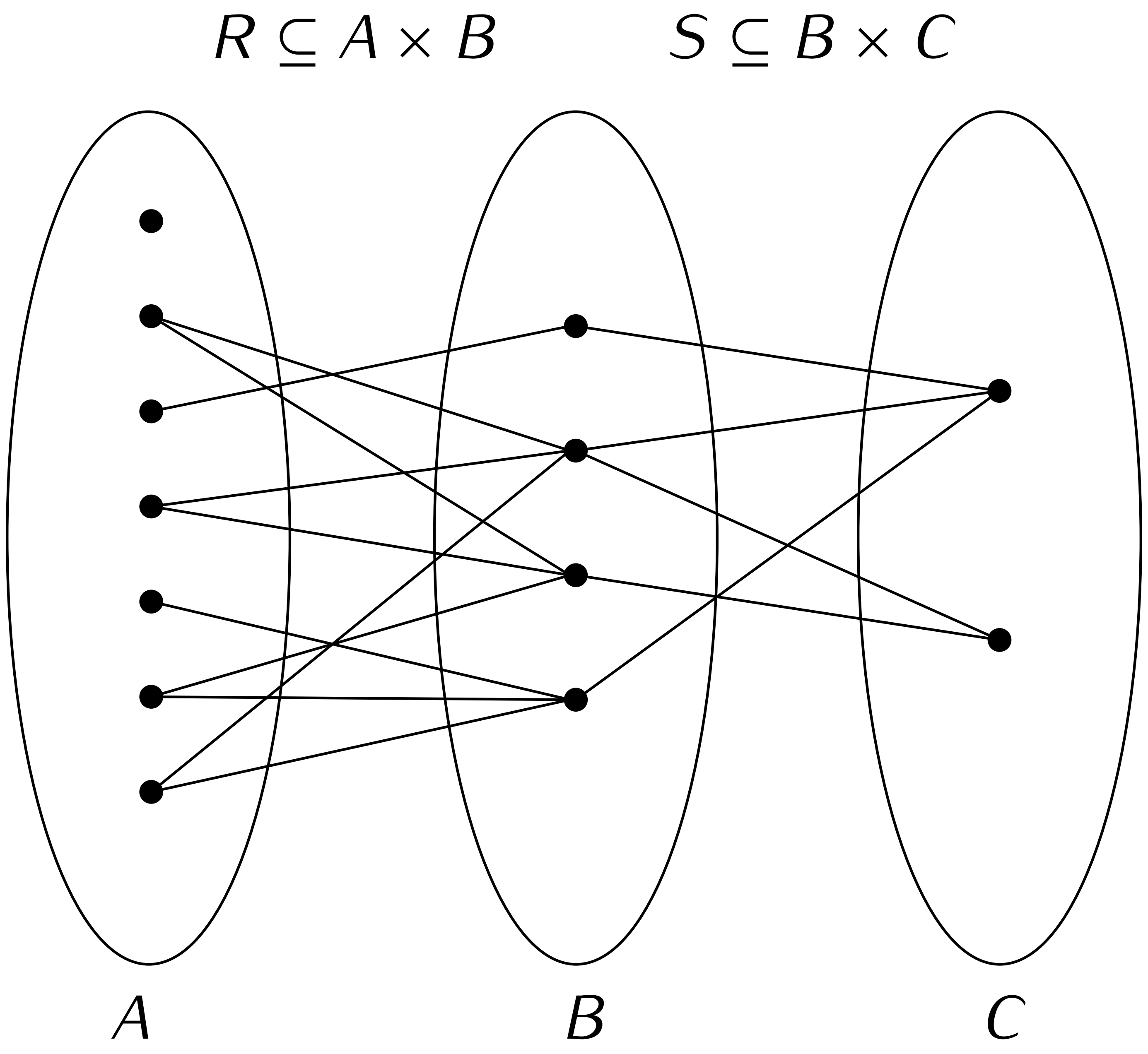
Relations: ?

Linear algebra

Relational algebra

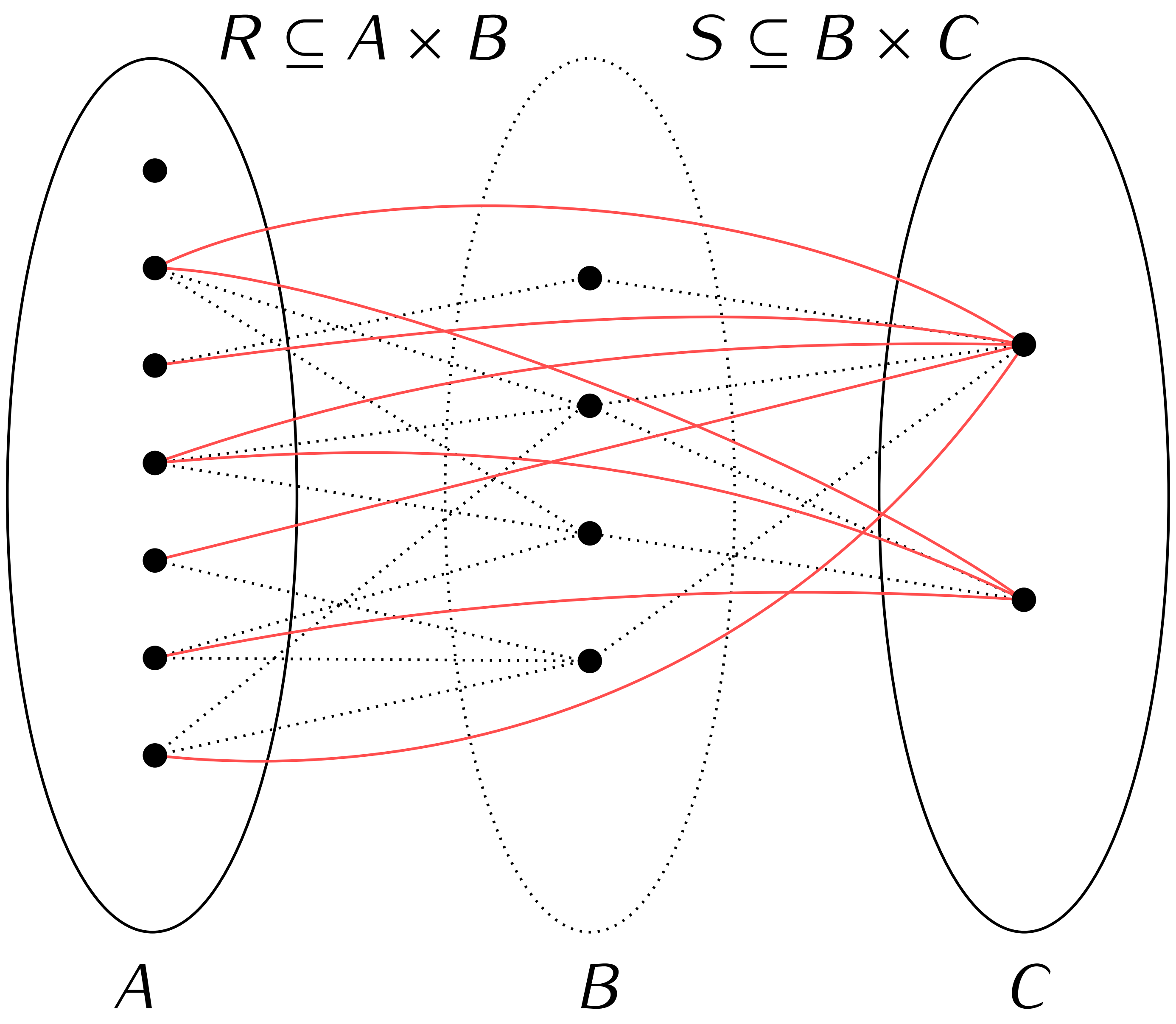

$R \subseteq A \times B$
A
B


$S \subseteq B \times C$
B
C



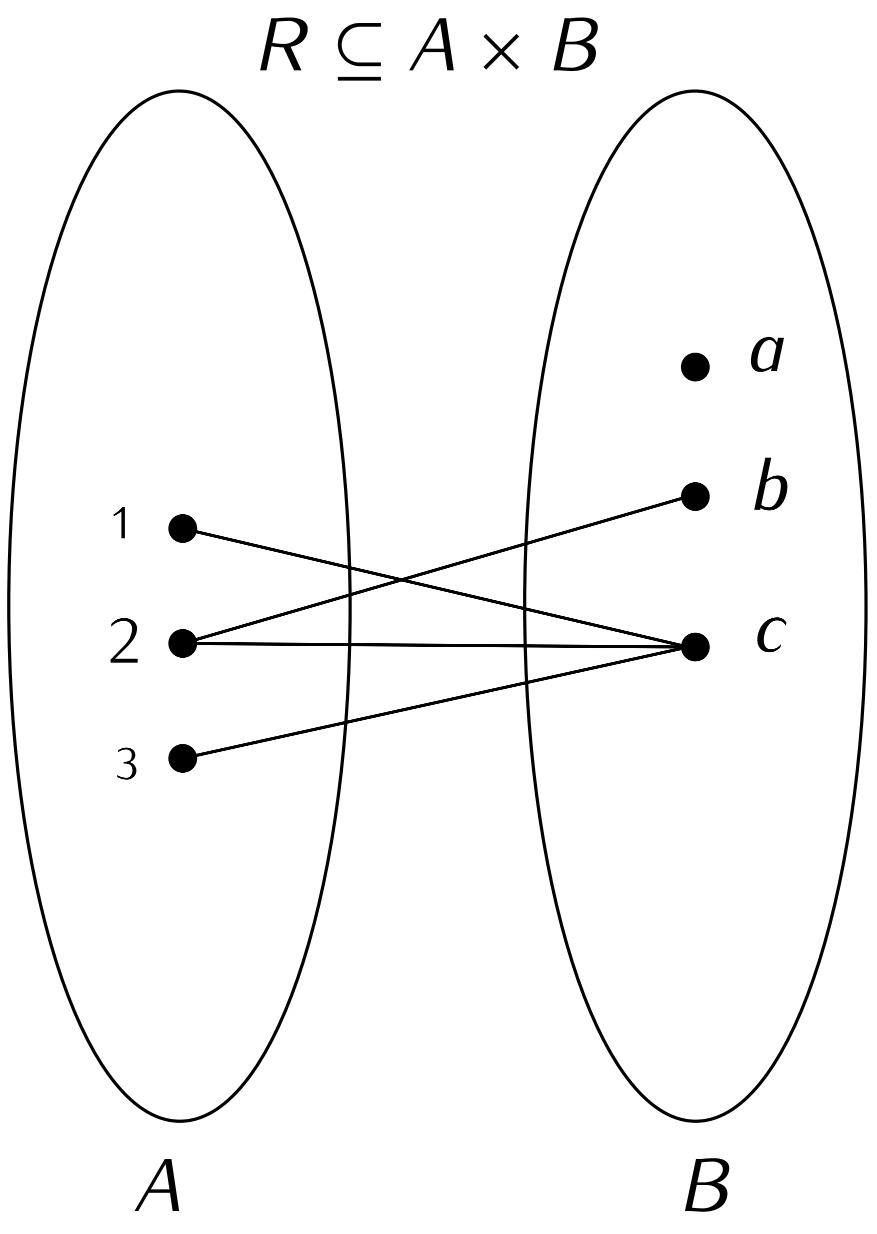
1. Glue the potatoes together

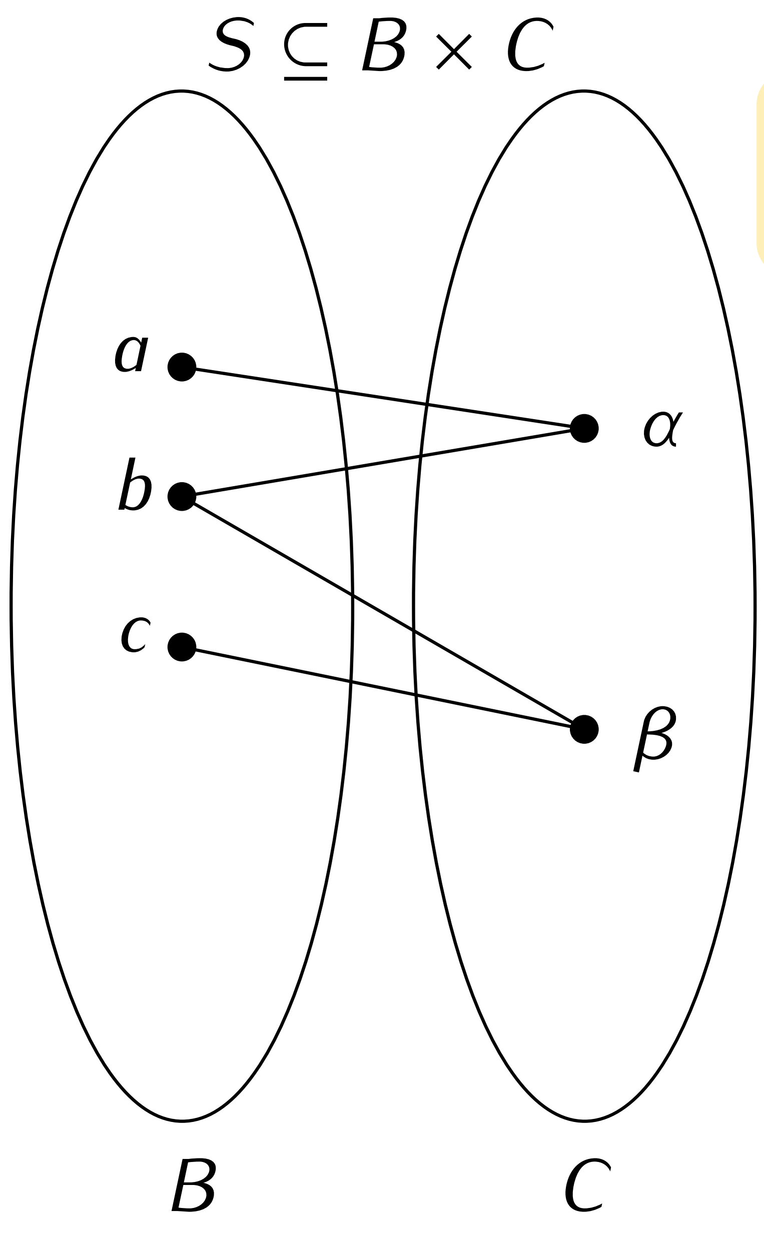
Definition. Let $R \subseteq A \times B$ and $S \subseteq B \times C$. The composition is defined by $S \circ R = \{(a, c) \in A \times C \mid \exists b \in B\, ((a, b) \in R \wedge (b, c) \in S)\}$.



1. Glue the potatoes together
2. Compute paths from A to C
3. Forget about B

Definition. Let $R \subseteq A \times B$ and $S \subseteq B \times C$. The composition is defined by $S \circ R = \{(a, c) \in A \times C \mid \exists b \in B \, ((a, b) \in R \wedge (b, c) \in S)\}$.



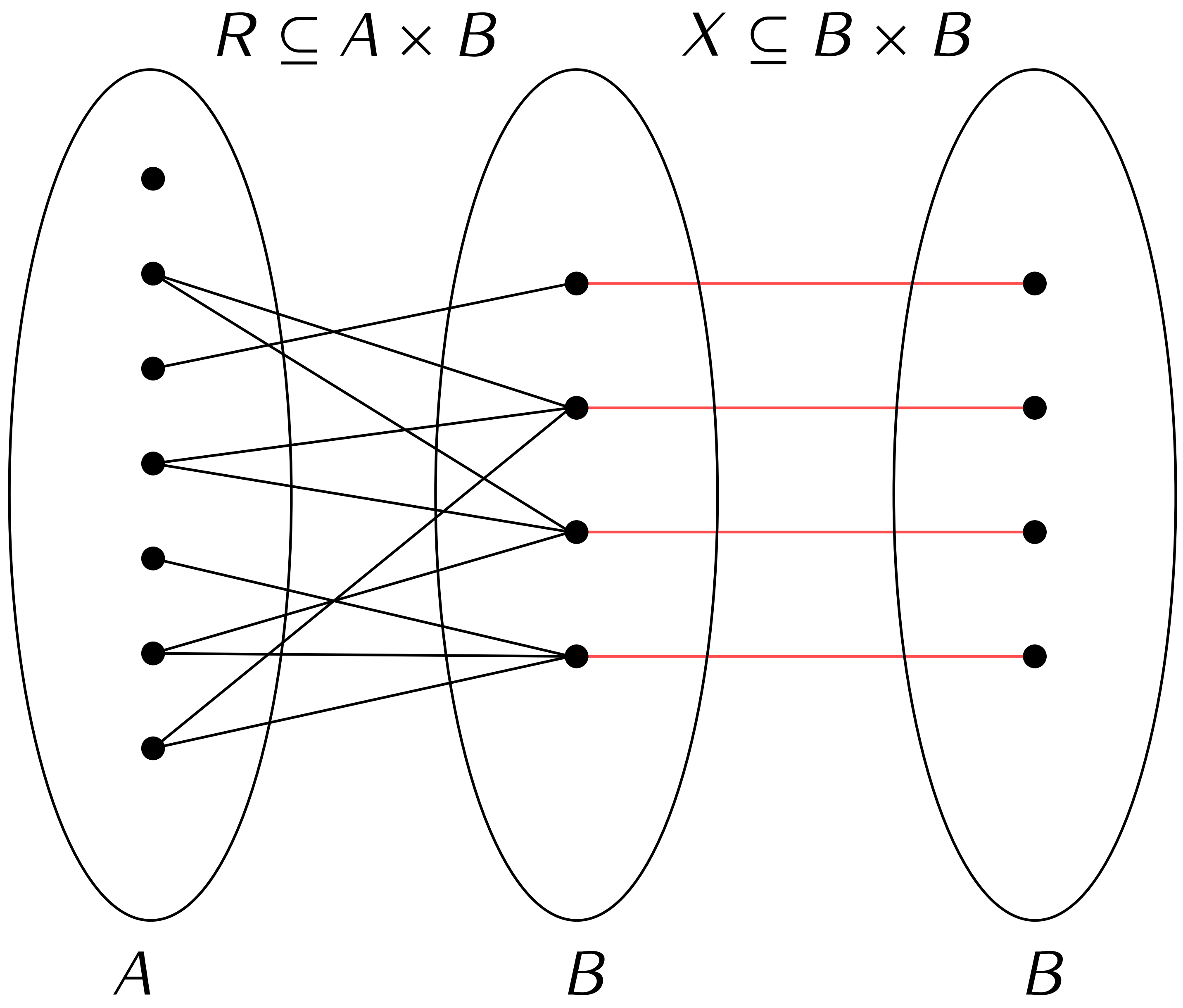


Compute the composition of the given relations. Which pairs does it contain?

Given $R \subseteq A \times B$, can one find a relation X on $B \times B$ such that $X \circ R = R$?

Theorem. Let $\mathrm{Id}_B = \{(b, b) \in B \times B\}$. For all $R \subseteq A \times B$, we have $\mathrm{Id}_B \circ R = R$ and $R \circ \mathrm{Id}_A = R$.

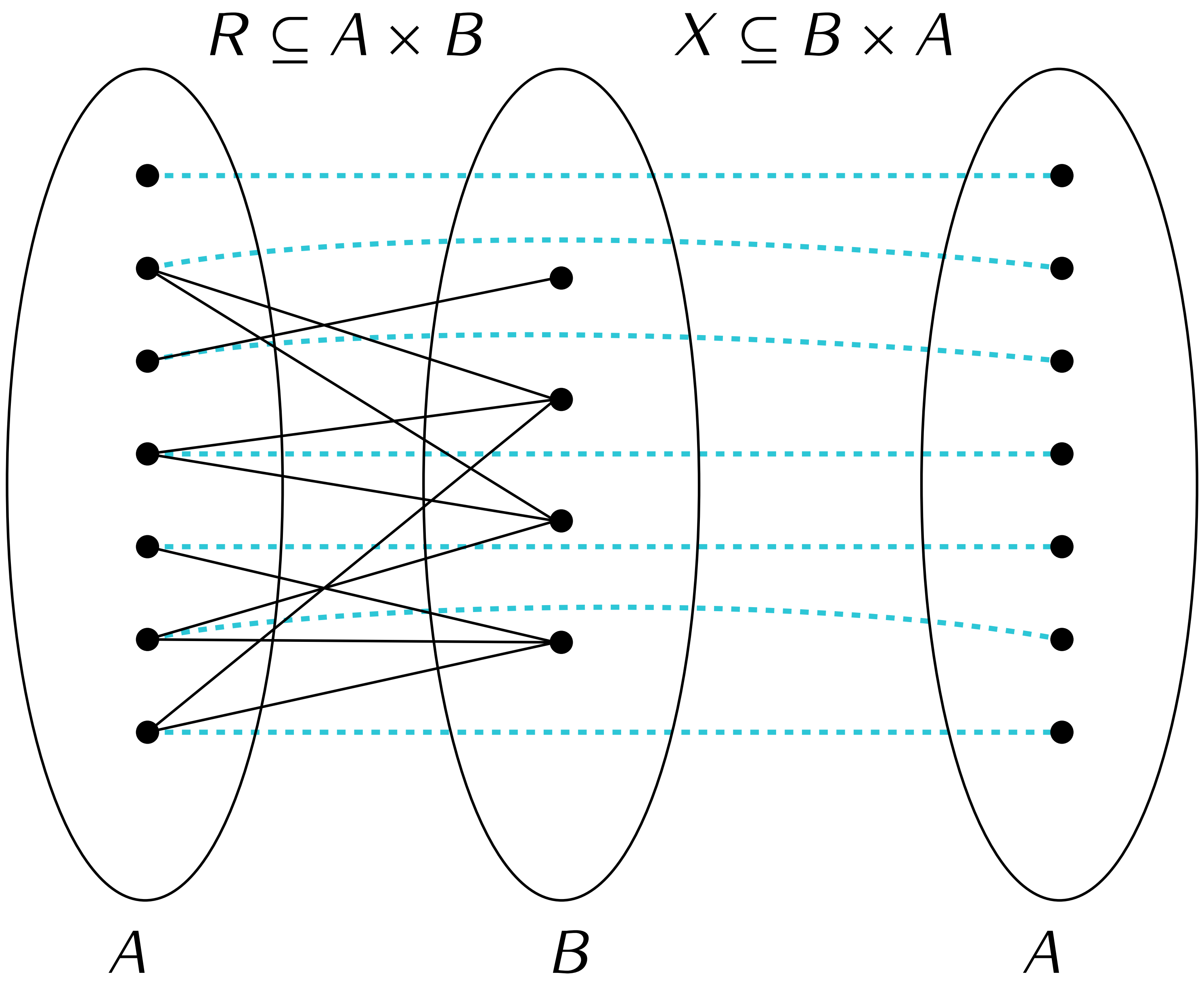
This is actually the function we already know! $\mathrm{Id}_B \colon B \to B$, $\mathrm{Id}_B(x) = x$.



Given $R \subseteq A \times B$, can one find a relation X on $B \times B$ such that $X \circ R = R$?

Theorem. Let $\mathrm{Id}_B = \{(b, b) \in B \times B\}$. For all $R \subseteq A \times B$, we have $\mathrm{Id}_B \circ R = R$ and $R \circ \mathrm{Id}_A = R$.

Given $R \subseteq A \times B$, is there an inverse? A $X \subseteq B \times A$ such that $S \circ X = \mathrm{Id}_B$ and $X \circ S = \mathrm{Id}_A$?



Exercise: which relations have such an inverse?

Given $R \subseteq A \times B$, can one find a relation X on $B \times B$ such that $X \circ R = R$?

Theorem. Let $\mathrm{Id}_B = \{(b, b) \in B \times B\}$. For all $R \subseteq A \times B$, we have $\mathrm{Id}_B \circ R = R$ and $R \circ \mathrm{Id}_A = R$.

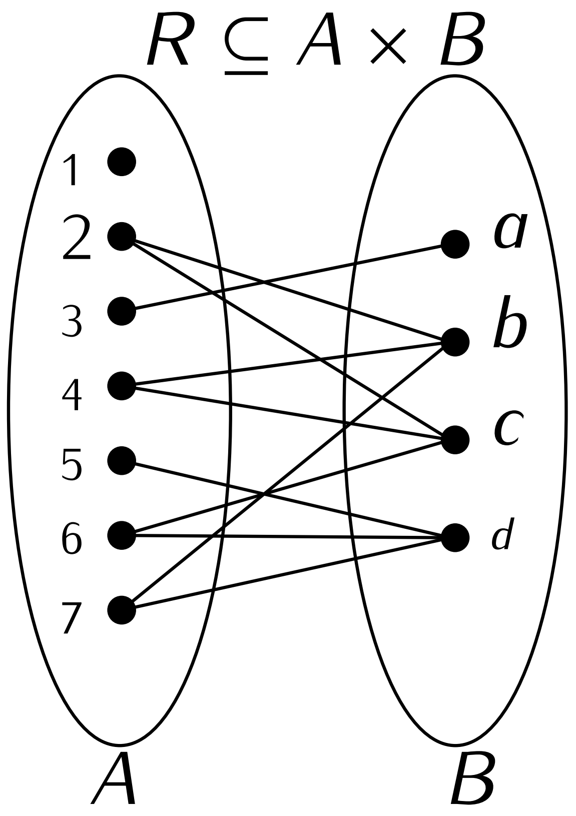
Given $R \subseteq A \times B$, is there an inverse? A $X \subseteq B \times A$ such that $S \circ X = \mathrm{Id}_B$ and $X \circ S = \mathrm{Id}_A$?
No, but:

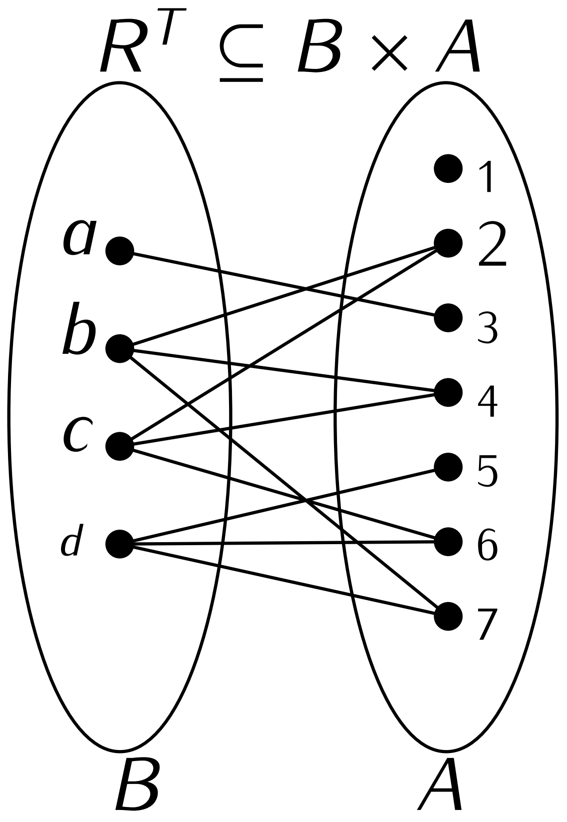
Definition. Let $R \subseteq A \times B$. The transpose of R is

$$R^T = \{(b, a) \in B \times A \mid (a, b) \in R\}.$$

- isChildOf $\subseteq$ People $\times$ People
  isChildOf^T $\subseteq$ People $\times$ People
- isSiblingOf $\subseteq$ People $\times$ People
  isSiblingOf^T $=$?

Remark. It is **not true** that $R \circ R^T = \mathrm{Id}_B$ holds in general!
But **if** R has an inverse, then R^T is **the** inverse.





Algebra: study of **operations** and **equations** on *stuff*

Number theory

Boolean algebra

Numbers: $+, \times, 1/x, 1, 0$

Matrices: $+, \times, M^{-1}, I, 0$

Sets: $\cap, \cup, \times, \Delta, \emptyset, \ldots$

Functions: $\circ, f^{-1}, \mathrm{Id}_A$

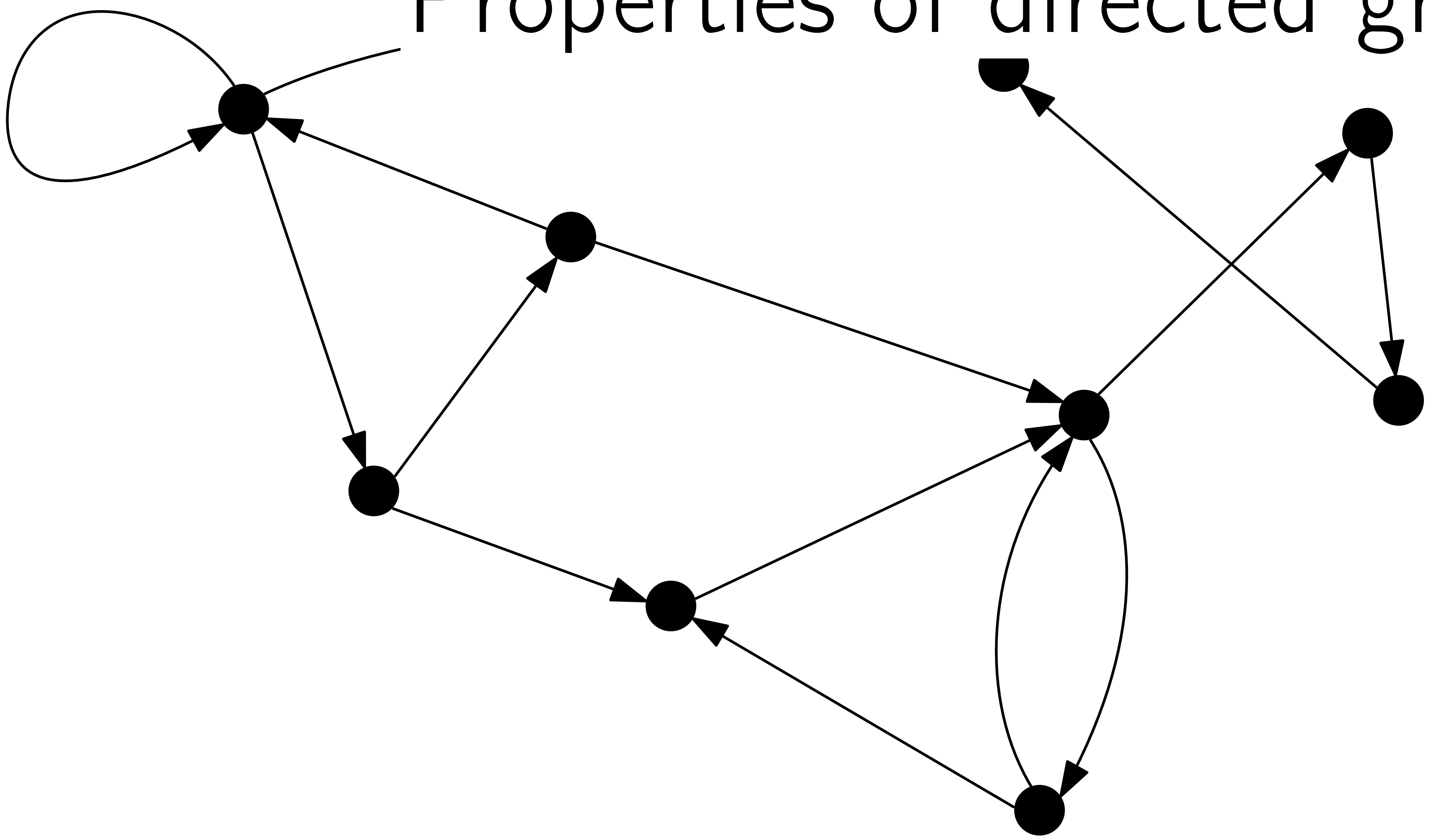
Booleans: $\wedge, \vee, \Rightarrow, \neg, \top, \bot$

Relations: $\circ, R^T, \mathrm{Id}, \cup, \cap, \times, \ldots$

Linear algebra

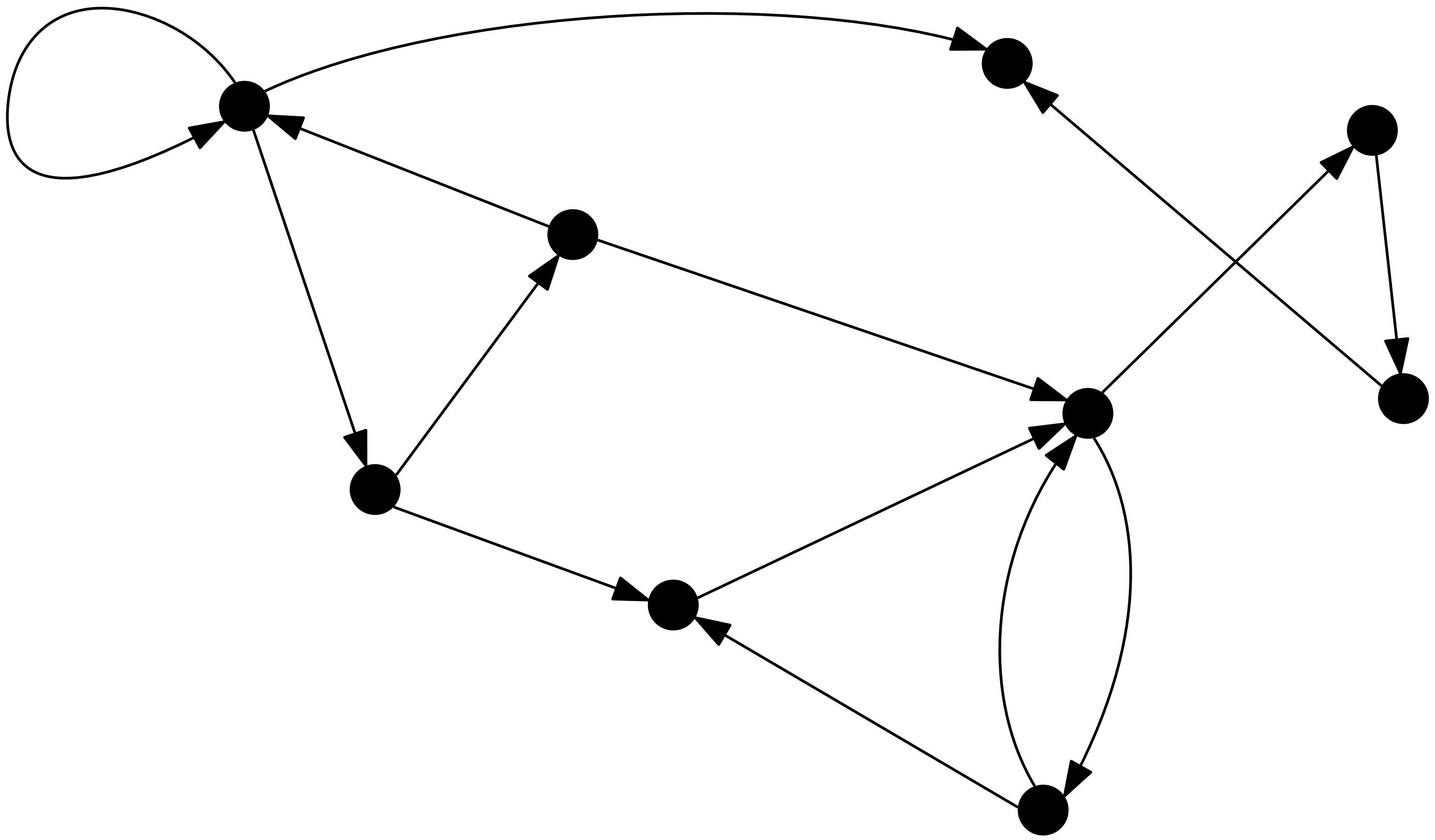
Relational algebra

Properties of directed graphs



Definition. A relation $R \subseteq A \times A$ is:

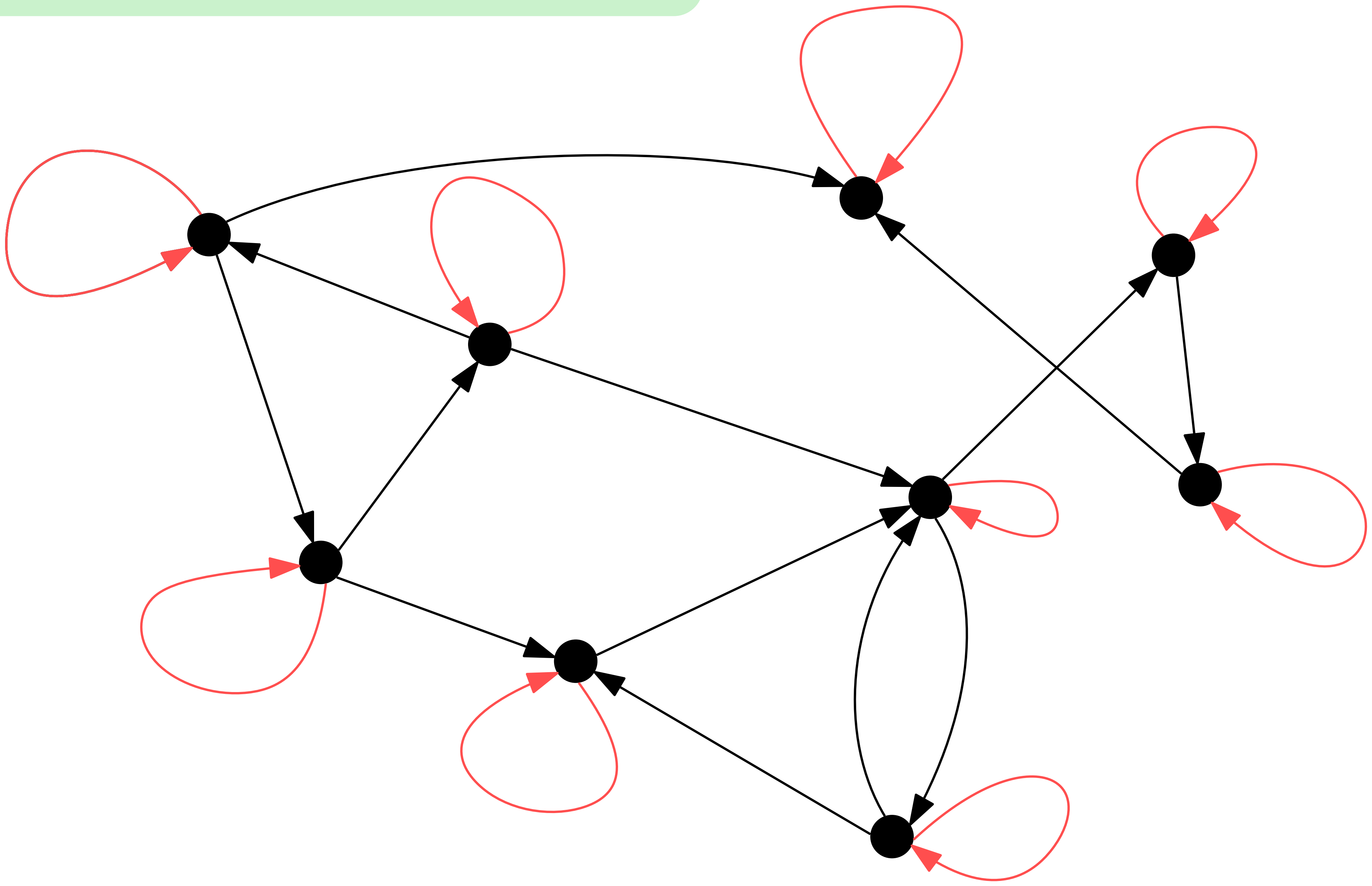
- reflexive: if for all $a \in A$, $(a, a) \in R$
- antireflexive: if for all $a \in A$, $(a, a) \notin R$
- symmetric: if for all $a, b \in A$, if $(a, b) \in R$, then $(b, a) \in R$
- antisymmetric: if for all $a, b \in A$, if $(a, b) \in R$ and $a \neq b$ then $(b, a) \notin R$
- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.



Definition. A relation $R \subseteq A \times A$ is:

- reflexive: if for all $a \in A$, $(a, a) \in R$
- antireflexive: if for all $a \in A$, $(a, a) \notin R$

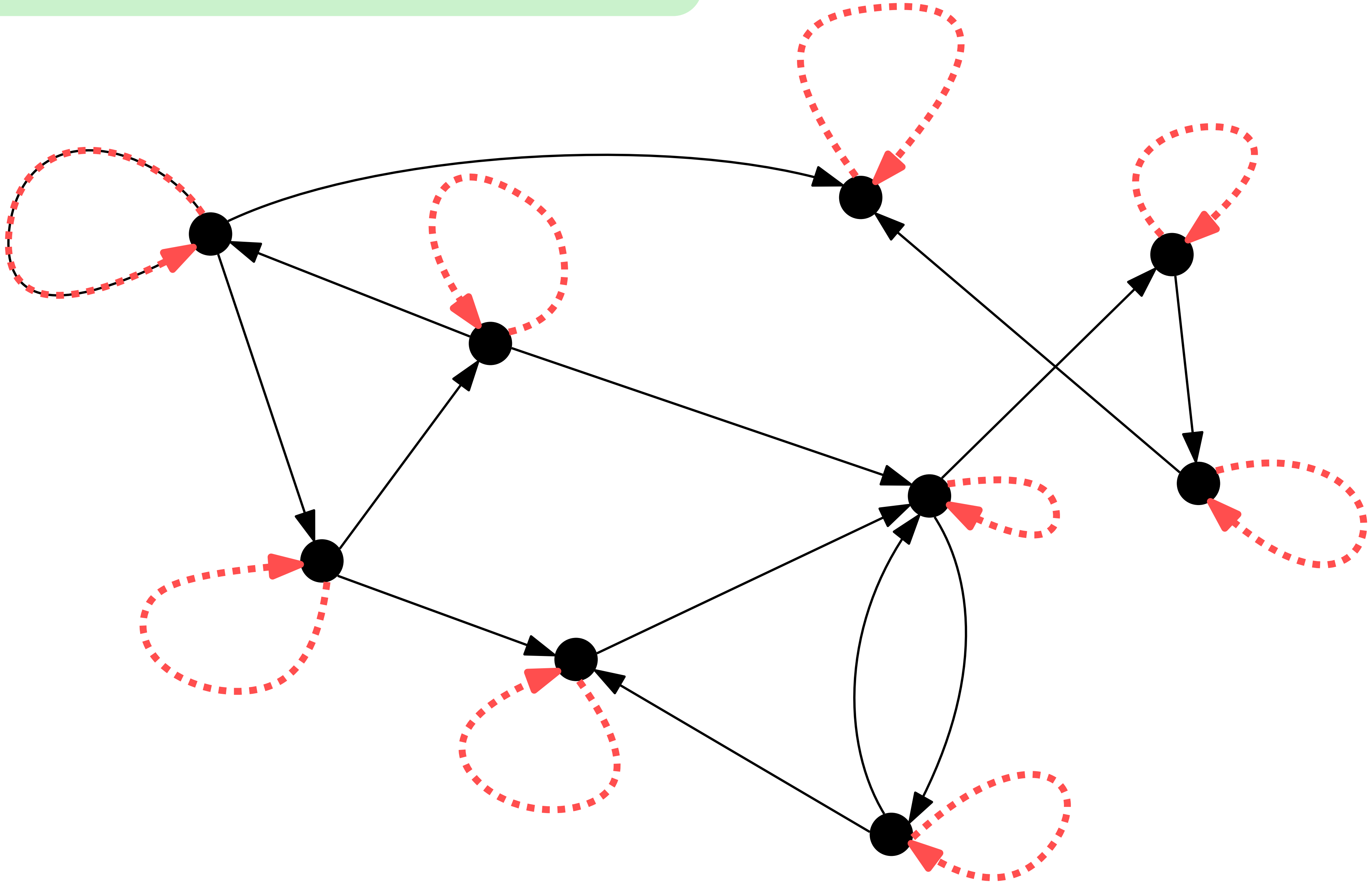
Examples. $x \leq y$, $x = y$



Definition. A relation $R \subseteq A \times A$ is:

- reflexive: if for all $a \in A$, $(a, a) \in R$
- antireflexive: if for all $a \in A$, $(a, a) \notin R$

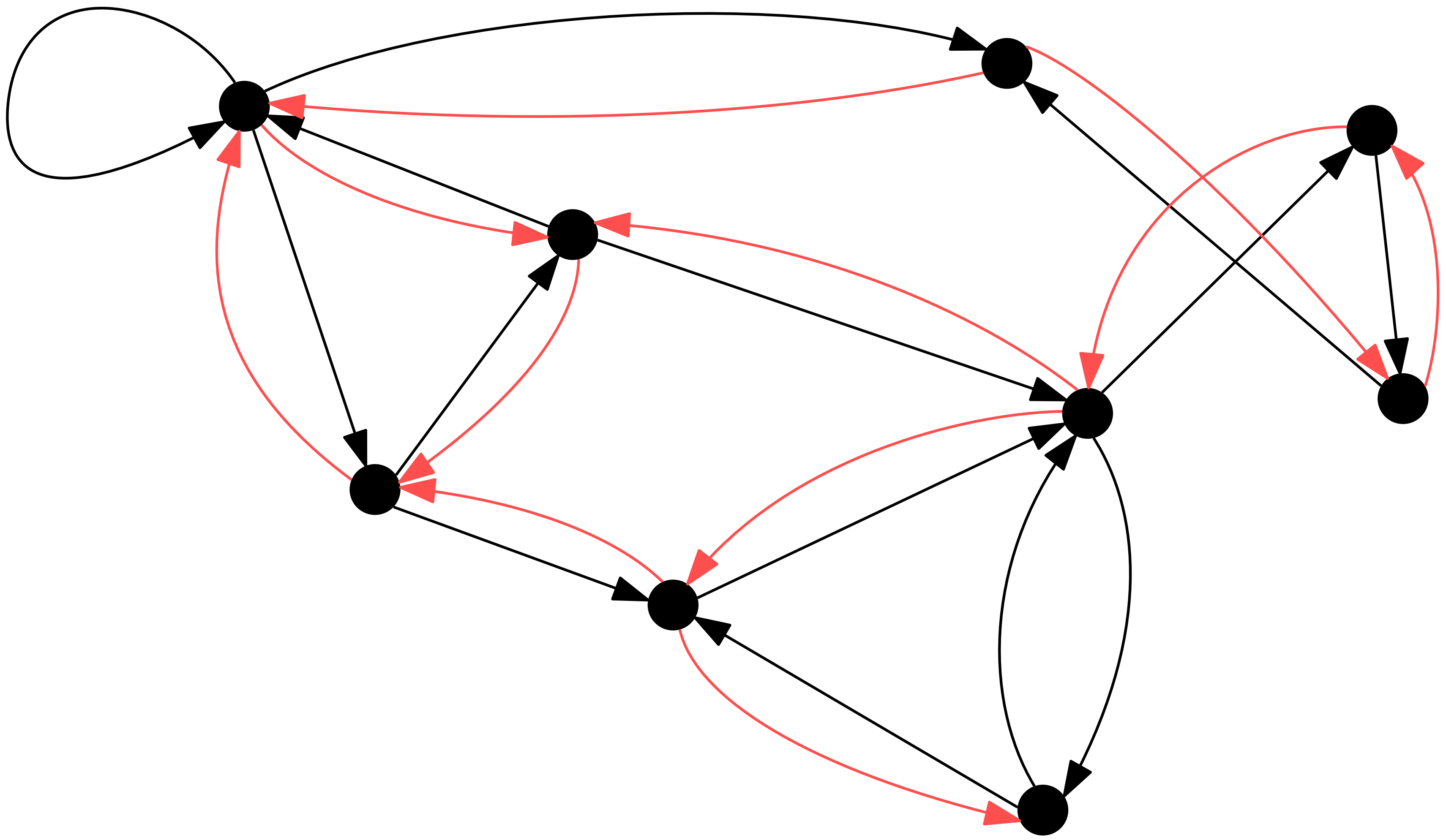
Examples. $x < y$, $x \neq y$, "is child of"



Definition. A relation $R \subseteq A \times A$ is:

- symmetric: if for all $a, b \in A$, if $(a, b) \in R$, then $(b, a) \in R$
- antisymmetric: if for all $a, b \in A$, if $(a, b) \in R$ and $a \neq b$ then $(b, a) \notin R$

Examples. $x = y$, "is sibling of"

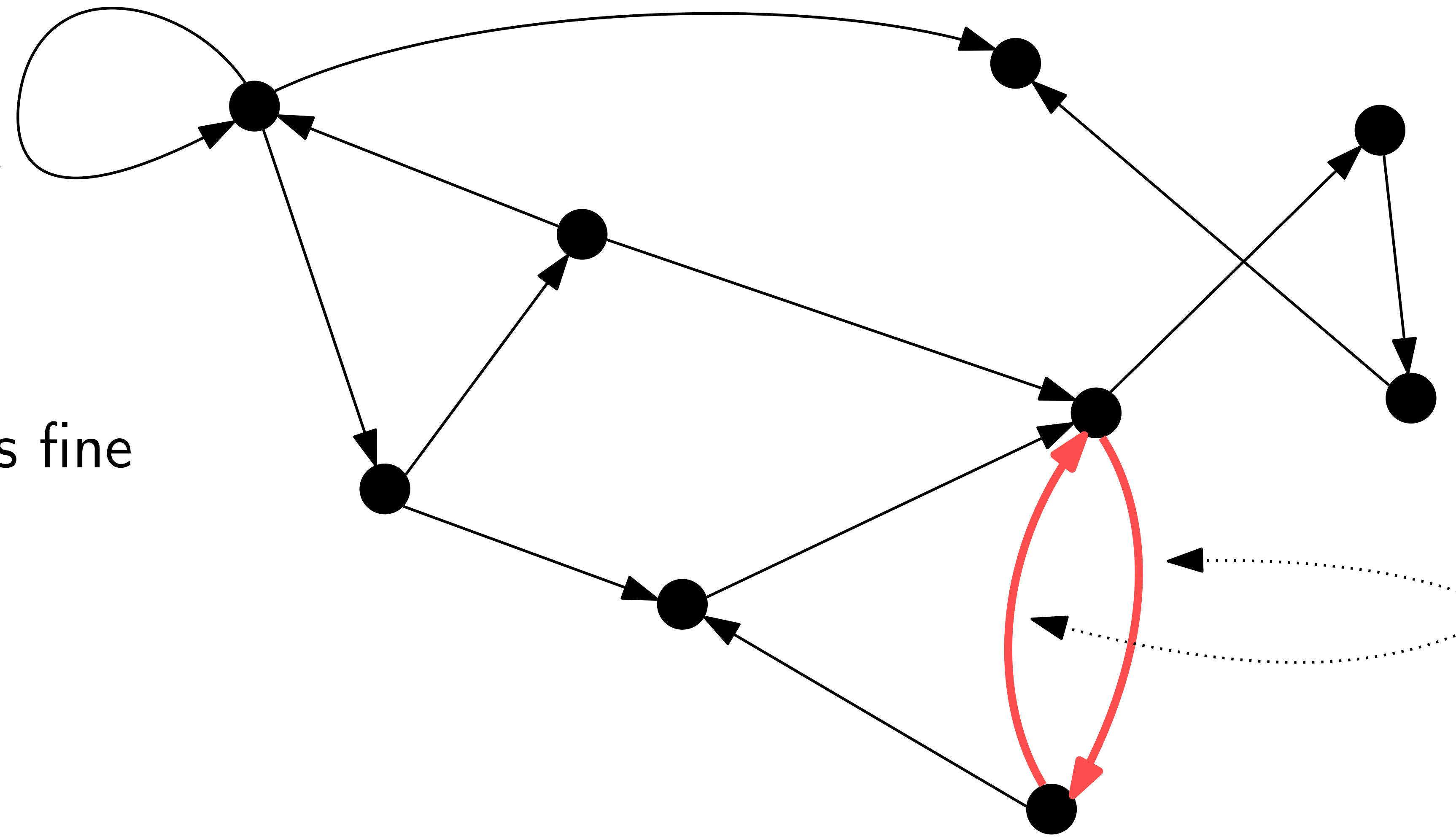


Definition. A relation $R \subseteq A \times A$ is:

- symmetric: if for all $a, b \in A$, if $(a, b) \in R$, then $(b, a) \in R$
- antisymmetric: if for all $a, b \in A$, if $(a, b) \in R$ and $a \neq b$ then $(b, a) \notin R$

Examples. $x < y$, "is child of"

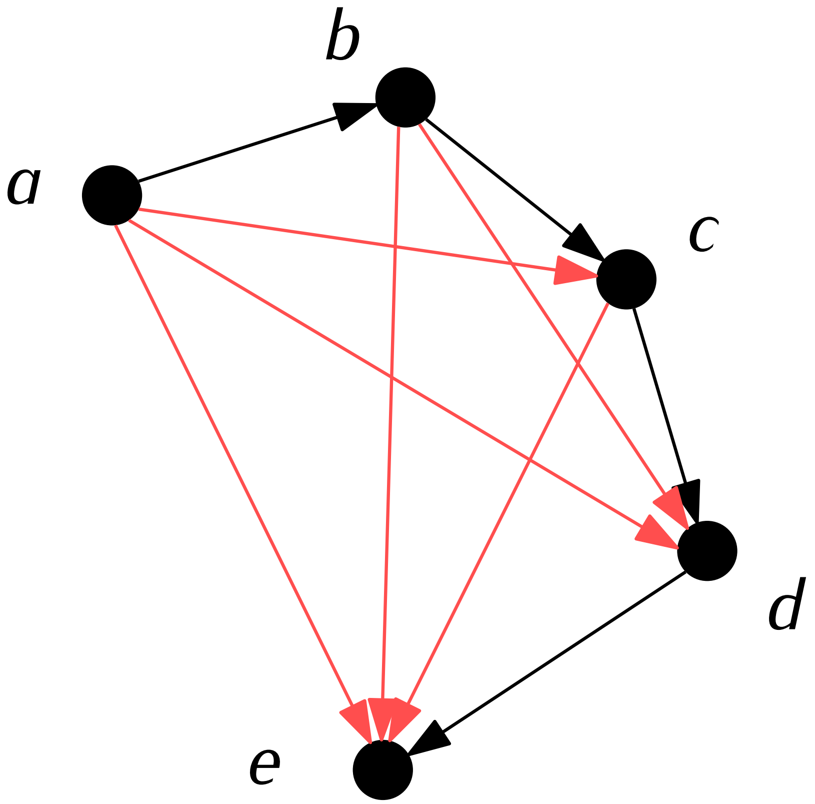
This edge is fine



One of these two edges cannot be there

Definition. A relation $R \subseteq A \times A$ is:

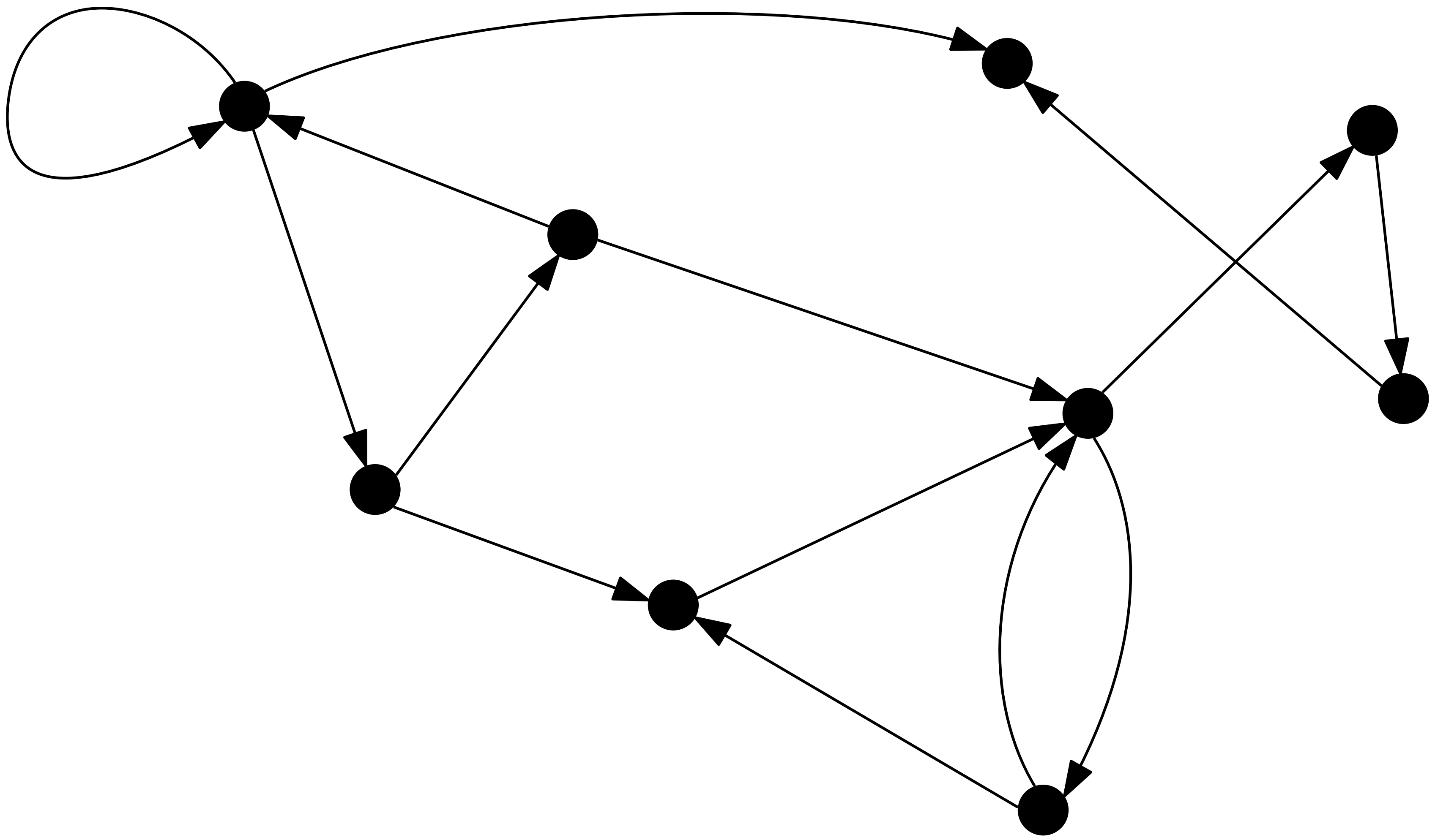
- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.



Definition. A relation $R \subseteq A \times A$ is:

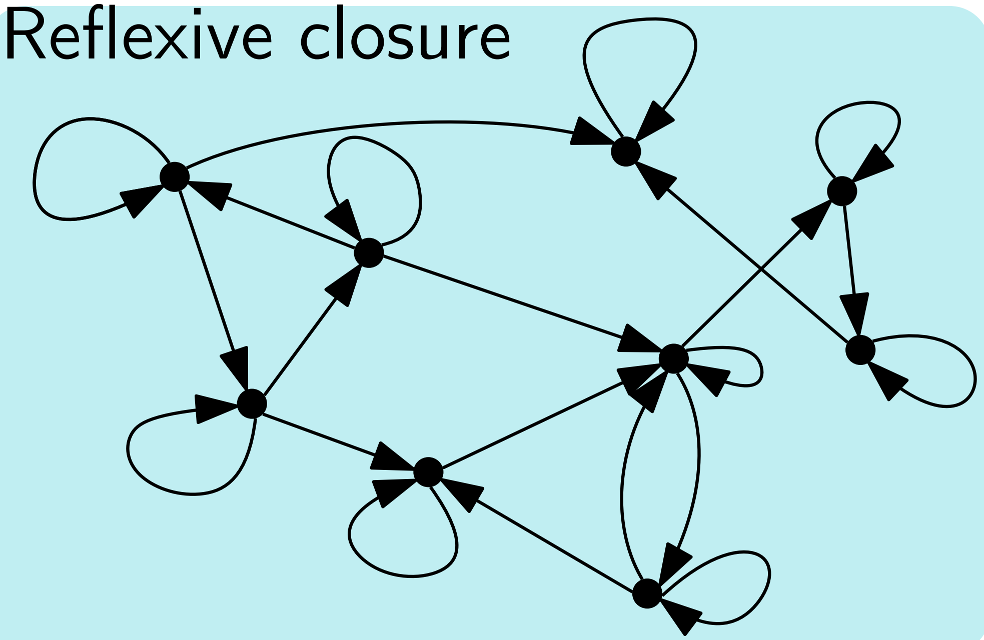
- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.

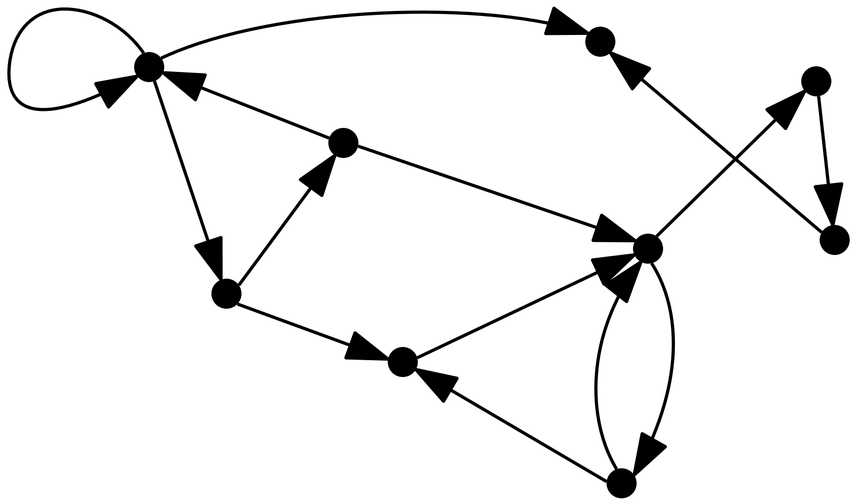
Examples. $x < y$, $x = y$, "x is an ancestor of y"



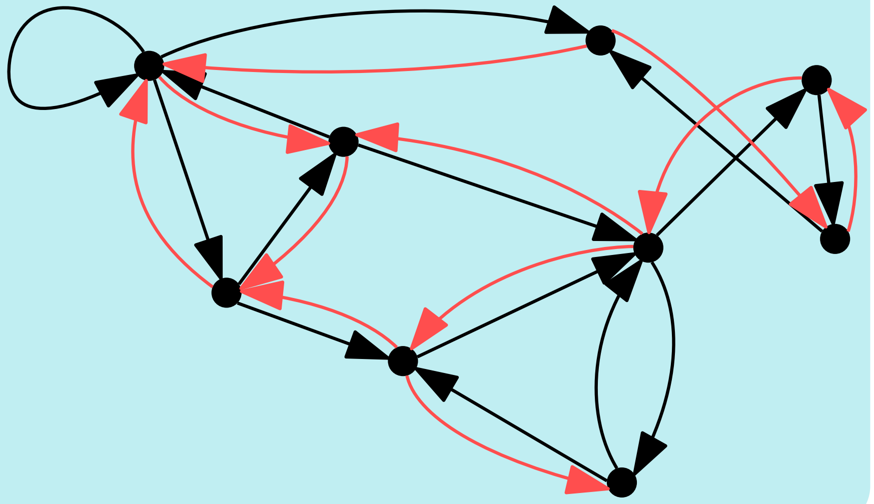
Idea of "closure": take something that does not satisfy a property, and make it bigger until it does

Reflexive closure





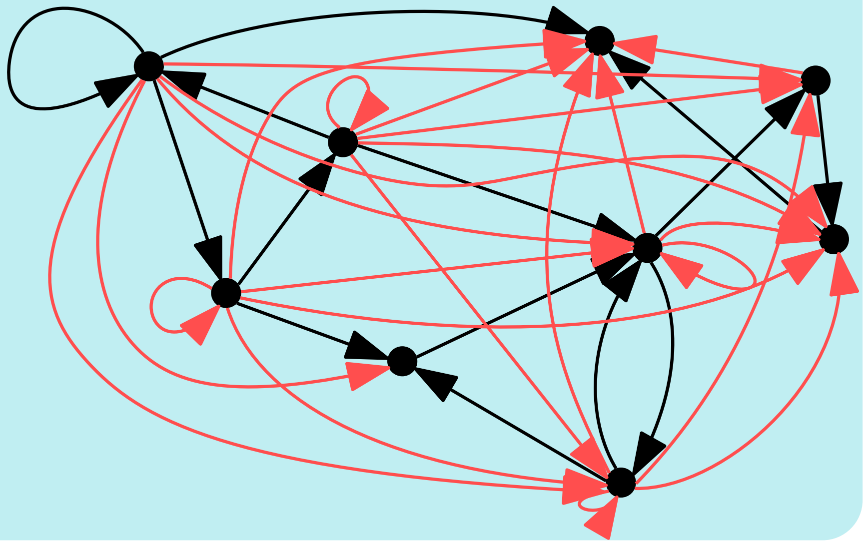
Symmetric closure



Reflexive closure of $R = R \cup \mathrm{Id}_A$

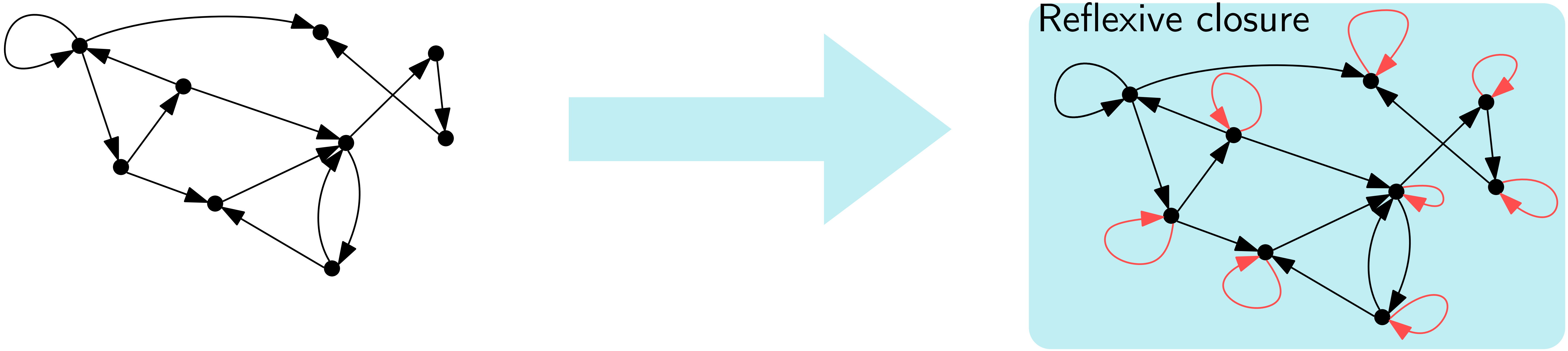
Symmetric closure of $R = R \cup R^T$

Transitive closure



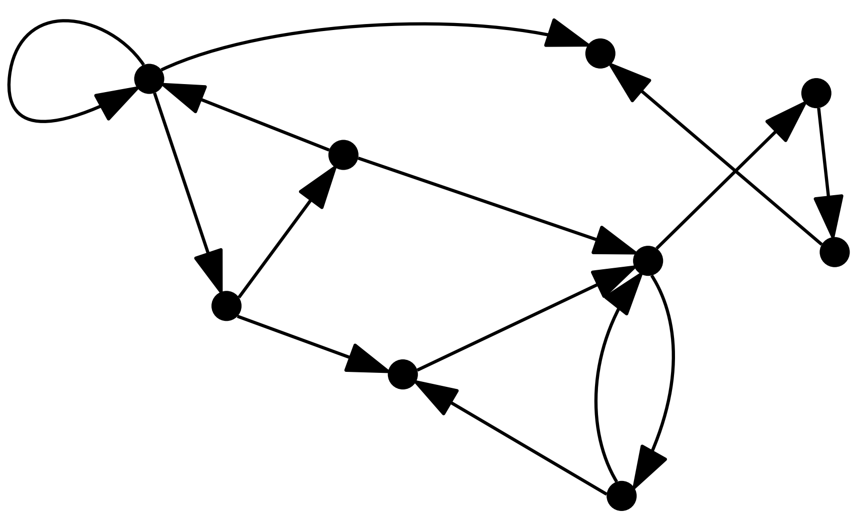
Transitive closure of $R =$
$R \cup (R \circ R) \cup (R \circ R \circ R) \cup \ldots$

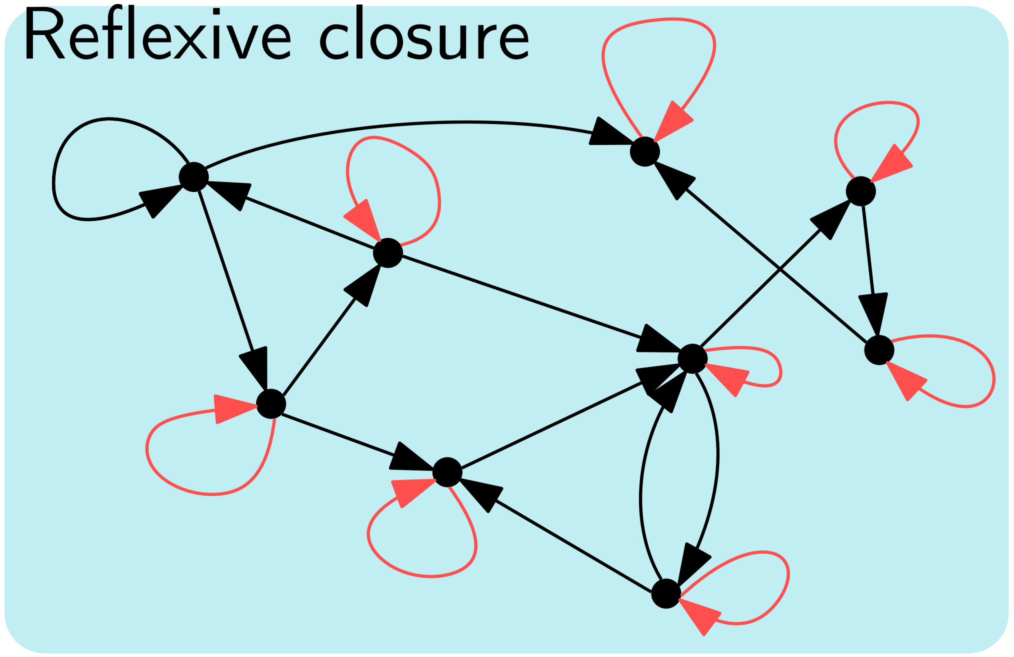
if $(a, b), (b, c), (c, d) \in R$, then (a, d) in the transitive closure



Reflexive closure of $R = R \cup \mathrm{Id}_A$

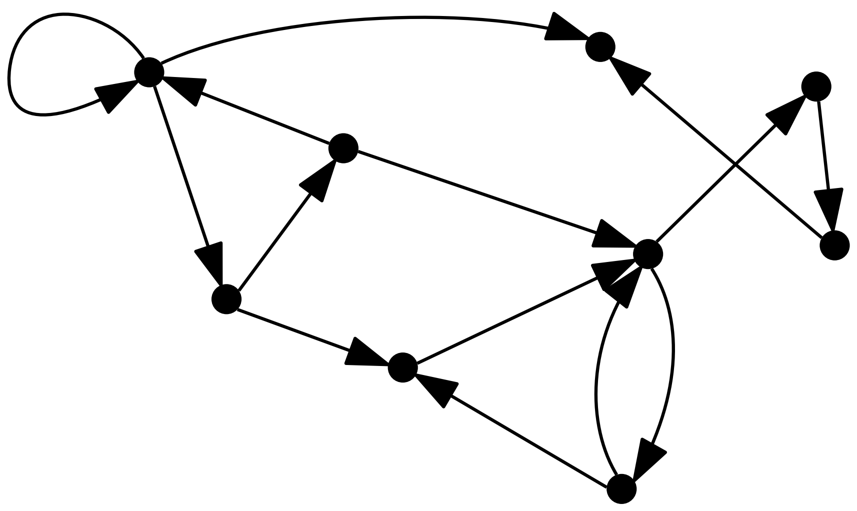
```
def reflexive_closure(A, R):
  T = R.copy() # make a copy of R
  # ???
  # ???
  return T
```

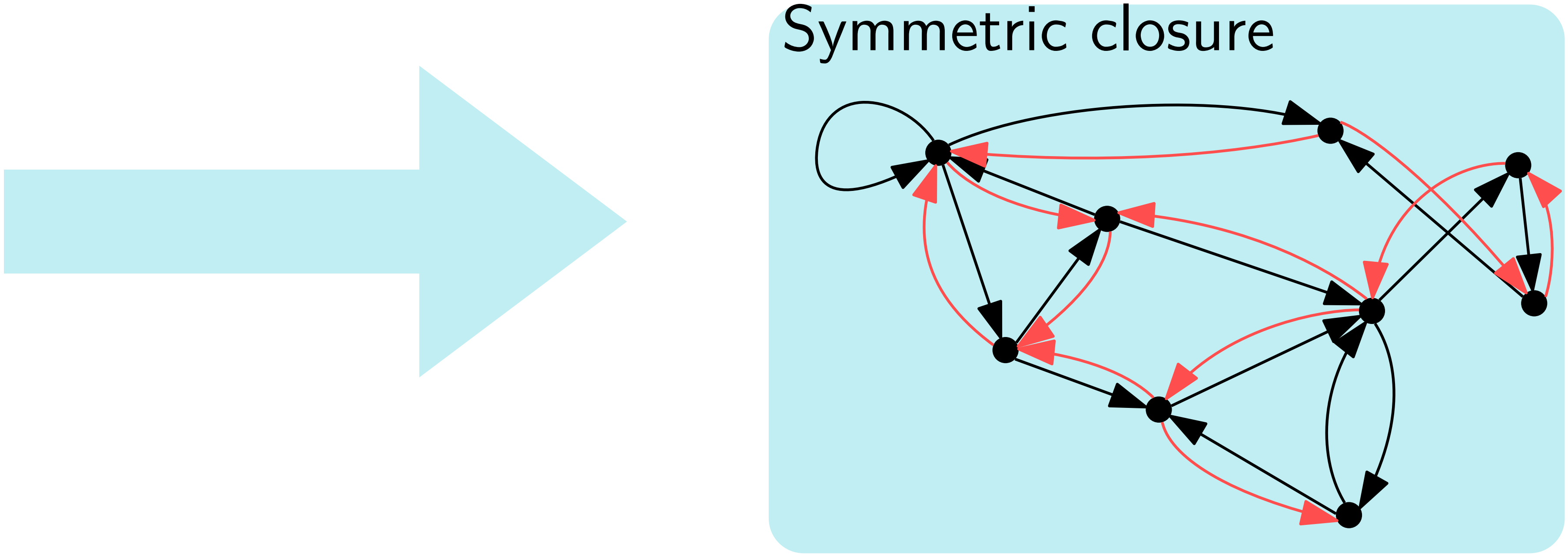




Reflexive closure of $R = R \cup \mathrm{Id}_A$

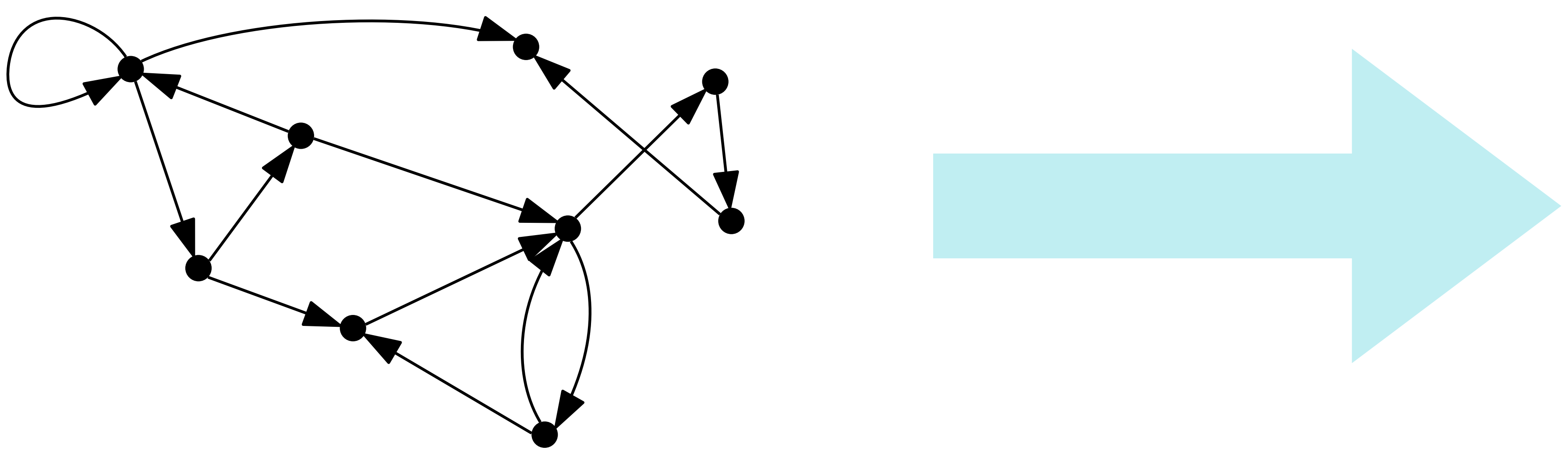
```
def reflexive_closure(A, R):
  T = R.copy() # make a copy of R
  for a in A:
    T.add((a,a))
  return T
```

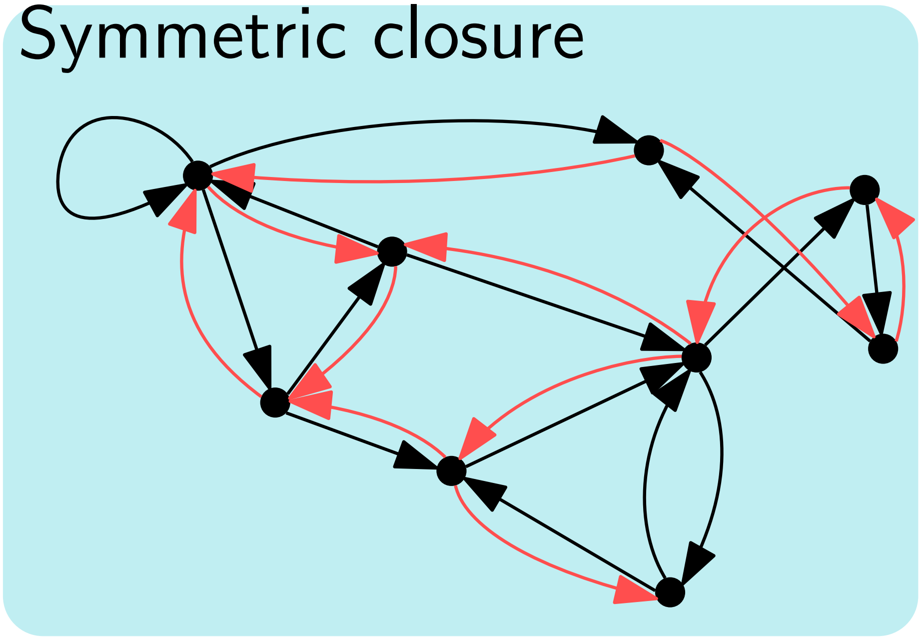




Symmetric closure of $R = R \cup R^T$

```
def symmetric_closure(R):
  T = R.copy() # make a copy of R
  # ???
  # ???
  return T
```



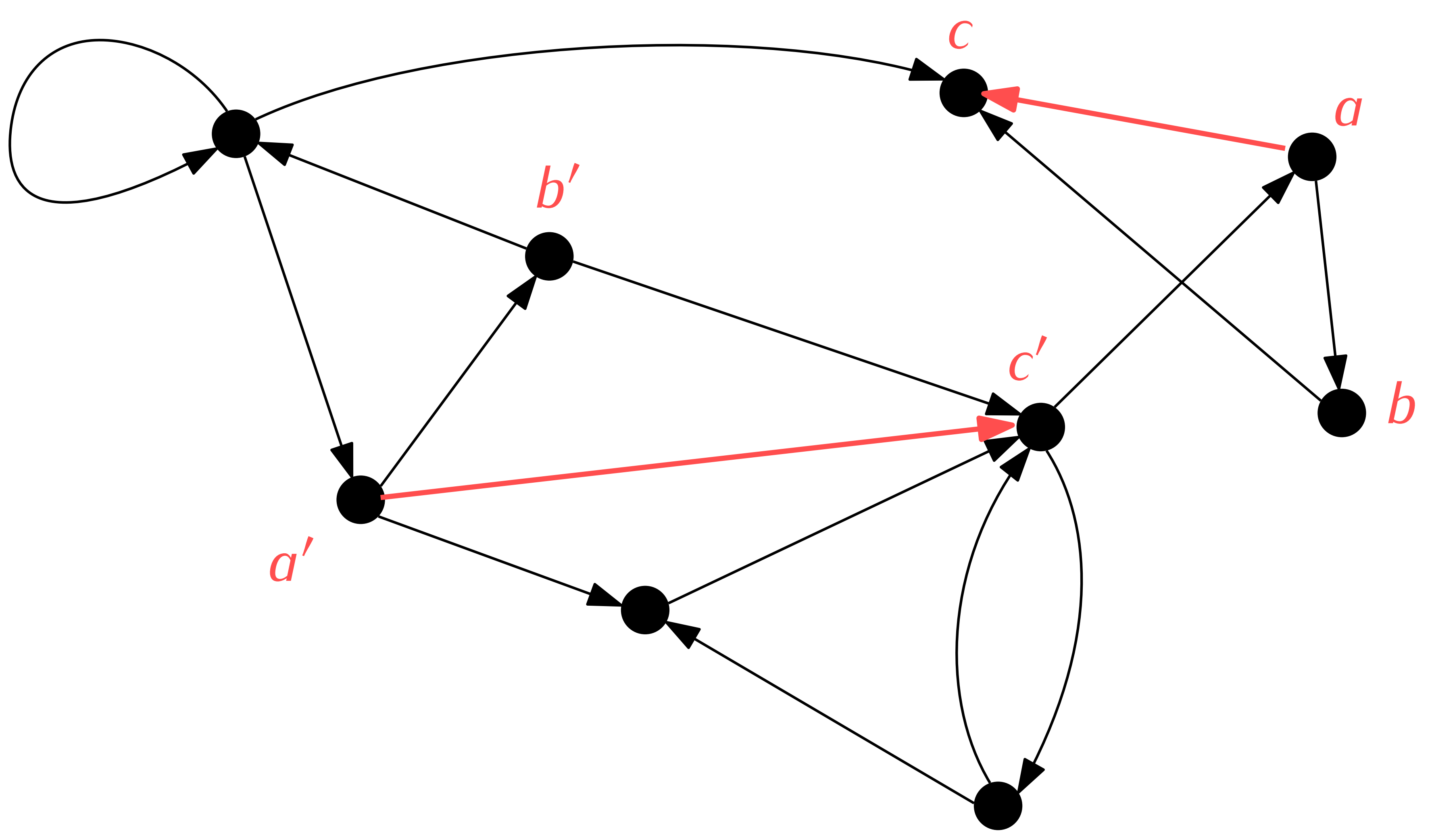


Symmetric closure of $R = R \cup R^T$

```
def symmetric_closure(R):
  T = R.copy() # make a copy of R
  for (a,b) in R:
    T.add((b,a))
  return T
```

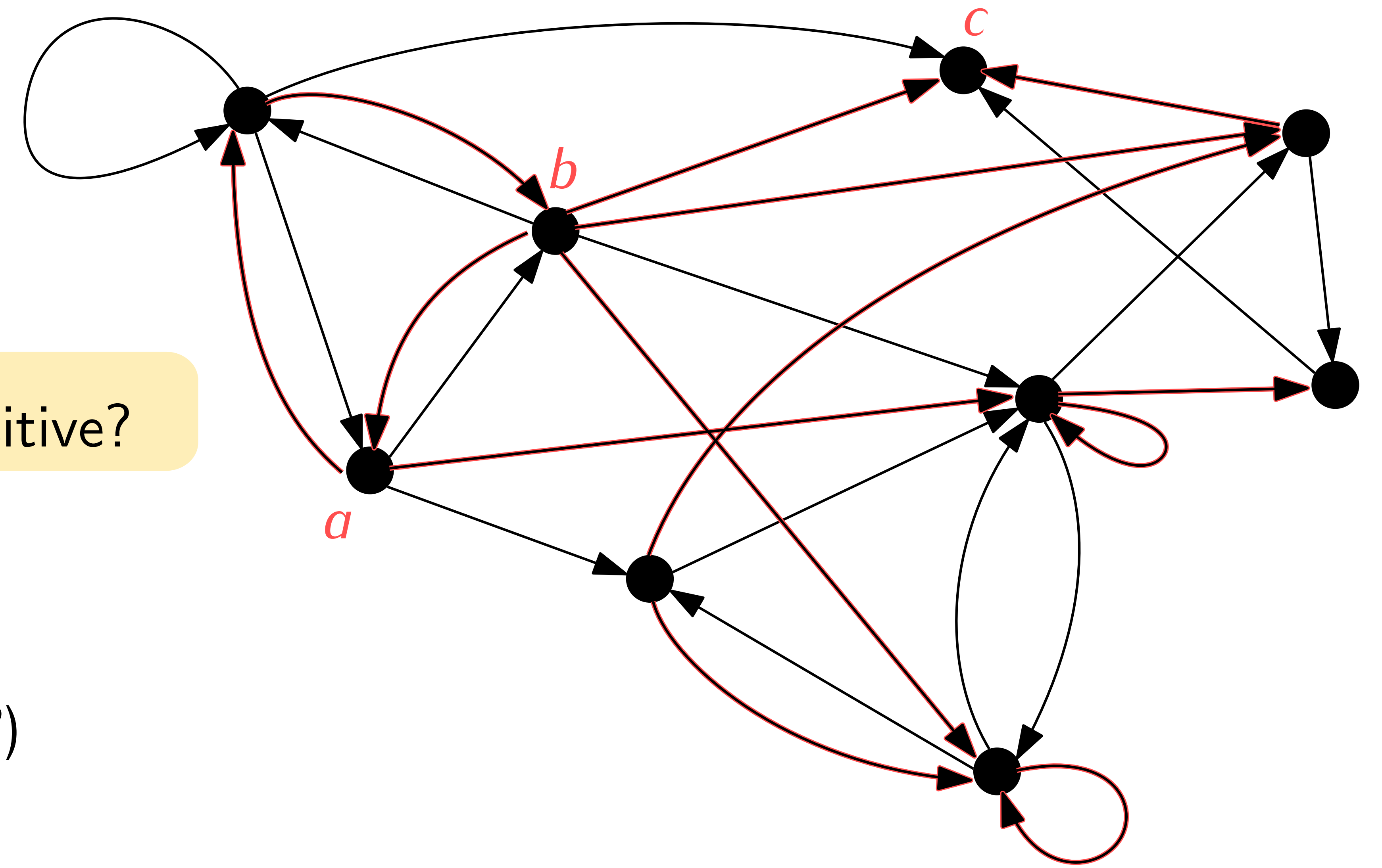
Definition. A relation $R \subseteq A \times A$ is:

- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.



Definition. A relation $R \subseteq A \times A$ is:

- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.

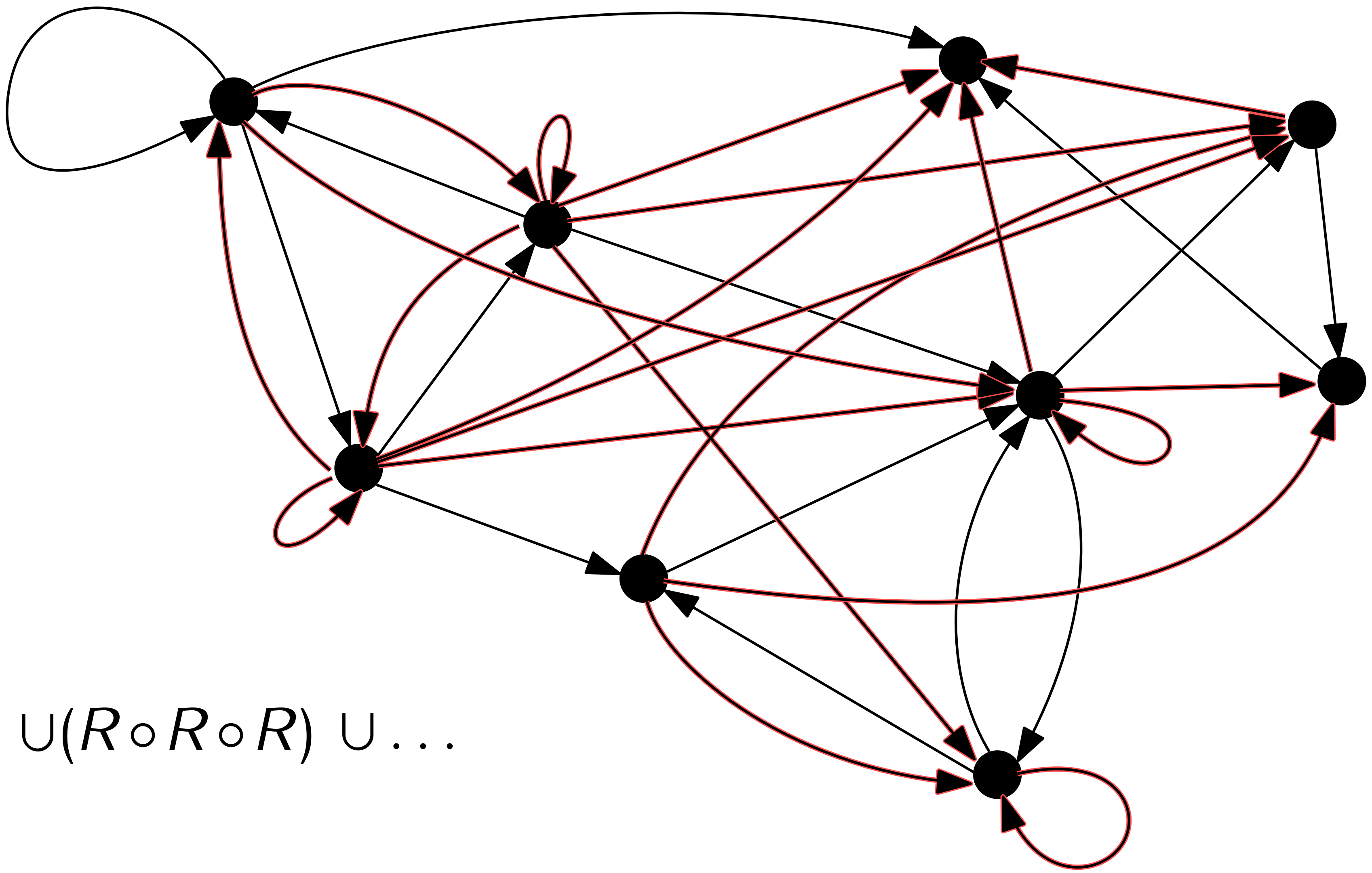


Question. Is T transitive?

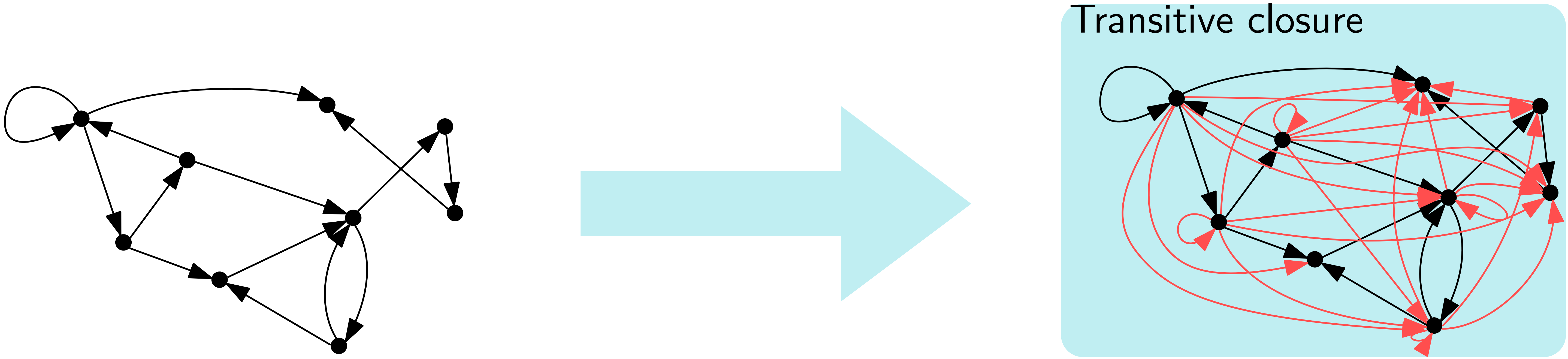
$$T = R \cup (R \circ R)$$

Definition. A relation $R \subseteq A \times A$ is:

- transitive: if for all $a, b, c \in A$, if $(a, b) \in R$ and $(b, c) \in R$ then $(a, c) \in R$.



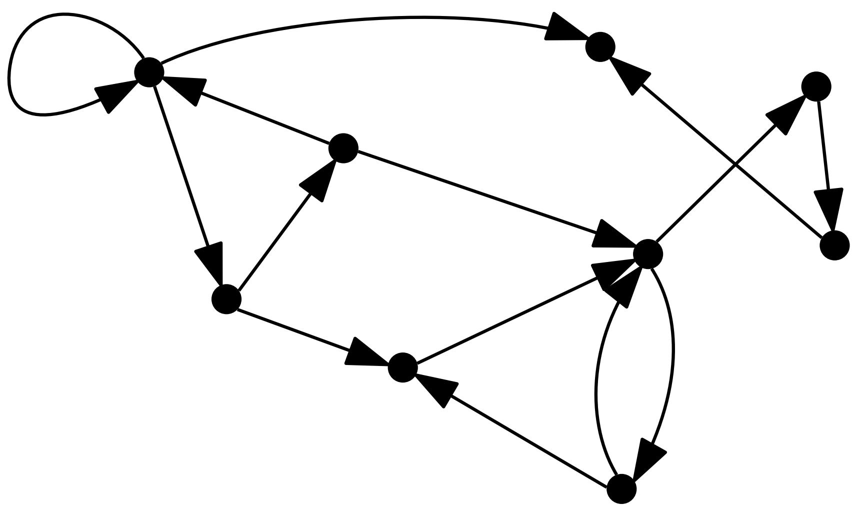
$$T = R \cup (R \circ R) \cup (R \circ R \circ R) \cup \ldots$$

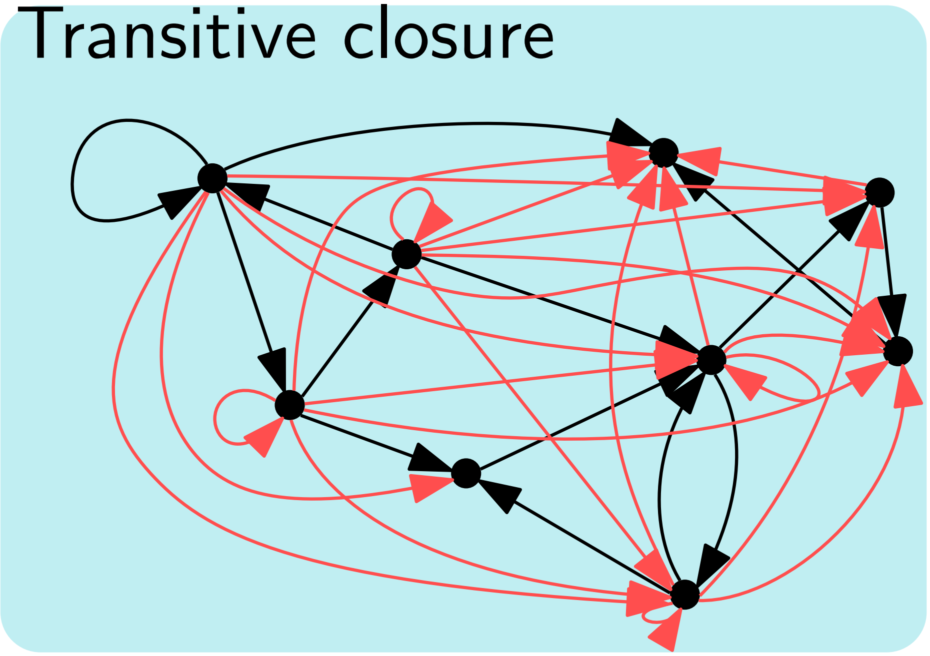



```
def transitive_closure(R):
  T = R.copy()
  keepGoing = True
  while keepGoing:
    ???
    ???
    ???
    ???
    ???
    ???
  return T
```

Transitive closure of $R =$
$R \cup (R \circ R) \cup (R \circ R \circ R) \cup \ldots$

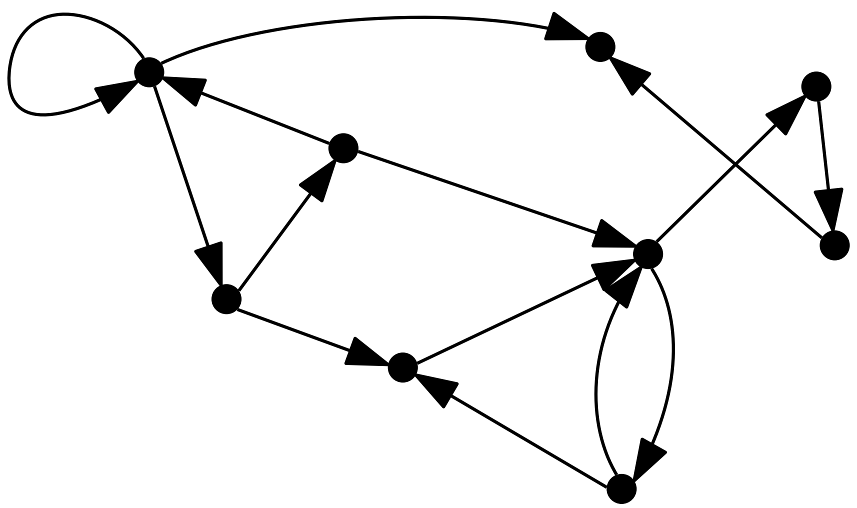
Write code that from R, computes all pairs (a, b) such that there is a *path* from a to b

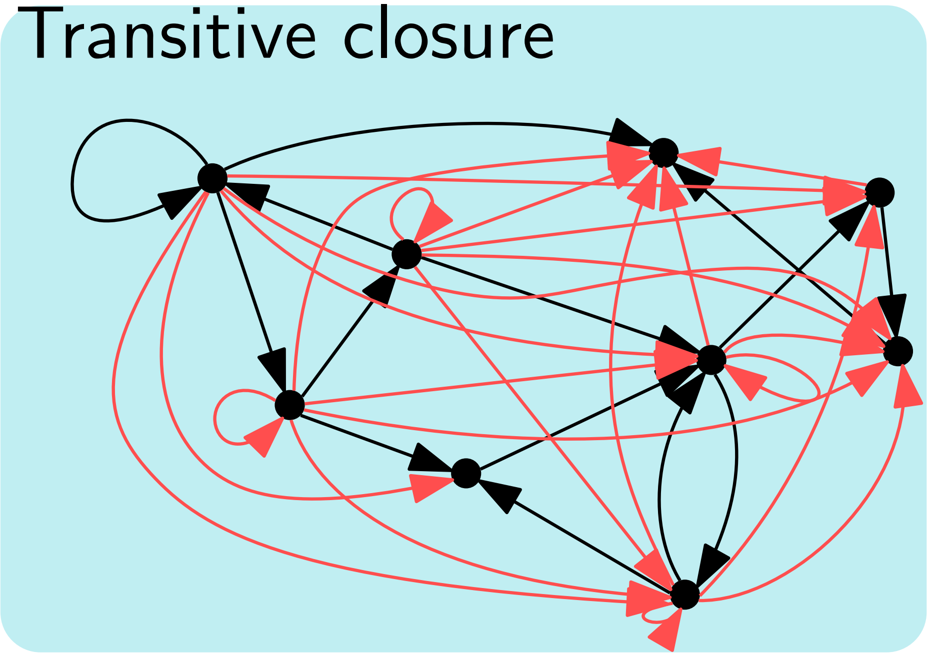




Transitive closure of $R =$
$R \cup (R \circ R) \cup (R \circ R \circ R) \cup \ldots$

```
def transitive_closure(R):
  T = R.copy()
  keepGoing = True
  while keepGoing:
    keepGoing = False
    for (a,b) in T:
      for (c,d) in R:
        if b==c and (a,d) not in T:
          T.add((a,d))
          keepGoing = True
  return T
```





Transitive closure of $R =$
$R \cup (R \circ R) \cup (R \circ R \circ R) \cup \ldots$

Compute the transitive closure of $\{(0,1),(0,2),(1,4),(2,4),(0,3)\}$:

- $\{(0,4)\}$
- $\{(0,1),(0,2),(1,4),(2,4),(0,3),(0,4)\}$
- $\{(0,1),(0,2),(1,4),(2,4),(0,3),(1,2)\}$
- $\{(0,1),(0,2),(1,4),(2,4),(0,3),(3,4)\}$

1. Draw the relation as a graph (how many vertices?, how many edges?)
2. Add the missing edges
3. Then compare your solution to the given solutions

- What a relation is
- How to draw a relation $R \subseteq A \times B$
- How to compute the composition and transpose of binary relations
- The basic properties
  (anti)reflexivity, (anti)symmetry, transitivity
- How to compute the closures