Class starts after this song… and some more

King Gnu – Specialz (2023)
requested by Jacob Lee (TA-of-CM7)

I am a sophomore double majored in CompSci and Mechanical Engineering. I am interested in robotics. In my free time, I enjoy playing games (board games and video games). I am adopted from Korea. I have anaphylactic food allergies to nuts, seeds, and lentils.

Duke

Bipartite Matching

- Matching, but with the underlying graph already bipartite:
$$G = (L \cup R, E)$$
- Matching TAs/residents/interns with positions
- Matching computing tasks with machine cycles/slots
- … but **NOT**: ~~matching men with women~~
- These applications often involves preferences and some additional ideas of what an *optimal* matching looks like (stable, fair, …)
- We will just discuss whether a perfect matching exists

Hall's theorem (for the special case $|L| = |R|$)

- Theorem.
  A bipartite graph $G = (L \cup R, E)$ has a perfect matching if and only if

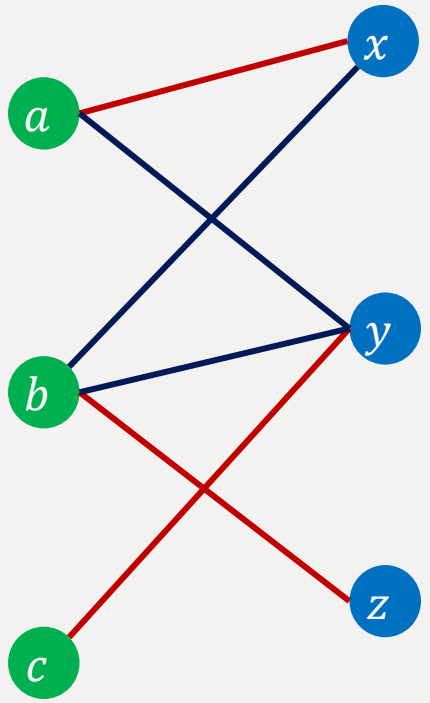
Hall's condition

for any subset $S \subseteq L$, the size of the image of S w.r.t. E, $N(S)$, is at least as large as $|S|$.

  - $N(S)$ is just the set of all vertices in R that are connected to some vertices in L by edges in E

Terminology Practice

S	$\lvert S\rvert$	$N(S)$	$\lvert N(S)\rvert$	Hall's condition satisfied?
$\{a\}$	1	$\{x, y\}$	2	Yes
$\{b\}$	1	$\{x, y, z\}$	3	Yes
$\{c\}$	1	$\{y\}$	1	Yes
$\{a, b\}$	2	$\{x, y, z\}$	3	Yes
$\{a, c\}$	2	$\{x, y\}$	2	Yes
$\{b, c\}$	2	$\{x, y, z\}$	3	Yes
$\{a, b, c\}$	3	$\{x, y, z\}$	3	Yes



Proof of Hall's theorem (1)

- Part 1. perfect matching exists ⇒ Hall's condition
  - Simple prove by contradiction: assume Hall's condition doesn't hold
  - Then there exists a subset $S \subseteq L$ such that $|N(S)| < |S|$
  - But if a perfect matching exists, every vertex in S needs to match with a distinct vertex in $N(S)$, which means $|N(S)| \geq |S|$, a contradiction

Proof of Hall's theorem (2)

- Part 2. Hall's condition ⇒ perfect matching exists
- The set of balanced bipartite graphs, $\mathcal{B}$, can be alternatively defined as:
  - **Base Case:** $G = (\emptyset \cup \emptyset, \emptyset) \in \mathcal{B}$ (the "empty graph")
  - **Constructor Case:** If $G = (L \cup R, E) \in \mathcal{B}$, then:
    - $G = ((L \cup \{l\}) \cup (R \cup \{r\}), E \cup E') \in \mathcal{B}$
      ("adding one vertex to each side, then add some edges")
    - … this is hard to work with! We will instead resort to a (weak) induction

Proof of Hall's theorem (2)

- Part 2. Hall's condition ⇒ perfect matching exists for all $|L| = |R| = n$
- **Proof**:
  - **Base Case.** $n = 0$. For $G = (\emptyset \cup \emptyset, \emptyset)$, $\emptyset$ is a perfect matching (yes it is)
  - **Inductive Hypothesis.** Assume the theorem holds for $n = k$ for $k \geq 0$.
  - **Induction Step.** Consider a graph $G = (L \cup R, E)$ with $|L| = |R| = k + 1$.
    - Case 1. Hall's condition is loosely satisfied (equation never holds).
    - Case 2. Hall's condition is tightly satisfied (equation holds for some S).

Proof of Hall's theorem (2-1)

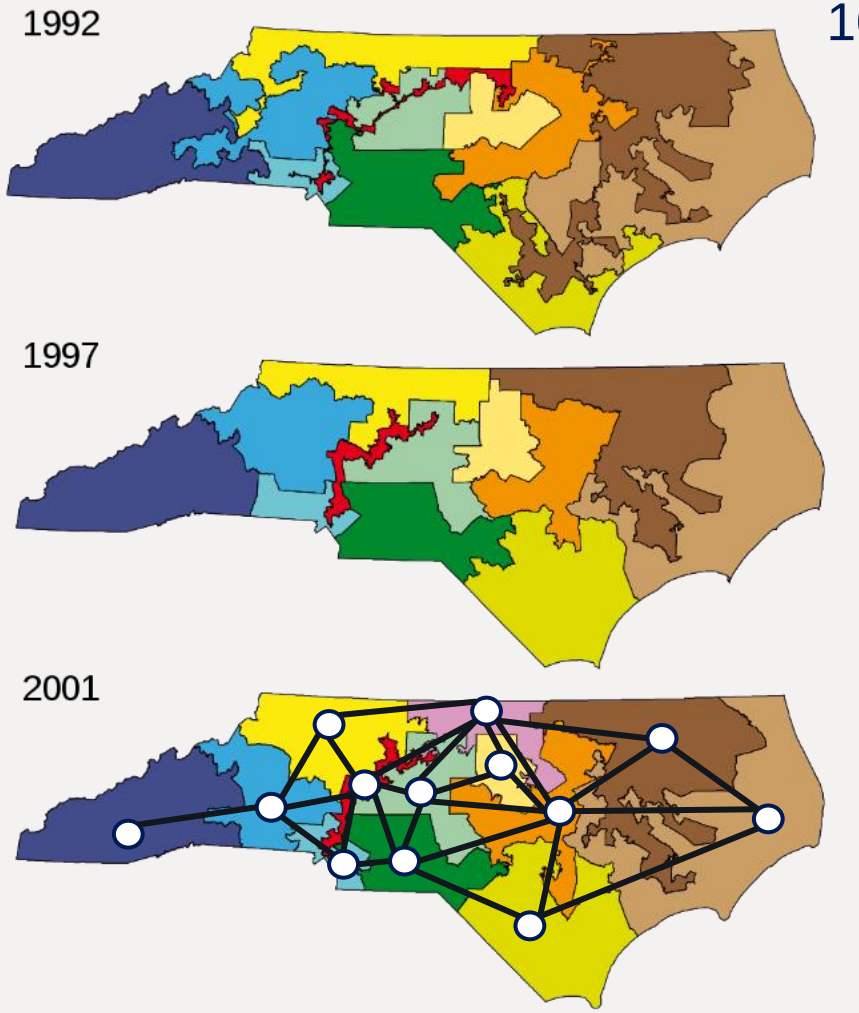
- Part 2-1. Hall's condition loosely satisfied ⇒ perfect matching exists
- **Induction Step.** Consider a graph $G = (L \cup R, E)$ with $|L| = |R| = k + 1$ such that for all $S \subseteq L$ we have $|N(S)| > |S|$.
  - Pick an arbitrary left vertex $l \in L$ and match l with an arbitrary right vertex $r \in N(\{l\})$
  - Consider the remainder graph $G' = (L - \{l\} \cup R - \{r\}, E')$ with k vertices each side.
  - Now for all $S' \subseteq L - \{l\}$, we have $|N'(S')| \geq |N(S')| - 1 \geq |S'|$.
  - In other words, Hall's condition is still satisfied (not necessarily loosely) for G'.
  - Any perfect matching of G' plus (l, r) is a perfect matching for G!

Proof of Hall's theorem (2-2 - *skeleton*)

- Part 2-2. Hall's condition strictly satisfied ⇒ perfect matching exists
- **Induction Step.** Consider a graph $G = (L \cup R, E)$ with $|L| = |R| = k + 1$ such that for all $S \subseteq L$ we have $|N(S)| \geq |S|$ and the equality holds for at least one S.
  - Find a perfect matching between $|S|$ and $|N(S)|$
  - Consider the rest of the graph without $|S|$ and $|N(S)|$
    - We can show Hall's condition still holds for this remainder graph
    - Therefore there is a perfect matching for this remainder graph; combine the two parts gives a perfect matching for the entire graph

Coloring planar graphs

- Real-world maps are usually colored to delineate boundaries of regions
  - Here the regions are electoral districts of NC
- We can abstract away the geometry by considering each region as a vertex
- Add an edge between neighboring regions
- Map coloring becomes graph coloring



Four-color theorem

- Theorem: Every planar graph is 4-colorable
- Famous as the first math theorem that's first proved via the assistance of a computer (Appel & Haken, 1976)

- The weaker theorem that states every planar graph is 5-colorable does not need the help from a computer, but is still too long (and requires many lemmas on planar graph properties) to fit into our class
- See MCS Chapter 13 (entirely devoted to planar graphs) for a thorough read

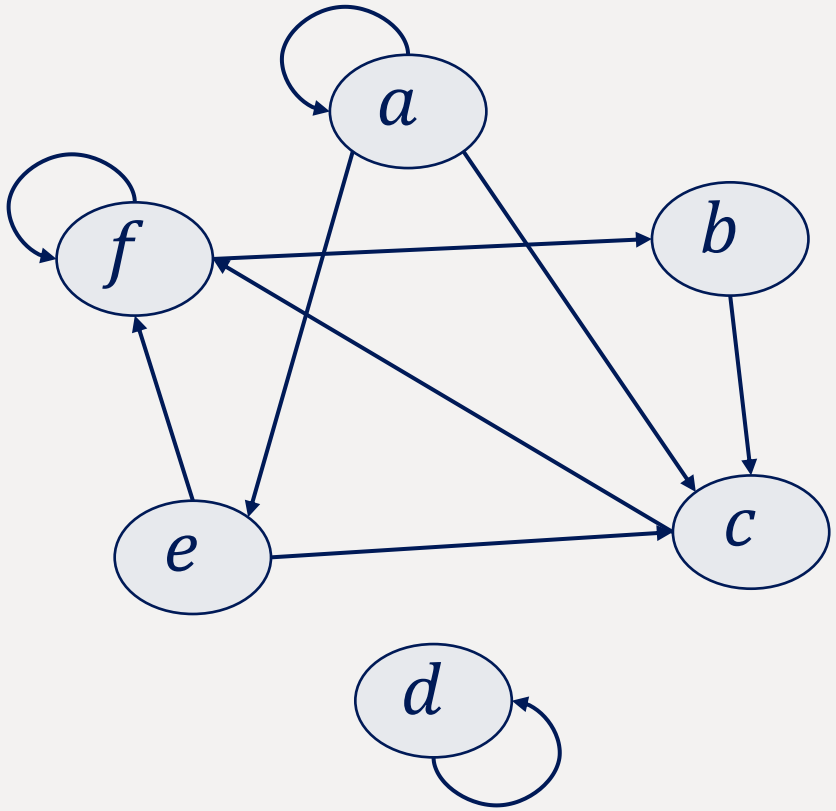
CS230 Spring 2024
Module 06: Graph Fundamentals
Directed Graphs and relations and inductions

Duke

$$R \subseteq V \times V$$

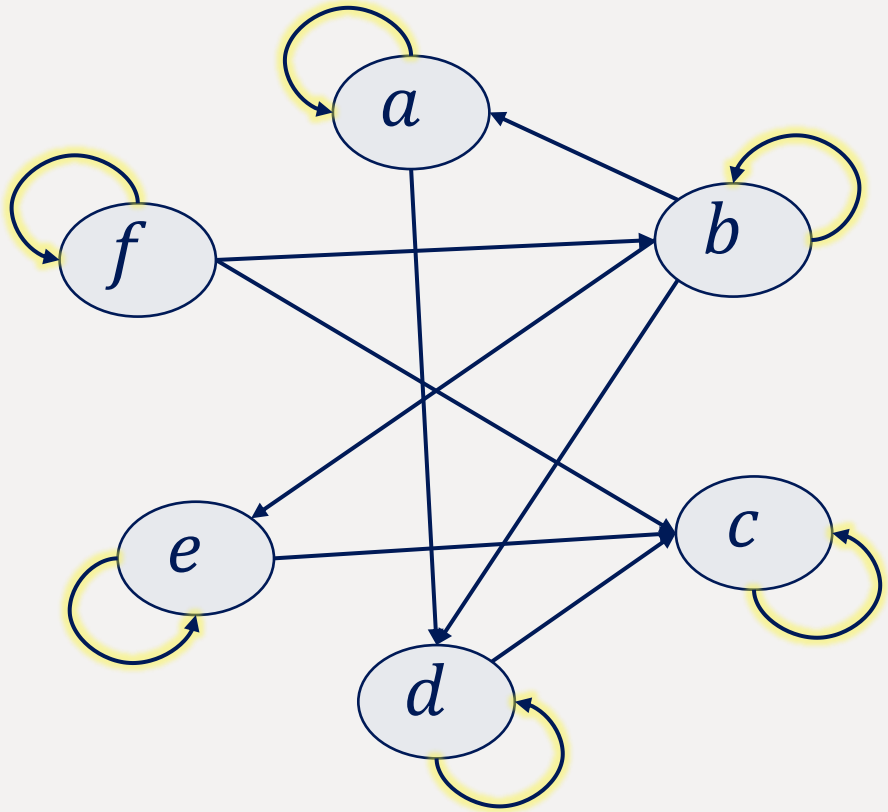
	a	b	c	d	e	f	…
a	X		X		X		
b			X				
c						X	
d				X			
e			X			X	
f		X				X	
⋮							

Edges from the “row vertex” to the “column vertex”



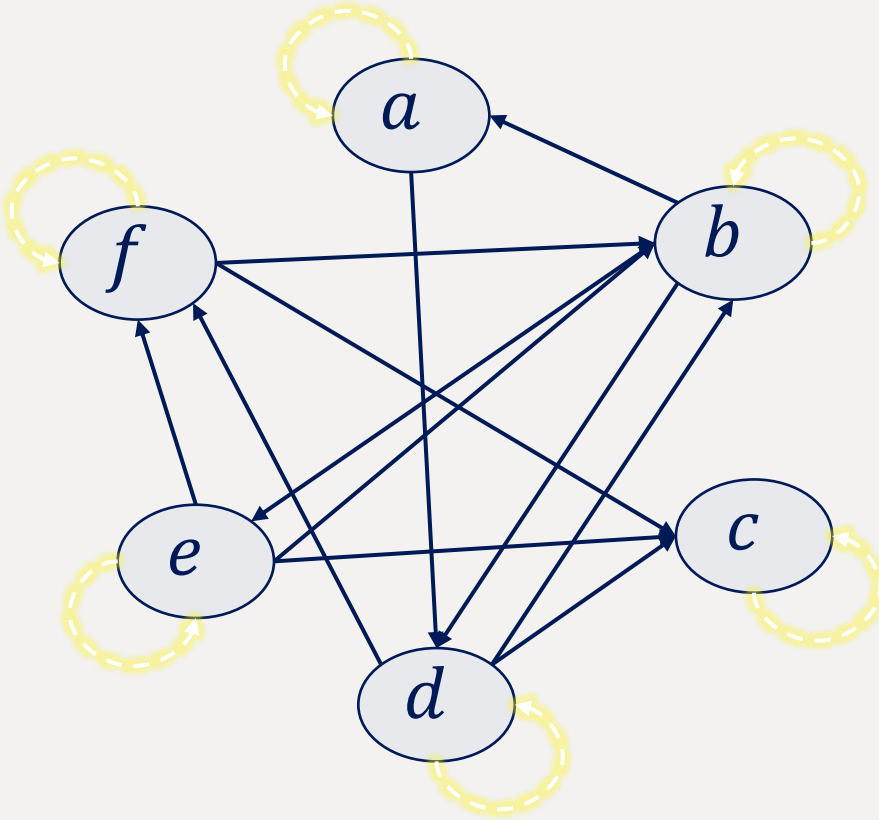
Reflexive

	a	b	c	d	e	f	…
a	X			X			
b	X	X		X	X		
c			X				
d			X	X			
e			X		X		
f		X	X			X	
⋮							…



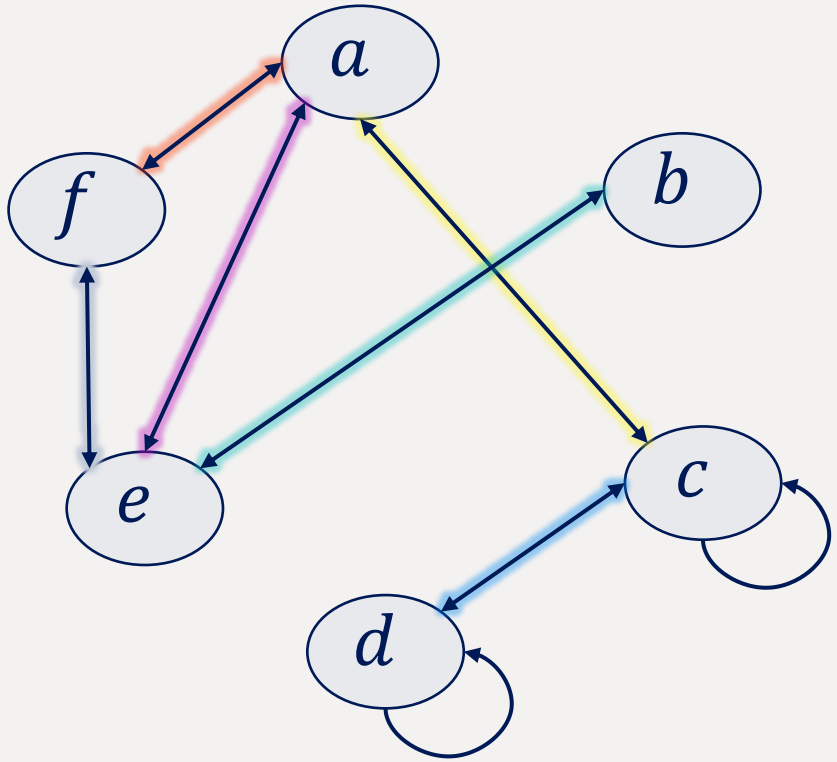
Irreflexive

	a	b	c	d	e	f	…
a				X			
b	X			X	X		
c							
d		X	X			X	
e		X	X			X	
f		X	X				
⋮							…



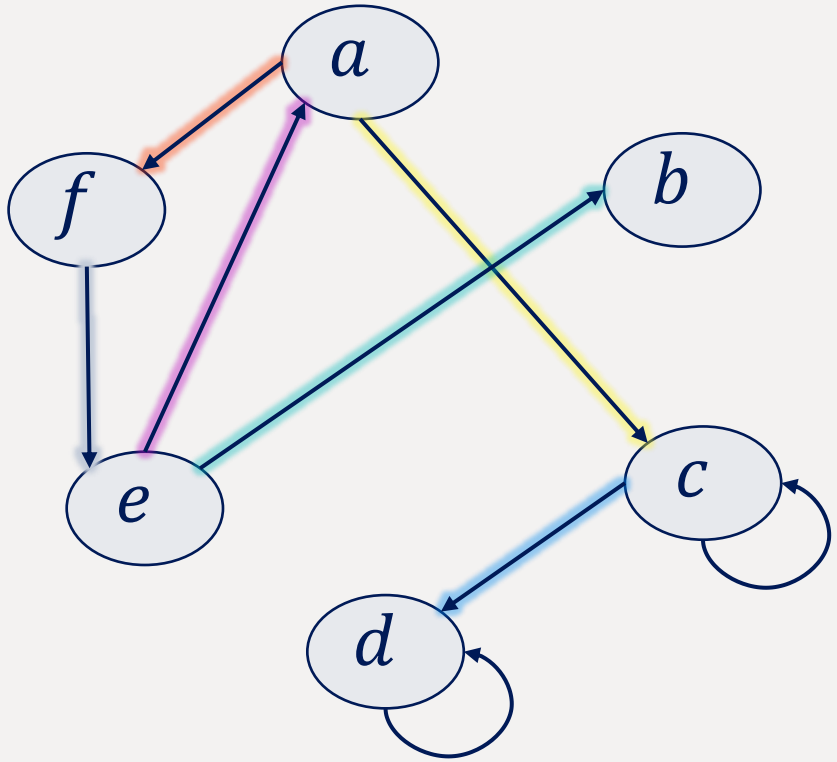
Symmetric

	a	b	c	d	e	f	…
a			X		X	X	
b					X		
c	X		X	X			
d			X	X			
e	X	X				X	
f	X				X		
⋮							…



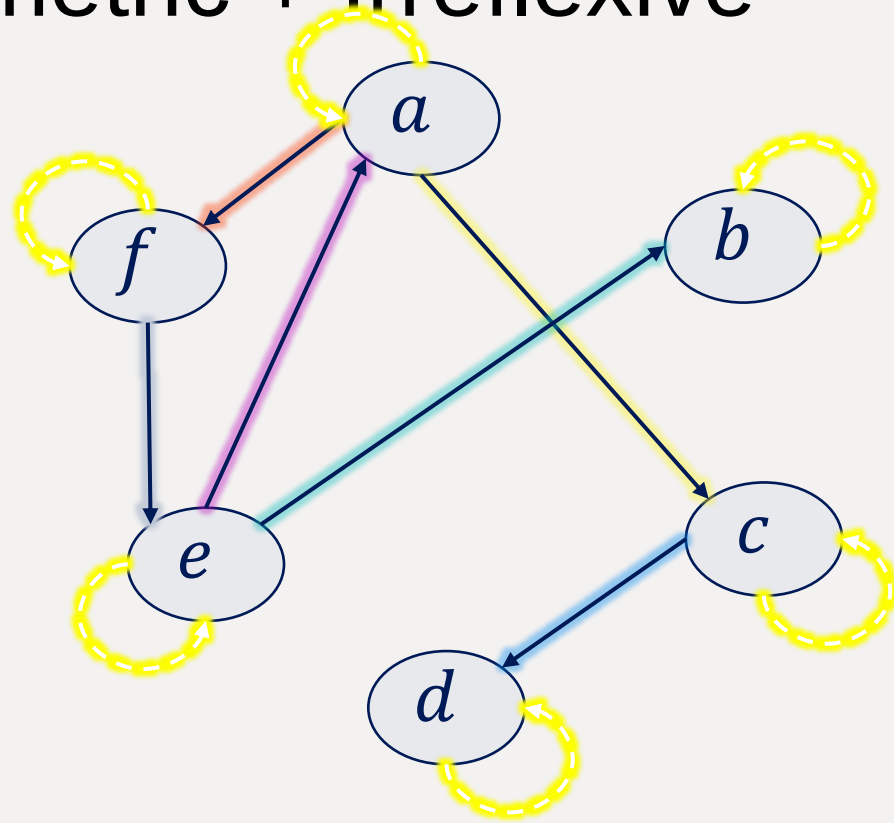
Anti-Symmetric

	a	b	c	d	e	f	…
a			X			X	
b							
c			X	X			
d				X			
e	X	X					
f					X		
⋮							…



Asymmetric = Anti-Symmetric + Irreflexive

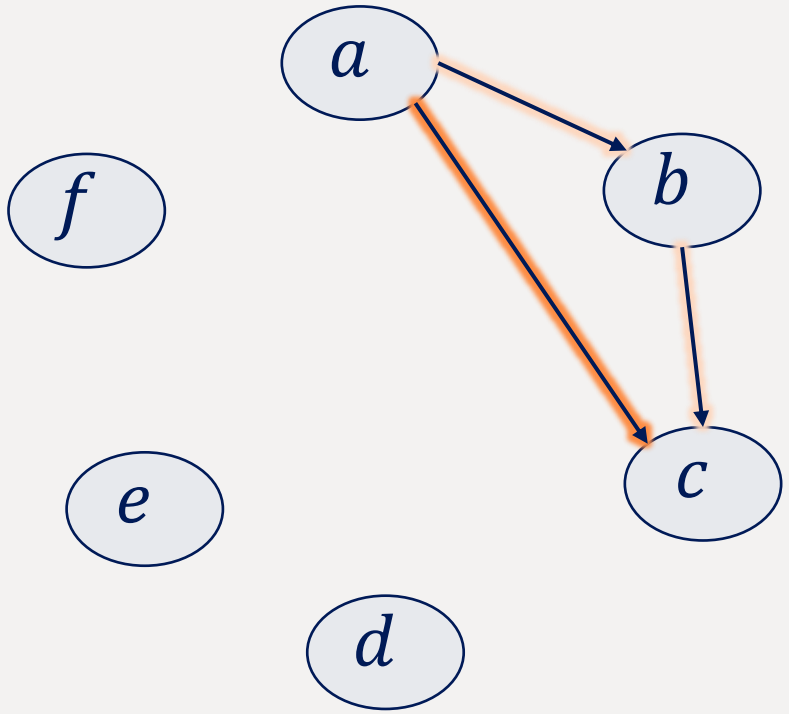
	a	b	c	d	e	f	…
a			X			X	
b							
c				X			
d							
e	X	X					
f					X		
⋮							…



Transitive

	a	b	c	d	e	f
a		X	X			
b			X			
c						
d						
e						
f						

$$(aRb) \wedge (bRc) \rightarrow (aRc)$$



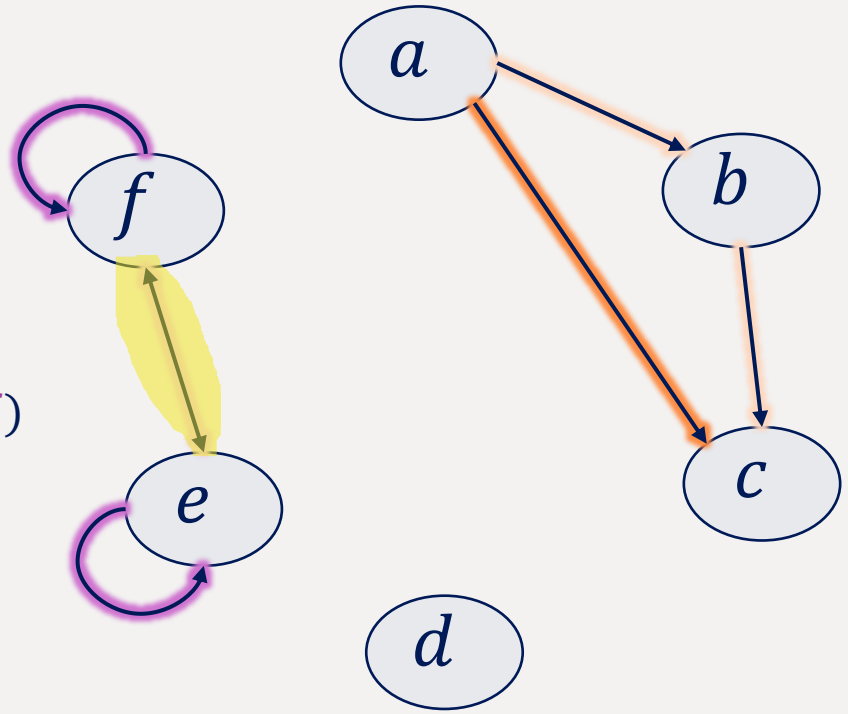
Duke

Transitive

	a	b	c	d	e	f
a		X	X			
b			X			
c						
d						
e					X	X
f					X	X

$$(aRb) \wedge (bRc) \rightarrow (aRc)$$

$$(eRf) \wedge (fRe) \rightarrow (eRe) \wedge (fRf)$$



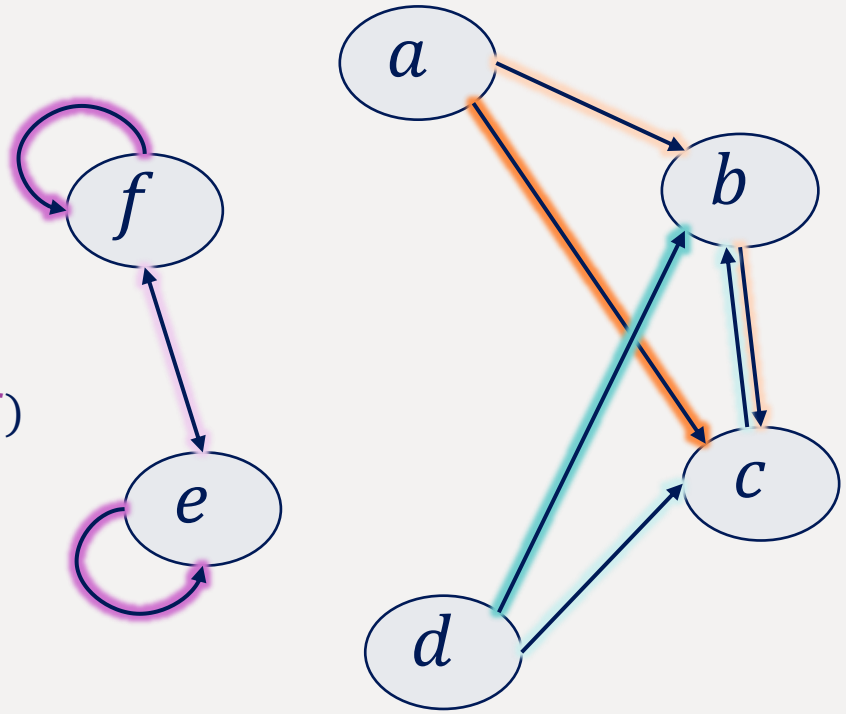
Transitive

	a	b	c	d	e	f
a		X	X			
b			X			
c		X				
d		X	X			
e					X	X
f					X	X

$$(aRb)\Lambda(bRc) \rightarrow (aRc)$$

$$(eRf)\Lambda(fRe) \rightarrow (eRe)\Lambda(fRf)$$

$$(dRc)\Lambda(cRb) \rightarrow (dRb)$$



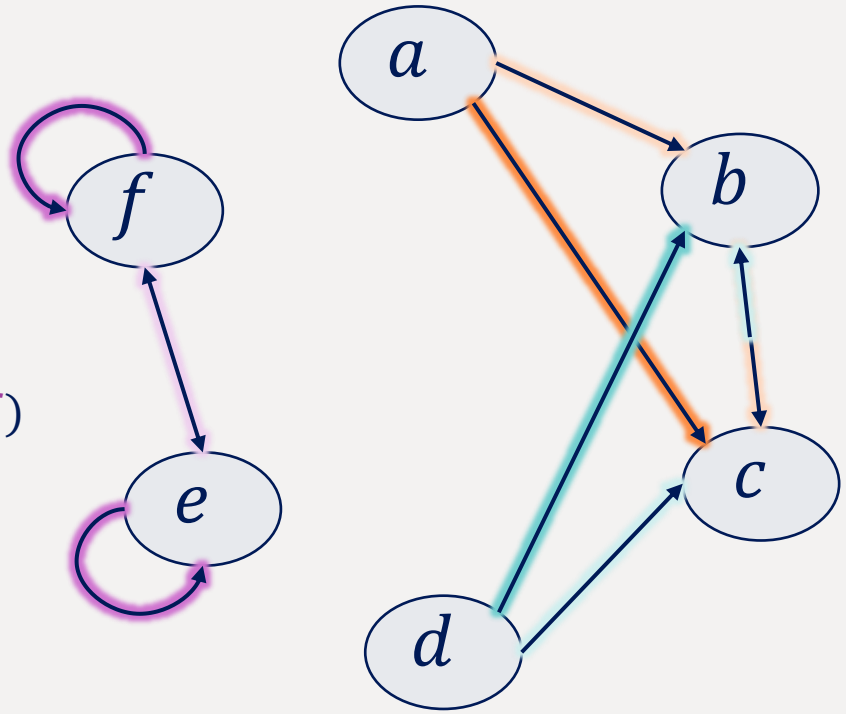
Transitive

	a	b	c	d	e	f
a		X	X			
b			X			
c		X				
d		X	X			
e					X	X
f					X	X

$$(aRb)\Lambda(bRc) \rightarrow (aRc)$$

$$(eRf)\Lambda(fRe) \rightarrow (eRe)\Lambda(fRf)$$

$$(dRc)\Lambda(cRb) \rightarrow (dRb)$$



Duke

Transitive

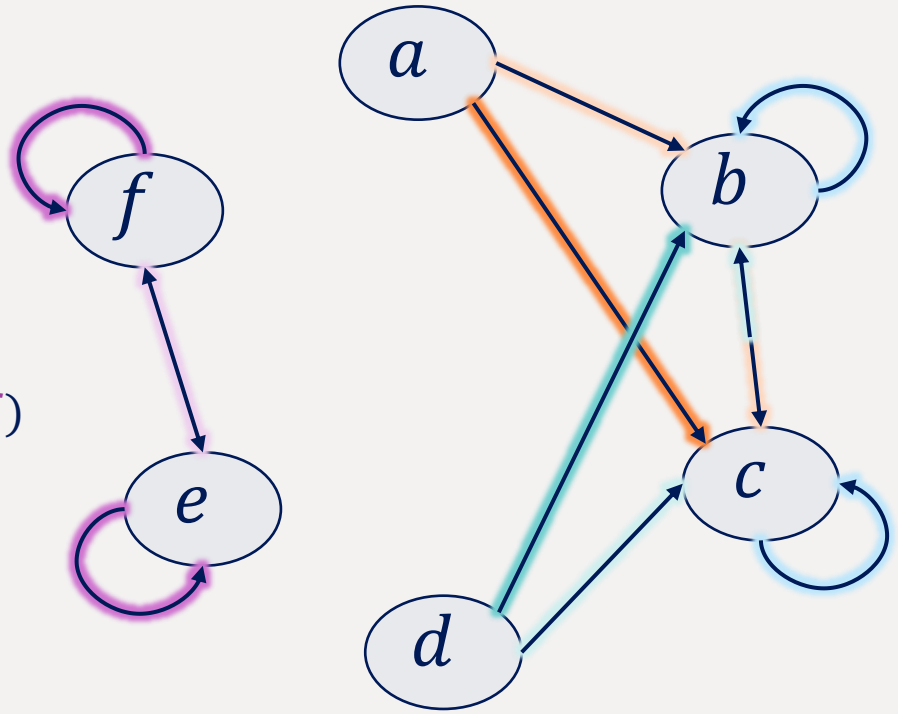
	a	b	c	d	e	f
a		X	X			
b		X	X			
c		X	X			
d		X	X			
e					X	X
f					X	X

$$(aRb)\Lambda(bRc) \rightarrow (aRc)$$

$$(eRf)\Lambda(fRe) \rightarrow (eRe)\Lambda(fRf)$$

$$(dRc)\Lambda(cRb) \rightarrow (dRb)$$

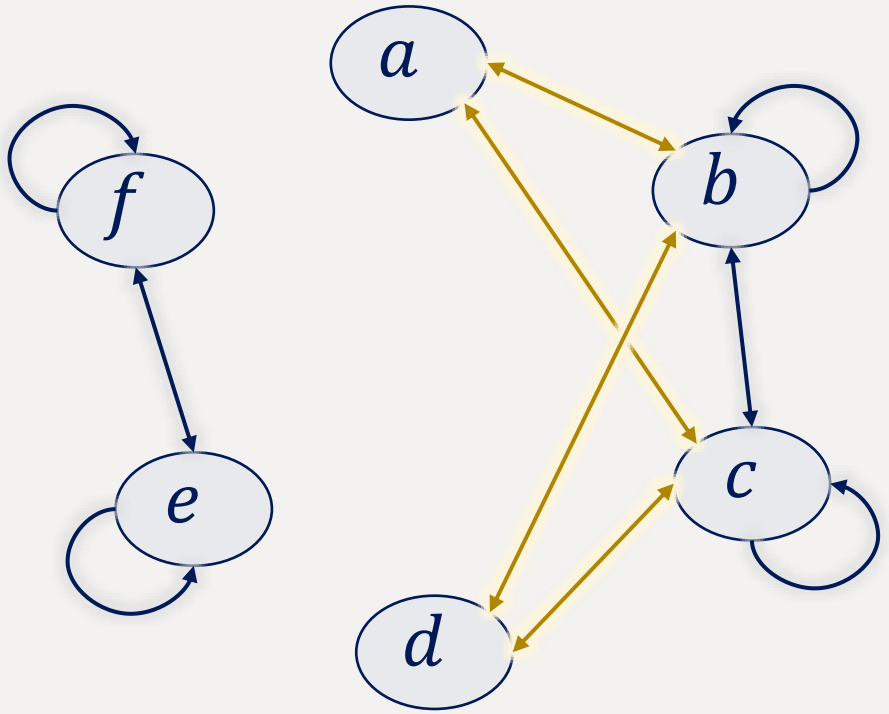
$$(bRc)\Lambda(cRb) \rightarrow (bRb)$$
$$(cRb)\Lambda(bRc) \rightarrow (cRc)$$



Duke

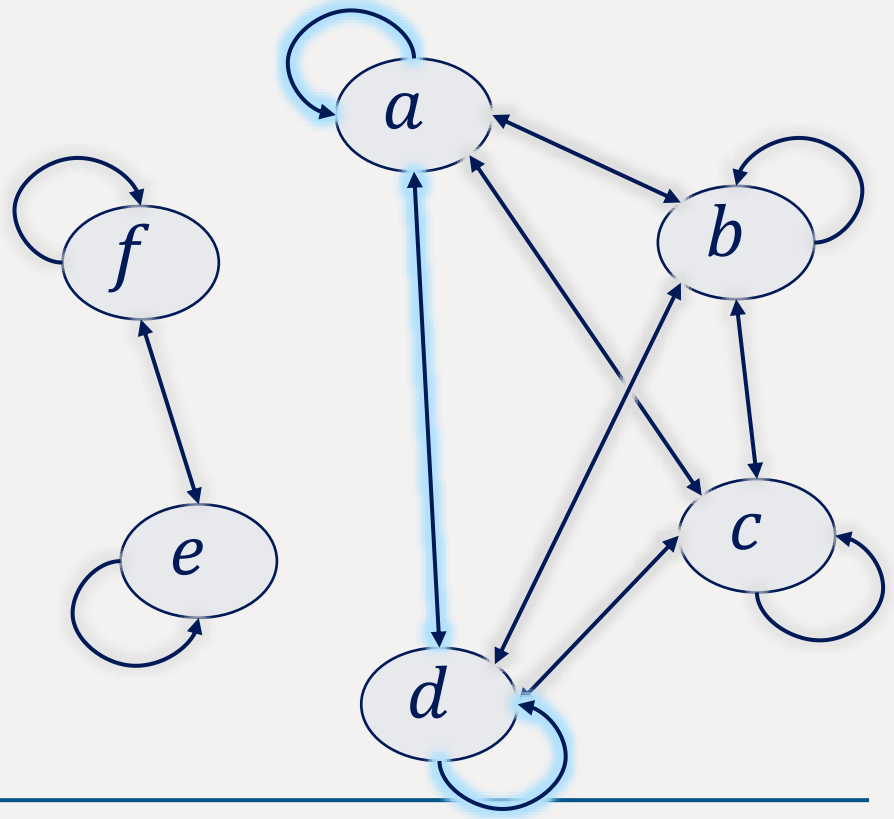
Transitive + Symmetric

	a	b	c	d	e	f
a		X	X			
b	X	X	X	X		
c	X	X	X	X		
d		X	X			
e					X	X
f					X	X



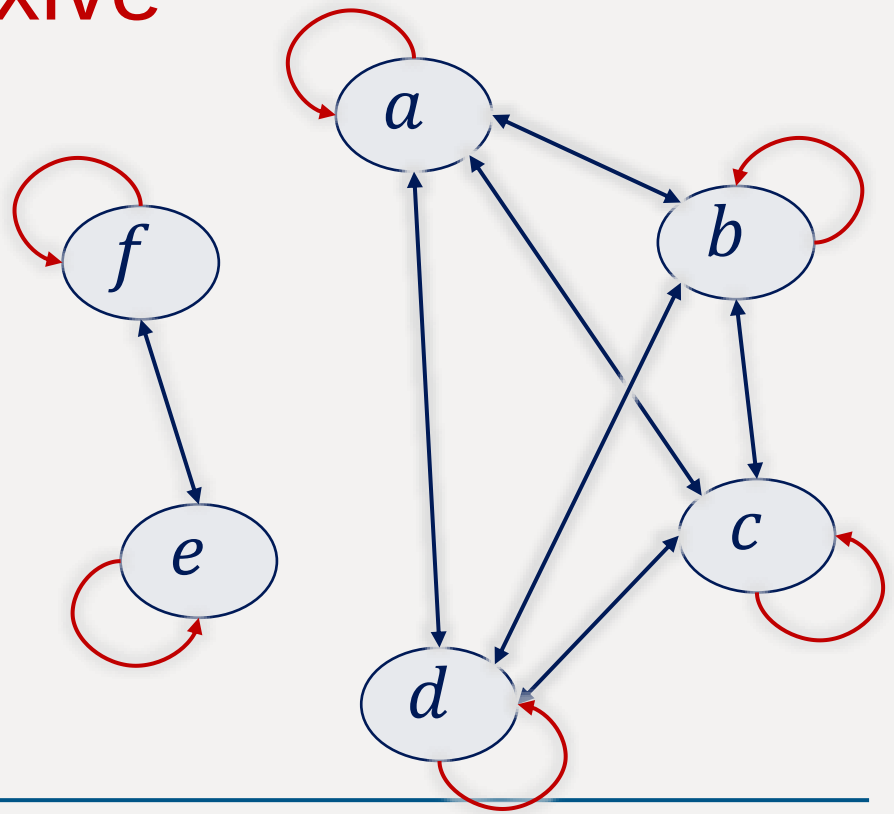
Transitive + Symmetric

	a	b	c	d	e	f
a	X	X	X	X		
b	X	X	X	X		
c	X	X	X	X		
d	X	X	X	X		
e					X	X
f					X	X



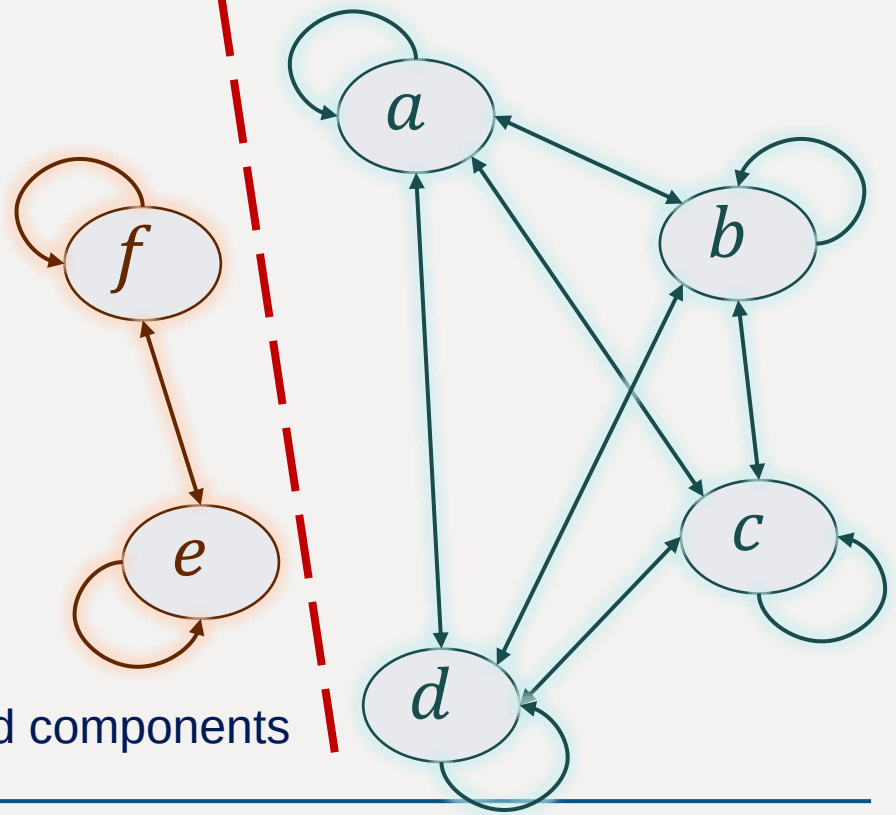
Transitive + Symmetric + Reflexive

	a	b	c	d	e	f
a	X	X	X	X		
b	X	X	X	X		
c	X	X	X	X		
d	X	X	X	X		
e					X	X
f					X	X



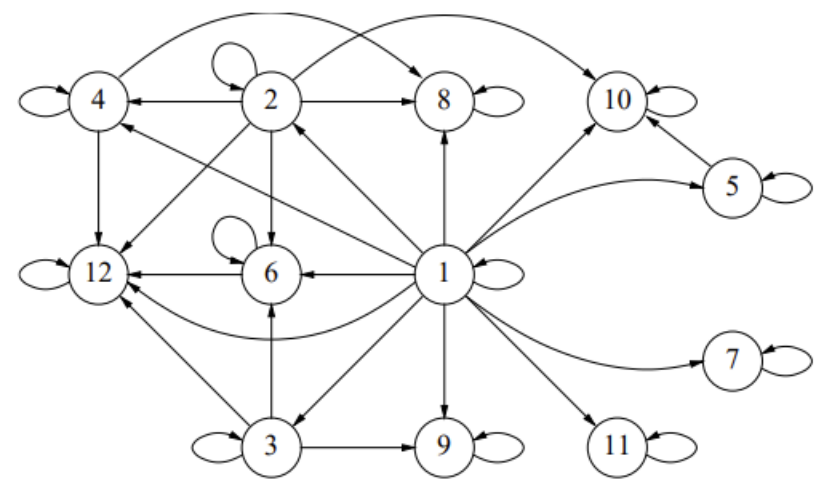
Transitive + Symmetric + Reflexive = Equivalence

	a	b	c	d	e	f
a	X	X	X	X		
b	X	X	X	X		
c	X	X	X	X		
d	X	X	X	X		
e					X	X
f					X	X

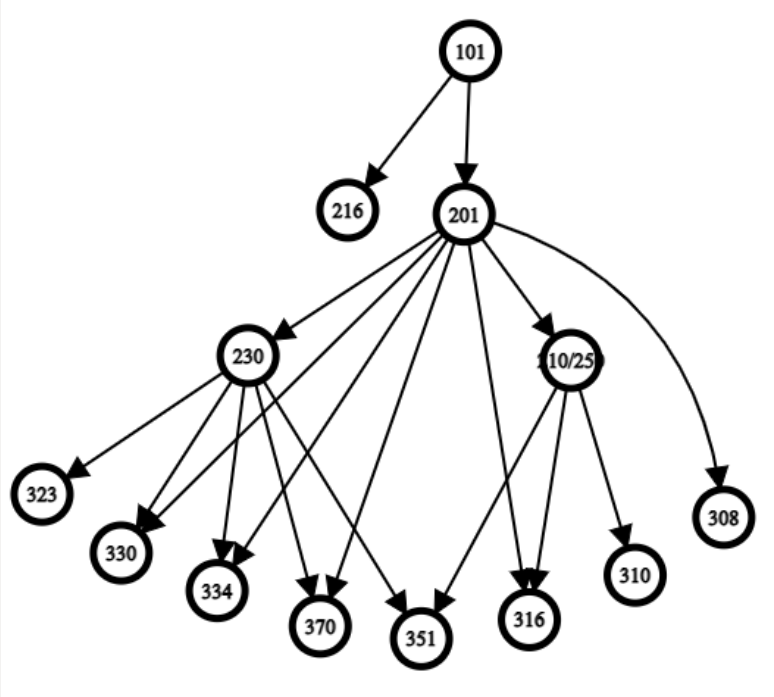


Strongly connected components

Transitive + Anti-symmetric = Partial Order



Weak partial order (reflexive)



Strong/strict partial order (irreflexive)

Directed Acyclic Graphs

Duke

PI: graphs meet relations

1 point

Categorize the following relations defined on graphs by their properties.

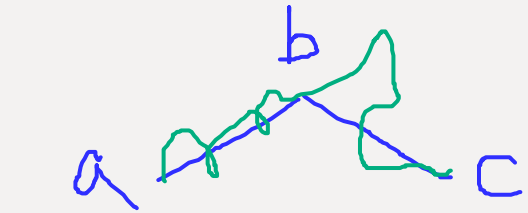
Equivalence relations

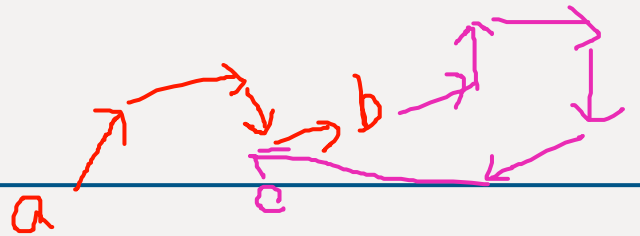
No Answers Chosen

Partial orders

Possible answers

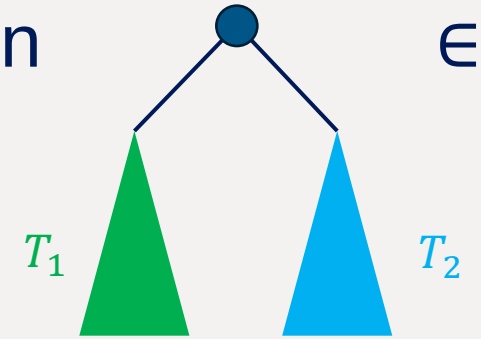
- uRv if there is a path between vertex u and vertex v in a simple undirected graph
- uRv if there is a path from vertex u to vertex v in a directed acyclic graph (DAG)



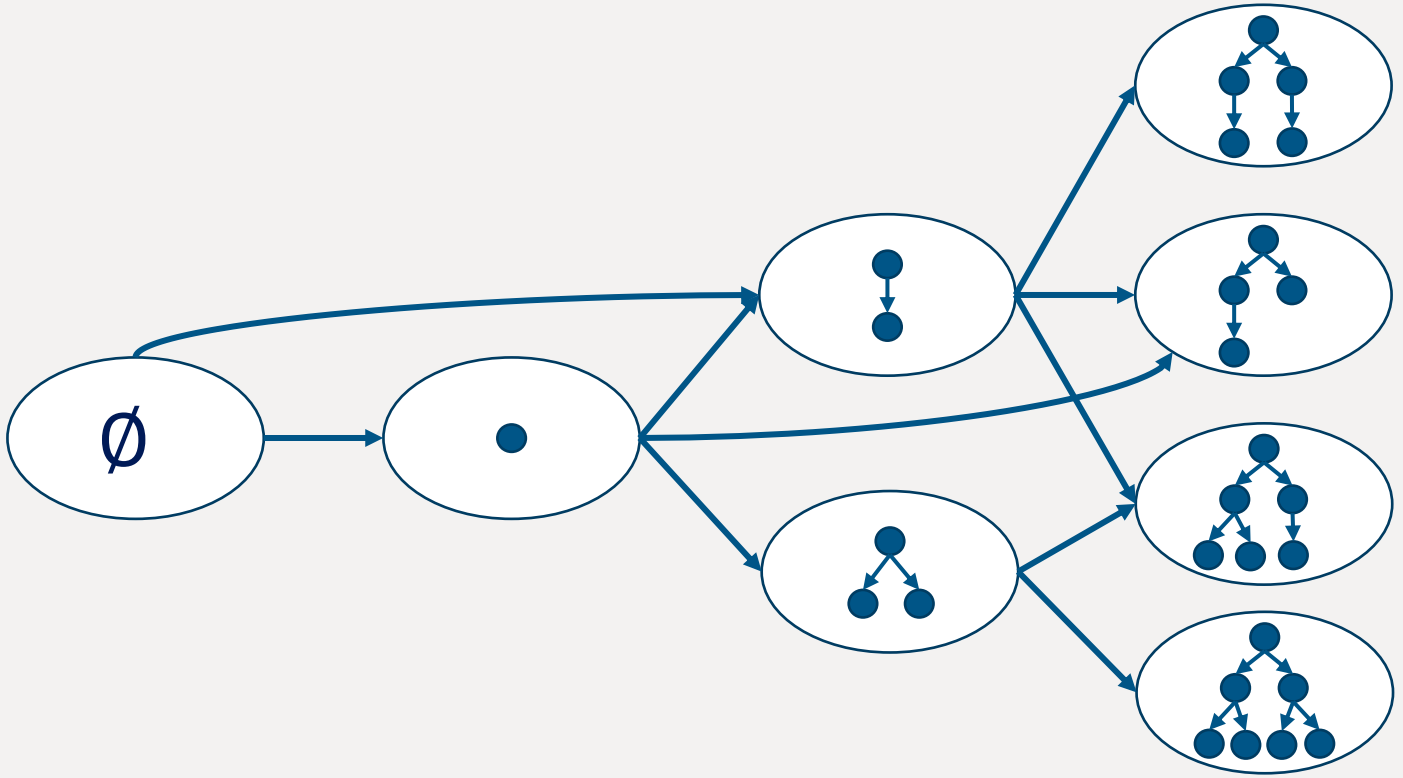


Duke

Recursively Defined Structures, Revisited

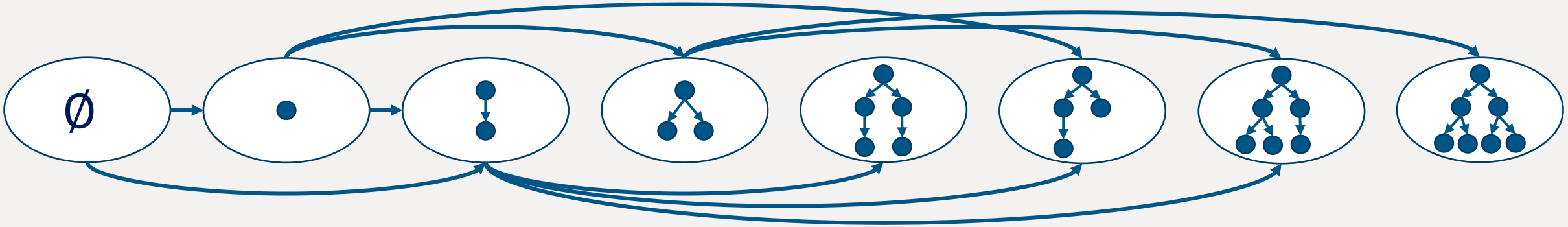
- Example 3. The set of (undirected) binary trees, $\mathcal{T}$, can be defined as:
  - **Base Case:** $T = (\emptyset, \emptyset) \in \mathcal{T}$ (the "empty tree")
  - **Constructor Case:** If $T_1, T_2 \in \mathcal{T}$, then  $\in \mathcal{T}$

Partial Order of Recursively Defined Structures



Topological Sort of Recursively Defined Structures

- How to obtain this order? We can use a modified DFS



strong induction = structural induction

- Obtain a topological sort of the recursively defined structure
- Number elements $(0), 1, 2, \ldots$ using this order
- Do strong induction on this order

- So anything achievable by structural induction is also achievable by strong induction

All inductions are the same

- weak induction = strong induction (CM5)
- weak induction can be simulated by structural induction (last week)
- structural induction can be simulated by strong induction (we just saw it)
- Therefore, weak induction = strong induction = structural induction

all inductions are the same

If ~~weak induction = strong induction~~,

- why do we teach/learn them as separate?

- Equally powerful, equally correct, NOT necessarily equally comprehensible
- Certain inductions more readable in weak/strong/structural form