VANDERBILT UNIVERSITY School of Engineering

Discrete Structures

CS 2212

(Fall 2020)

13 – Binary Relations

Chapter 5:

Relations

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)

Partial Order

Partial Order: A binary relation R, is referred to as a **partial order** if it meets the following criteria:

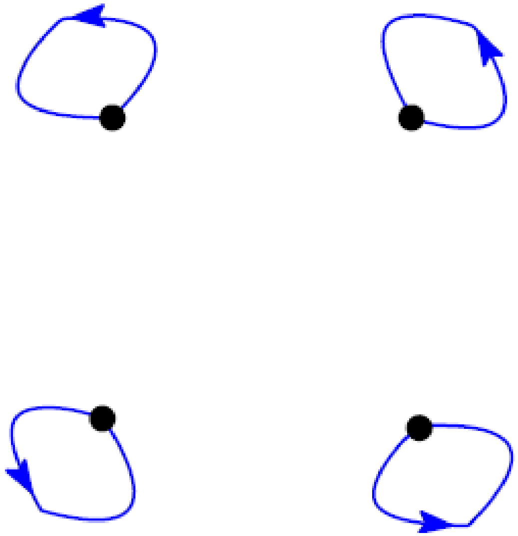
1. **R**eflexive
2. **A**ntisymmetric
3. **T**ransitive

Partial Order

Partial Order: A binary relation R, is referred to as a **partial order** if it meets the following criteria:

1. **R**eflexive
2. **A**ntisymmetric
3. **T**ransitive

Reflexive



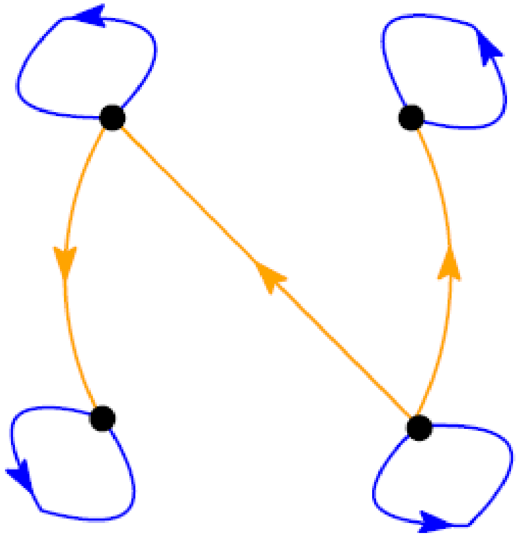
Partial Order

Partial Order: A binary relation R, is referred to as a **partial order** if it meets the following criteria:

1. **R**eflexive
2. **A**ntisymmetric
3. **T**ransitive

Reflexive

Antisymmetric



Partial Order

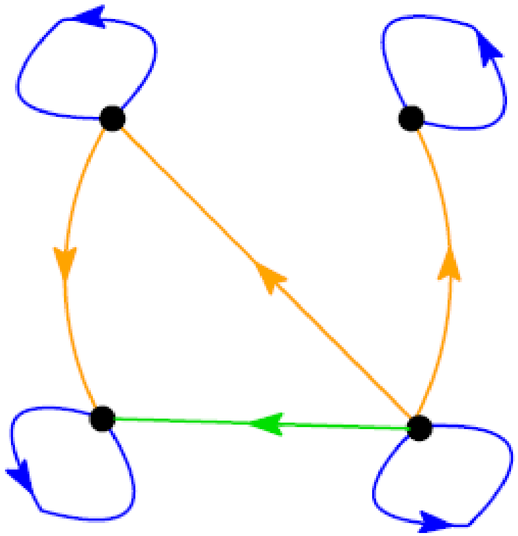
Partial Order: A binary relation R, is referred to as a **partial order** if it meets the following criteria:

1. **R**eflexive
2. **A**ntisymmetric
3. **T**ransitive

Reflexive

Antisymmetric

Transitive



Partial Order

Notation:

We use

$$\mathbf{a} \preceq \mathbf{b} \text{ to express } \mathbf{aRb}$$

noting that a partial order acts like an "ordering" operator (because "a" must come before "b").

Partial Order: A binary relation R, is referred to as a **partial order** if it meets the following criteria:

1. **R**eflexive
2. **A**ntisymmetric
3. **T**ransitive

Partially Ordered Set (POSET)

Partially Ordered Set: The **domain** along with a **partial order** defined on it is denoted **(A, ⪯)** and is called a partially ordered set or poset.

Example: The ≤ (*less than or equal to*) operator acting on the set of integers is a partial order, denoted by (**Z**, ⪯).

- The relation is **reflexive** ($x \leq x$)
- The relation is **anti-symmetric**
  (if $x \leq y$ and $y \leq x$ then $x = y$)
- The relation is also **transitive**
  ($x \preceq y$ and $y \preceq z$ imply that $x \preceq z$)

Representing POSETs

Example:

Partial order defined on a power set.

X = { 1, 2 , 3 }

P(X) = { ϕ , {1} , {2} , {3} , {1, 2} , {1, 3} , {2, 3} , {1, 2, 3} }

Domain

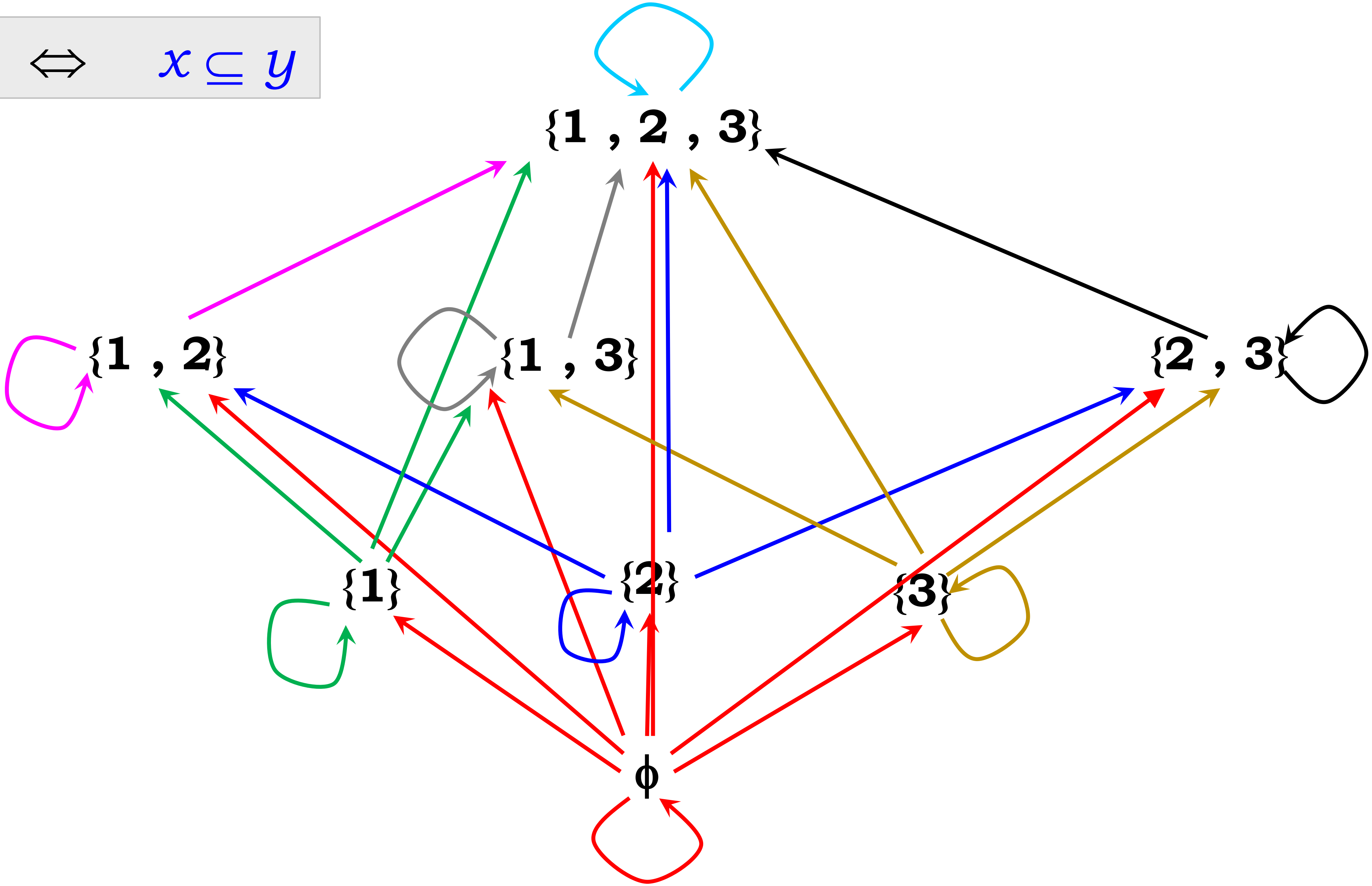
Partial order:

For all x, $y \in$ P(X),

$$x \preceq y \quad \Leftrightarrow \quad x \subseteq y$$

Representing POSETs

$$x \preceq y \quad \Leftrightarrow \quad x \subseteq y$$



Partially Ordered Sets (POSETs)

Question: Is the following relation a **Partial Order** where the domain is the set of natural numbers and

$\boldsymbol{x} \preceq \boldsymbol{y} \quad \Leftrightarrow \quad$ ***x*** **evenly divides** ***y*** ?

Answer: Yes, the above relation is a Partial Order.

1. (reflexive) ***x*** evenly divides itself.
2. (anti-symmetric) If ***x*** evenly divides ***y*** and ***y*** evenly divides ***x***, then ***x*** = ***y***.
3. (transitive) If ***x*** evenly divides ***y*** and ***y*** evenly divides ***z***, then ***x*** evenly divides ***z***.

Partially Ordered Sets (POSETs)

- Two elements of a partially ordered set, x and y, are said to be **comparable** if $x \preceq y$ or $y \preceq x$. Otherwise they are said to be **incomparable**.
- A POSET is a **total order** if **every** two elements in the domain are comparable. The partial order **$(\mathbf{Z}, \leq)$** is an example of a total order.
- An element x is a **minimal element** in the POSET if there is no $y \neq x$ such that $y \preceq x$.
- An element x is a **maximal element** in the POSET if there is no $y \neq x$ such that $x \preceq y$.

Strict Order

A **strict order** acts similar to the **<** operator on the elements of its domain.

Strict Order: A relation R is a strict order if R is

1. **Transitive**
2. **Anti-reflexive**

Why haven't we mentioned anti-symmetry condition, although we do need it here?

(Because, transitive and anti-reflexive properties imply anti-symmetry.)
Can you show how?

Strict Order

Notation: The notation **a ≺ b** is used to express aRb and is read "a is less than b".

Total Order: A strict order where every pair of elements is comparable, that is for all pair of distinct elements *x* and *y*, either $x < y$, or $y < x$.

Example: The **real numbers along with the < relation** is a strict order because

- The relation is **transitive** since if a < b and b < c, then a < c.
- The relation is **anti-reflexive** because there is no real a such that a < a.

Strict Order – Some Terminology

If $$x < y$$

We say that in terms of the order,

x **precedes** *y*, or

x is a **predecessor** of *y*, or

y is a **successor** of *x*.

Strict Order – Some Terminology

If $\{z \mid x < z < y\} = \varnothing$,

then, we say that

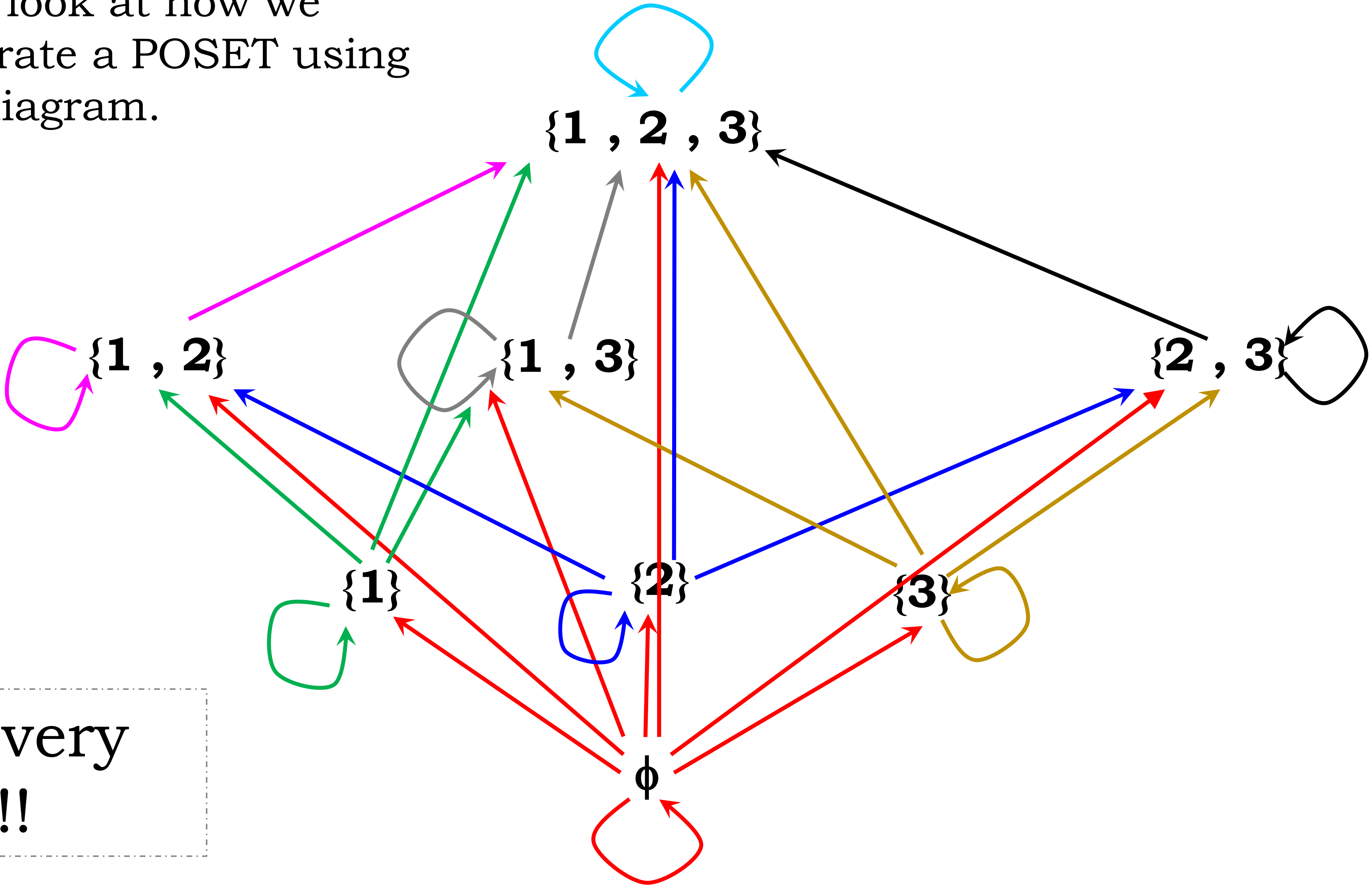
x is an **immediate predecessor** of y, or

y is an **immediate successor** of x

($\{z \mid x < z < y\} = \varnothing$, basically means that **nothing comes between x and y**).

Representing POSETs

Let's take a look at how we might illustrate a POSET using our graph diagram.



Can be very messy !!!

POSETs and Hasse Diagrams

A **Hasse diagram** is a graph representation of a POSET but is easier to read because

- it *only lists immediate predecessor edges*.
- edges are usually *oriented up* from x to y when $\boldsymbol{x} < \boldsymbol{y}$, so edge direction can be removed.

POSETs and Hasse Diagrams

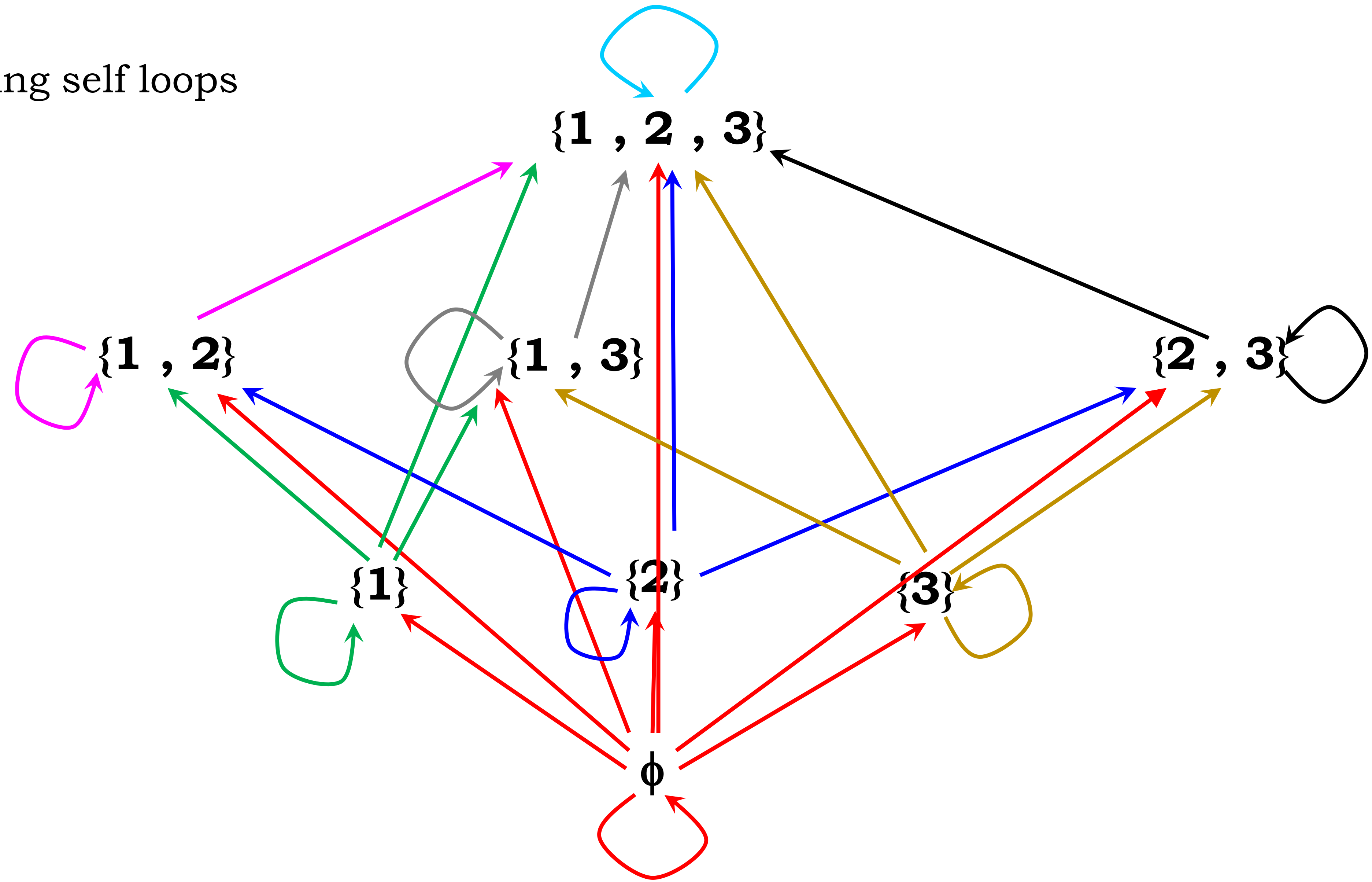
A **Hasse diagram** is a POSET diagram that…

1. Removes all reflexive edges (self-loops).
2. Removes all transitive edges.
3. Removes directions on edges (they are implied).

(If $x < y$, then x is drawn below y, and we understand that edges point upwards in the original graph)

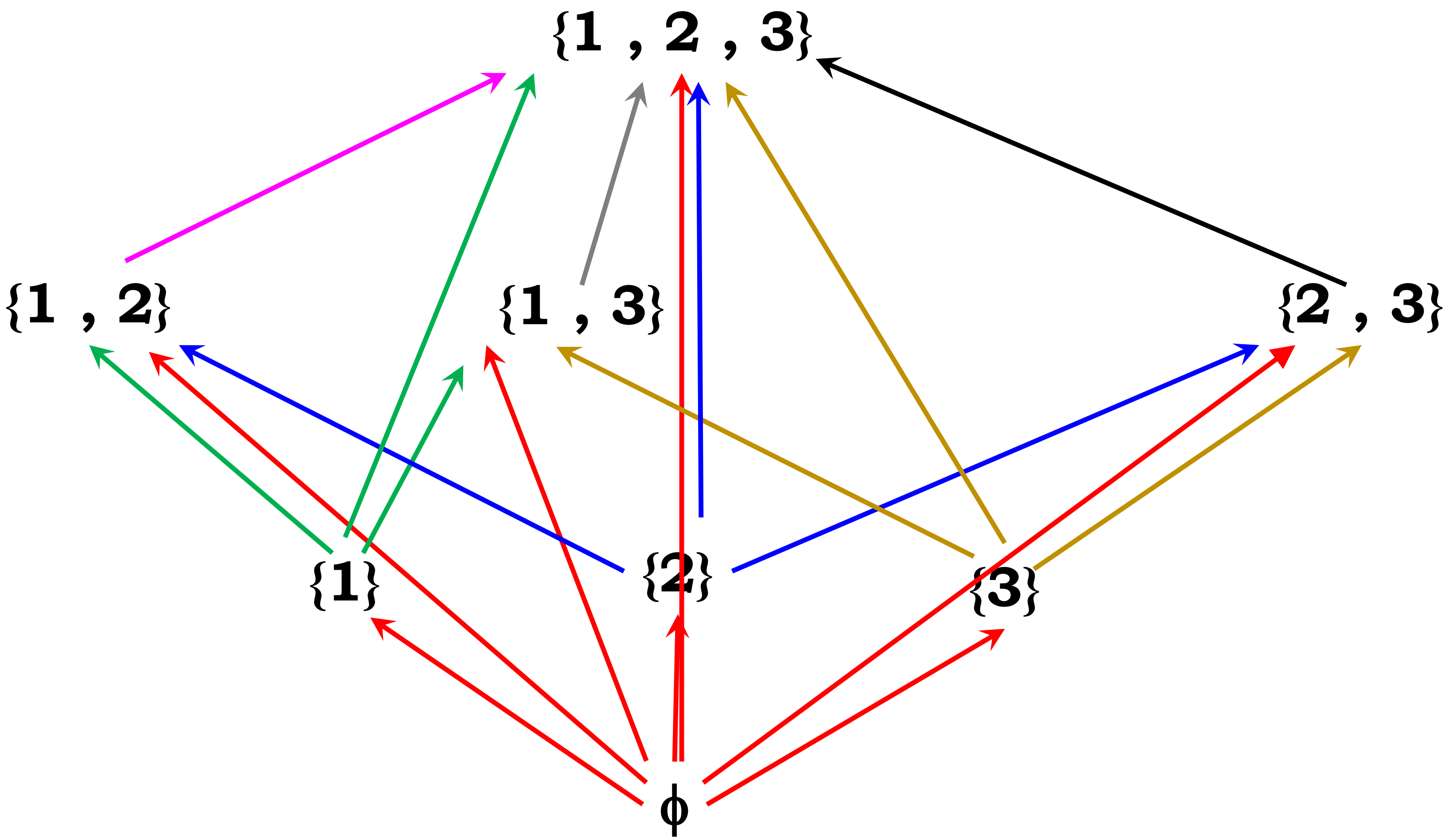
Representing POSETs

- Removing self loops



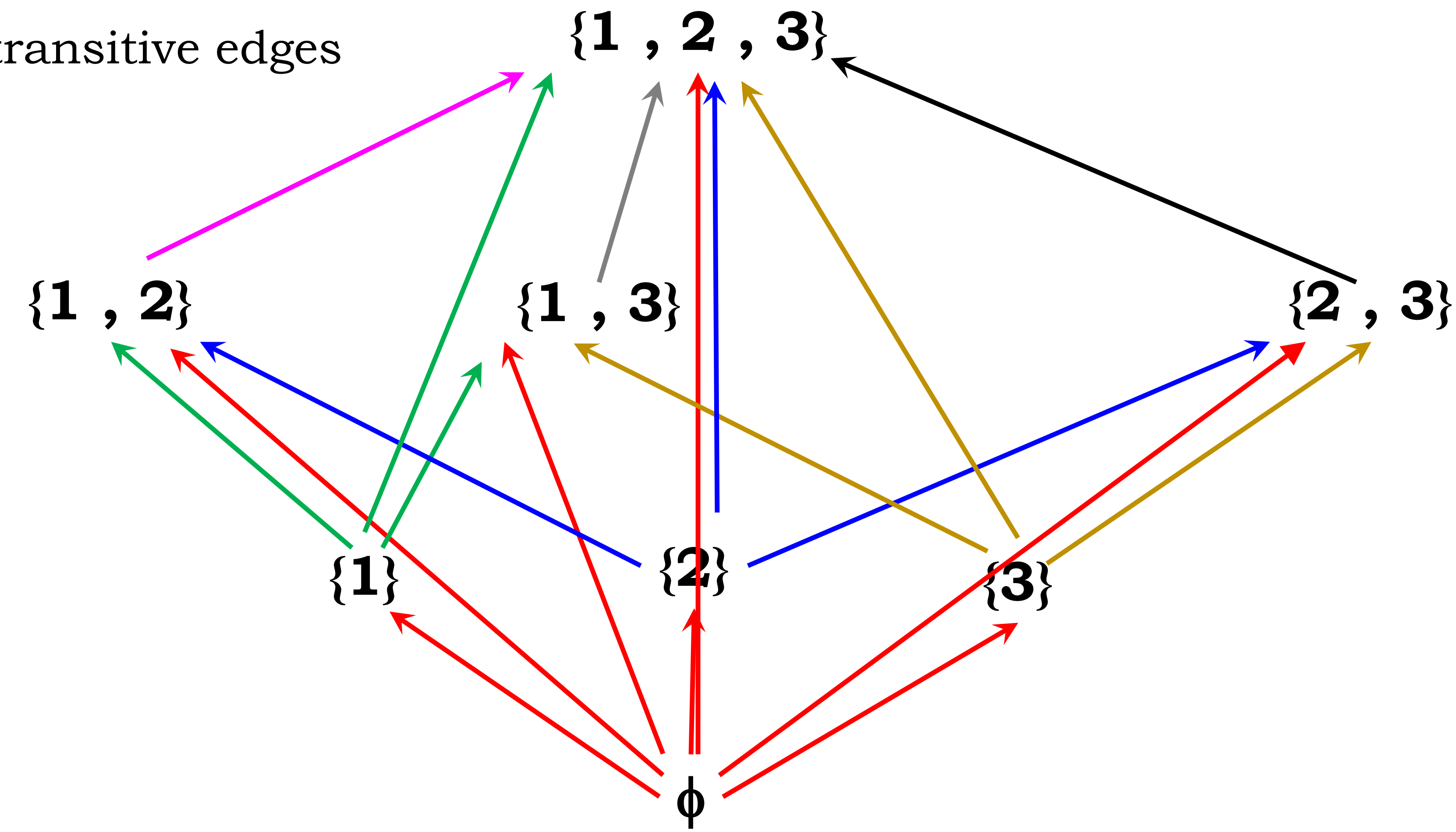
Representing POSETs

- Removing self loops



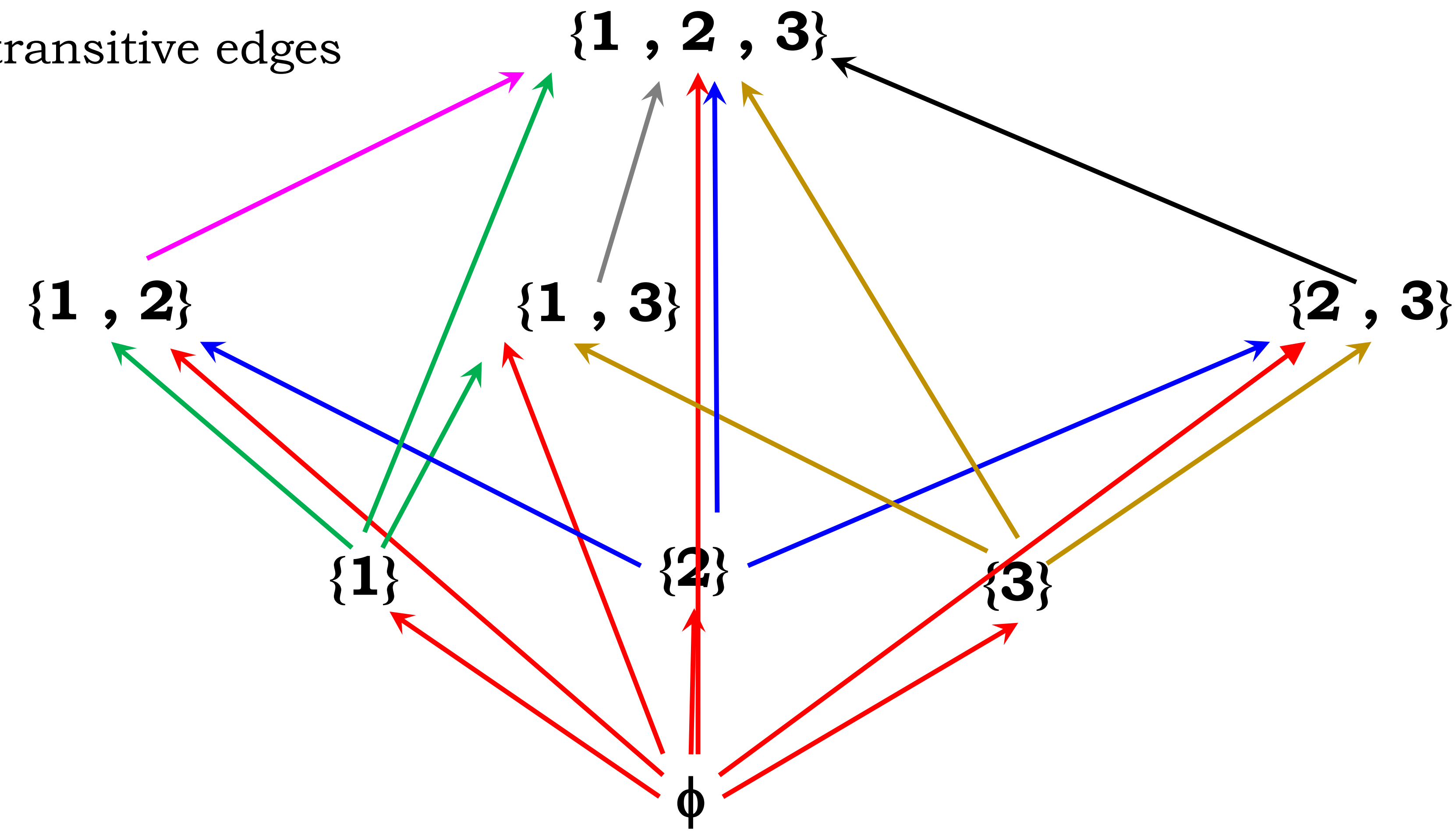
Representing POSETs

- Removing self loops
- Removing transitive edges



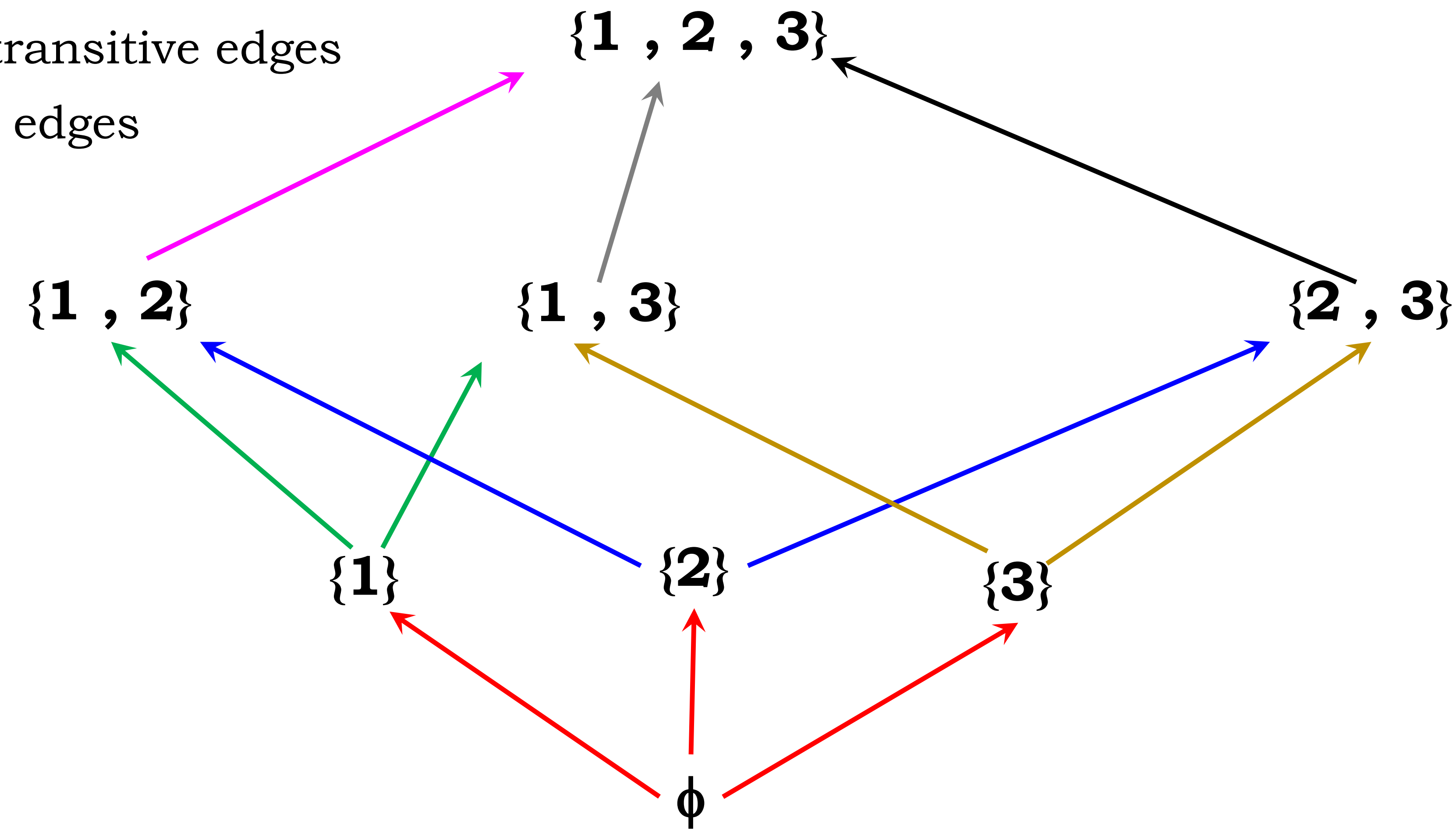
Representing POSETs

- Removing self loops
- Removing transitive edges



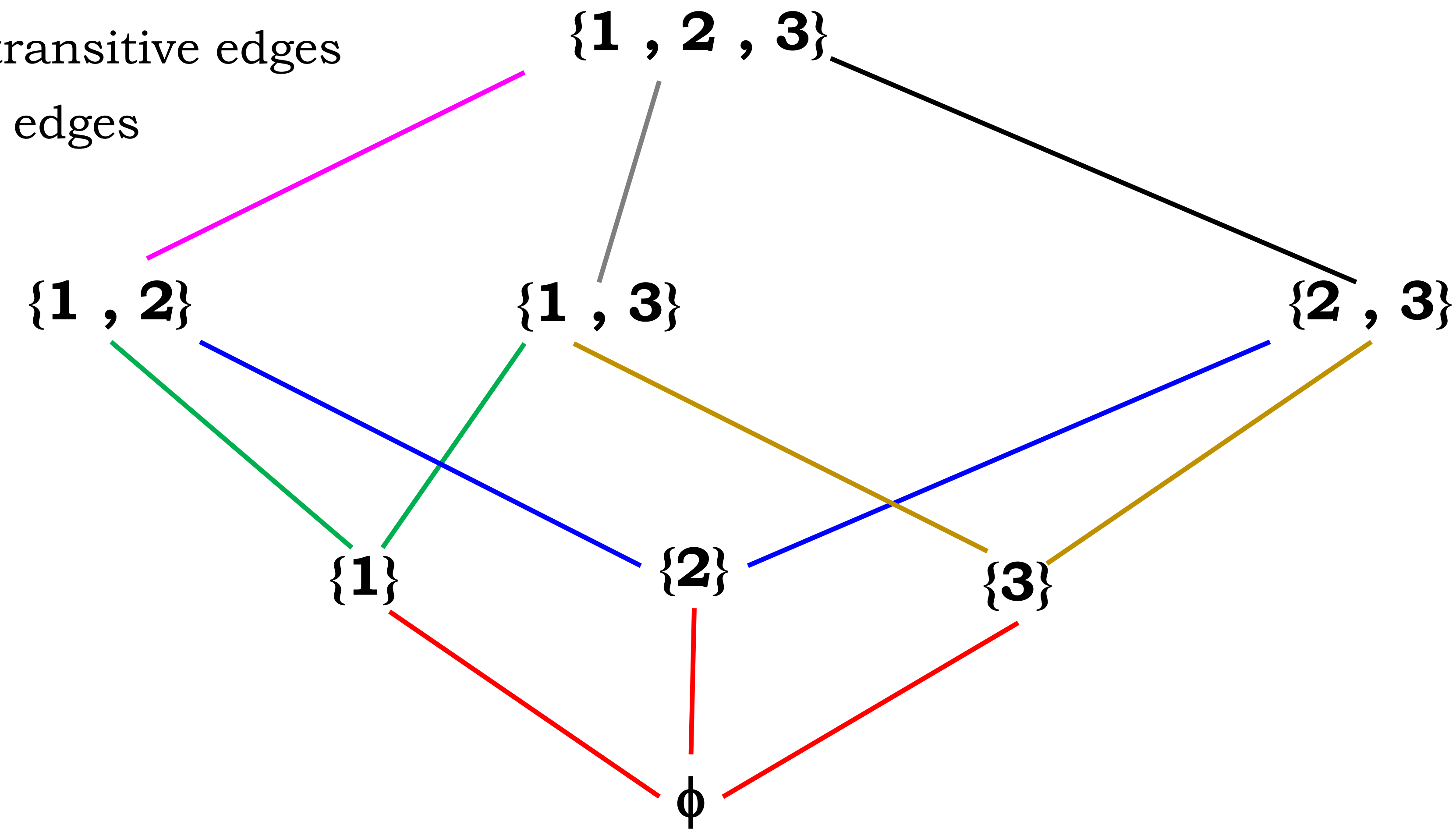
Representing POSETs

- Removing self loops
- Removing transitive edges
- Undirected edges



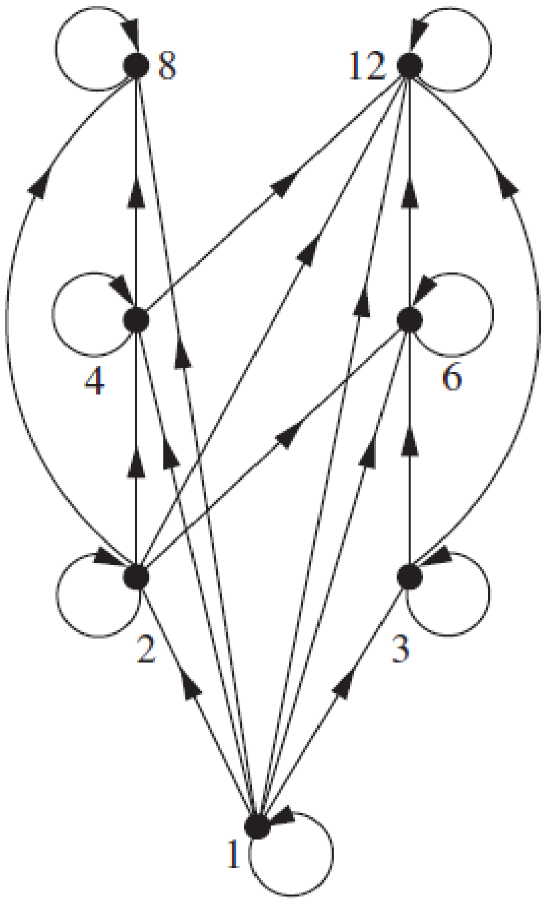
Representing POSETs

- Removing self loops
- Removing transitive edges
- Undirected edges



Hasse Diagram

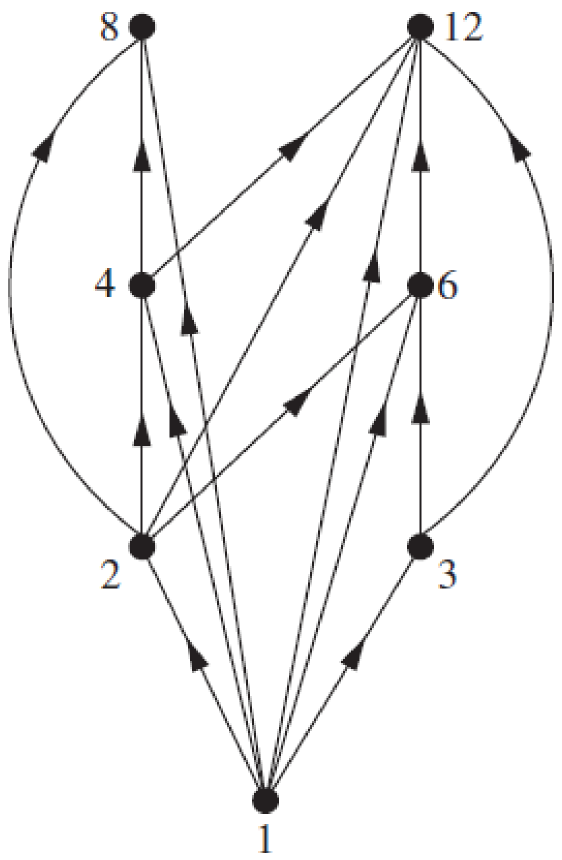
Draw the Hasse diagram representing the partial ordering {(a,b) | a divides b} on {1, 2, 3, 4, 6, 8, 12}.



Hasse Diagram

Draw the Hasse diagram representing the partial ordering {(a,b) | a divides b} on {1, 2, 3, 4, 6, 8, 12}.

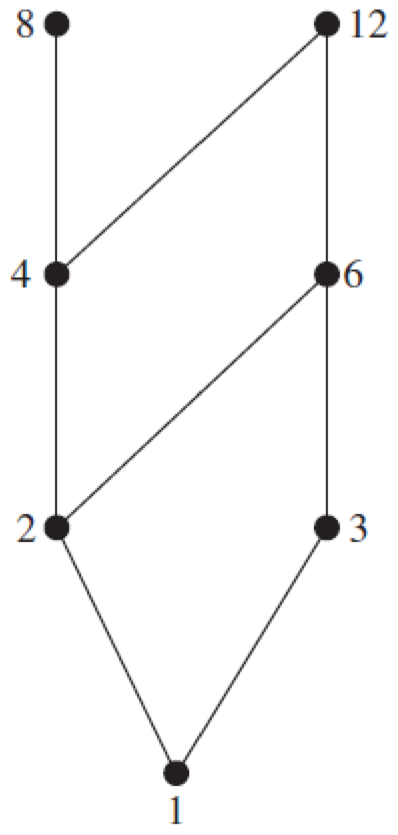
- Removing self loops



Hasse Diagram

Draw the Hasse diagram representing the partial ordering {(a,b) | a divides b} on {1, 2, 3, 4, 6, 8, 12}.

- Removing self loops
- Removing edges implied by transitivity
- Arrange all edges to point upward and then make them undirected,



Hasse Diagram – Maximal and Minimal

Let S be a subset of a POSET called P.

$x \in$ S is **minimal** element of S if there is no $y \neq x$ s.t. $y \preceq x$

(x has no predecessors).

$x \in$ S is a **maximal** element of S if there is no $y \neq x$ s.t. $x \preceq y$

(x has no successors).

There can be **multiple** maximal or minimal elements in S.

Reminder: Some people use $\leq$ for $\preceq$ when discussing relations.

Hasse Diagram – Maximal and Minimal

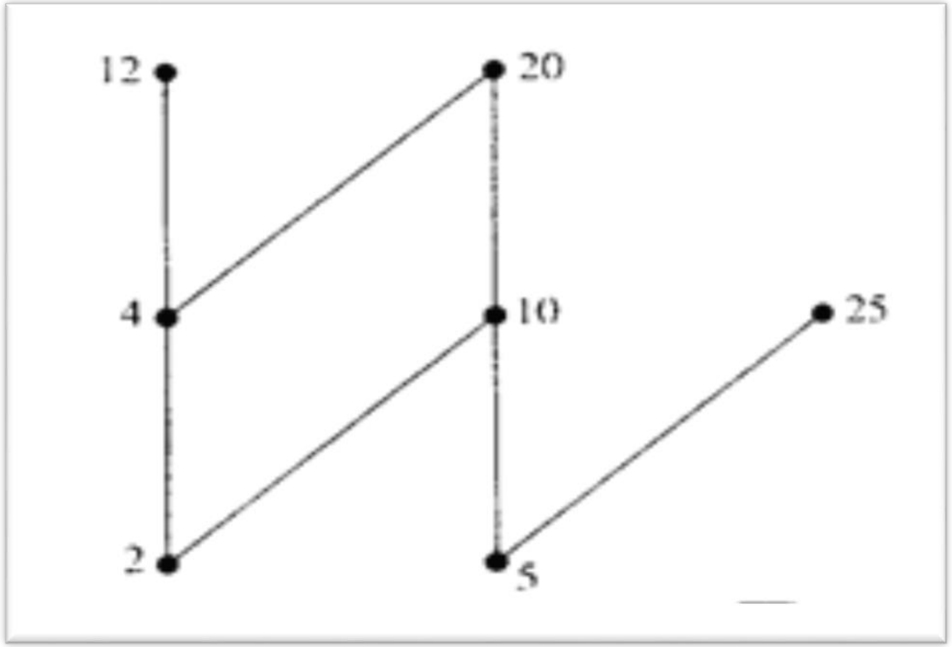
Example: Identify the **minimal** and **maximal** element(s).

Minimal elements:

2, 5

Maximal elements:

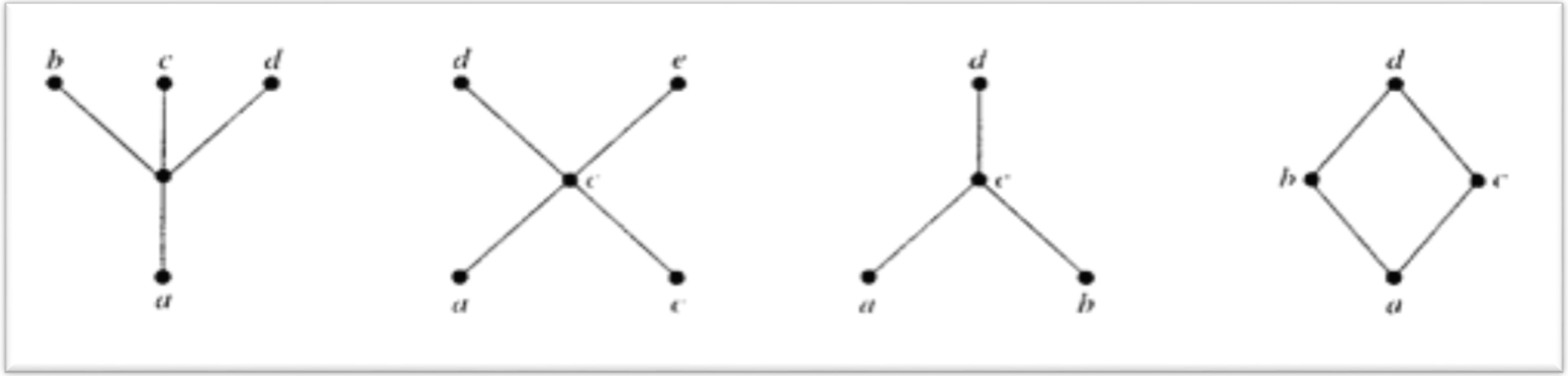
12, 20



Hasse Diagram – Greatest and Least

A maximal element $x \in S$ is the **greatest element** of S if ***x > y*** (a successor) **for all** $y \in$ S.

A minimal element $x \in$ S is the **least element** of S if ***x < y*** (a predecessor) **for all** $y \in$ S.



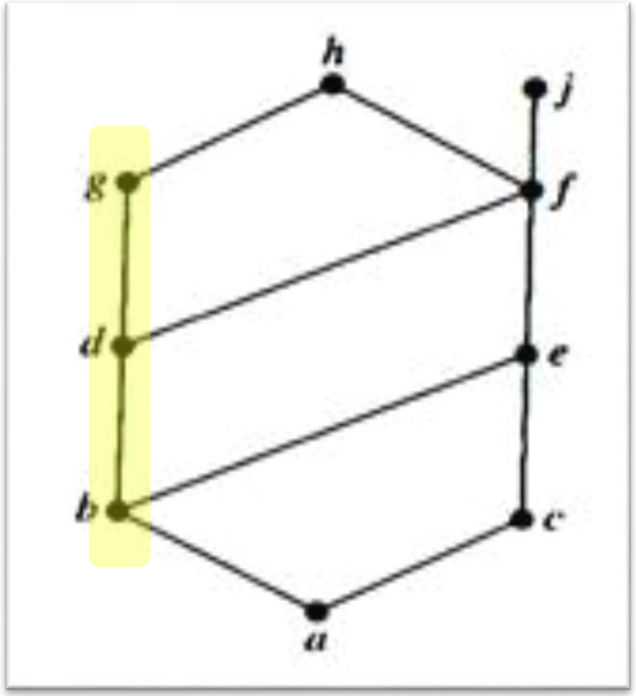
Greatest	None	None	d	d
Least	a	None	None	a

Hasse Diagram – Upper Bounds

Let $(S, \preceq)$ be a POSET and let A be a subset of S.

Upper bound of A: $u \in S$ s.t. $a \preceq u$ for all $a \in A$.

Least upper bound of A: An upper bound of A that is a *predecessor of all upper bounds* of A. Denoted as lub(A).



Upper bounds of { b, d, g }: ***g* , *h***

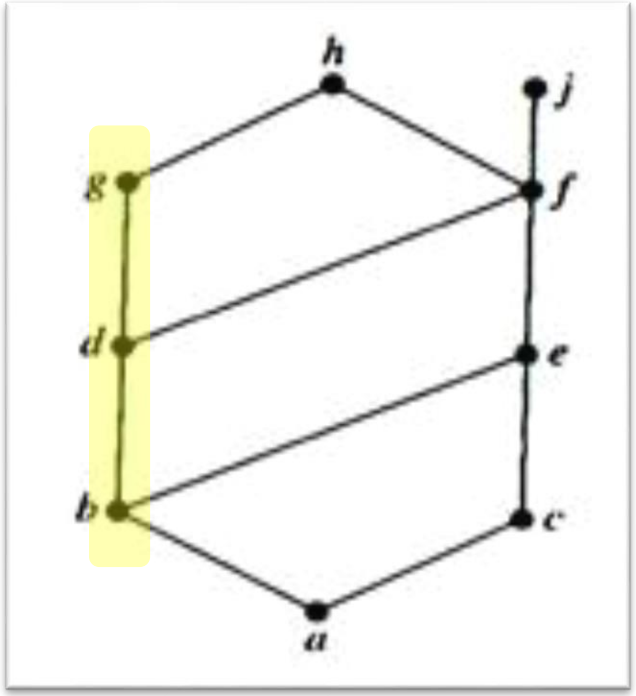
Lub of { b, d, g }: ***g*** (Why? Because $g < h$, i.e., g is a predecessor of h.)

Hasse Diagram – Lower Bounds

Let (S, ≤) be a POSET and let A be a subset of S.

Lower bound of A: $l \in S$ s.t. $l \leq a$ for all $a \in A$.

Greatest lower bound of A: A lower bound of A that is a *successor of all lower bounds* of A. Denoted as glb(A).



Lower bounds of { *b, d, g* }: ***a* , *b***

Glb of { *b, d, g* }: ***b*** (Why? Because $a < b$, i.e., *b* is a successor of *a*.)

Equivalence Relations (RST)

Equivalence: A relation R, is considered an **equivalence relation** if it is

1. **R**eflexive
2. **S**ymmetric
3. **T**ransitive

An equivalence relation can be thought of as a way to group elements together such that any two elements in the same group are **"equivalent".**

Equivalence Relations (RST)

To denote equivalence relation, the following notation is used:

a ~ b (which is read as "a is equivalent to b")

Examples:

The relation R over a set of students taking CS2212 such that

$x \sim y \quad \leftrightarrow \quad x$ and y have the same section.

The relation R over a set of people such that

$x \sim y \quad \leftrightarrow \quad x$ and y have the same birthday.

Equivalence Relations (RST)

Is the following an equivalence relation (RST) over the set of integers **Z**?

$x \mathbf{R} y$ if and only if $x \leq y$ or $x > y$.

Yes

this relationship is

- reflexive,
- Symmetric, and
- transitive.

Equivalence Relations (RST)

Is the following an equivalence relation (RST) over the set of integers **Z**?

x **R** y if and only if $|x - y| \le 2$.

No

- This relation is not transitive.
- Consider that (3, 5) and (5, 7) are in R but (3, 7) is not.

Equivalence Classes

If R is an equivalence relation over A, then for each a ∈ A the **equivalence class** of a, denoted by **[a]**, is the set

$$[a] = \{ x \mid x \, R \, a \}.$$

Consider students taking CS 2212, and

$$x \sim y \quad \leftrightarrow \quad x \text{ and } y \text{ have the same section.}$$

Equivalence Classes

If R is an equivalence relation over A, then for each a ∈ A the **equivalence class** of a, denoted by **[a]**, is the set

$$[a] = \{ x \mid x \, R \, a \}.$$

Consider students taking CS 2212, and

$x \sim y \quad \leftrightarrow \quad x$ and y have the same section.

Equivalence Classes:

{Students in Section 1}

{Students in Section 2}

{Students in Section 3}

{Students in Section 4}

{Students in Section 5}

Equivalence Classes

An equivalence class has important properties:

1. The equivalence classes over A form a **partition** of A.
2. For every pair a, b $\in$ A we have

   either $[a] = [b]$, or $[a] \cap [b] = \emptyset$.

In other words, every element is in only one equivalence class.

Equivalence Relations and Equivalence Classes

Example: Define the relation, R on Z, so that ⟨x, y⟩ ∈ R if and only if x mod 5 = y mod 5.

- **Is this an equivalence relation**? **Yes**. (R S T).
- **There are five equivalence classes** under R corresponding to the five possible values mod 5 {0,5,10...}

1. {1,6,11...}
2. {2,7,12...}
3. {3,8,13...},
4. {4,9,14...}

Note how the equivalence classes form a partition of the relation.