COMP 182 Algorithmic Thinking

Relations

Luay Nakhleh
Computer Science
Rice University

Reading Material

- Chapter 9, Section 1-6

- When we defined the Sorting Problem, we stated that to sort the list, the elements must be totally order-able.
- Mathematical induction doesn't apply only to positive integers; it applies to any well-ordered set.
- What is an order? What is a total order? What is an order that is not total? What is a well-ordered set?
- To answer this question, we learn about relations and special classes of relations, including orders.

- Relations also play an important role beyond the questions on the previous slide.
- For example, the *relational data model* for representing databases is based on the concept of a relation.

Binary Relations

Let A and B be sets. A *binary relation from A to B* is a subset of $A \times B$.

$$aRb \quad \text{denotes} \quad (a, b) \in R$$

$$a \not R b \quad \text{denotes} \quad (a, b) \notin R$$

Binary Relations and Functions

- If f is a function from A to B, then f is a relation from A to B.
- The converse is not necessarily true.

Relation on a Set

A *relation on a set* A is a relation from A to A.

- We have already seen such a relation: $E \subseteq V \times V$.

Properties of Relations

A relation R on a set A is called *reflexive* if $(a, a) \in R$ for every element $a \in A$.

A relation R on a set A is called *symmetric* if $(b, a) \in R$ whenever $(a, b) \in R$, for all $a, b \in A$. A relation R on a set A such that for all $a, b \in A$, if $(a, b) \in R$ and $(b, a) \in R$, then $a = b$ is called *antisymmetric*.

A relation R on a set A is called *transitive* if whenever $(a, b) \in R$ and $(b, c) \in R$, then $(a, c) \in R$, for all $a, b, c \in A$.

Properties of Relations

$$R_1 = \{(a, b) | a \text{ divides } b\}$$

$$R_2 = \{(a, b) | a < b\}$$

$$R_3 = \{(a, b) | a \leq b\}$$

$$R_4 = \{(a, b) | a \equiv_7 b\}$$

$$R_5 = \{(A, B) | A \cap B \neq \emptyset\}$$

Combining Relations

- Relations are sets; therefore, set operations apply to them (e.g., you can take the union of two relations).
- Furthermore, if R is a relation from A to B and S is a relation from B to C, then the <u>composite</u> of R and S, denoted by $S \circ R$, is

$$\{(a,c) | a \in A, c \in C, \exists b \in B, ((a,b) \in R \ \wedge (b,c) \in S)\}$$

Relations Beyond Two Sets

- One can define a relation on any number of sets.

Let $A_1, A_2, \ldots, A_n$ be sets. An *n-ary relation* on these sets is a subset of $A_1 \times A_2 \times \cdots \times A_n$. The sets $A_1, A_2, \ldots, A_n$ are called the *domains* of the relation, and n is called its *degree*.

- Central to database theory and implementation.

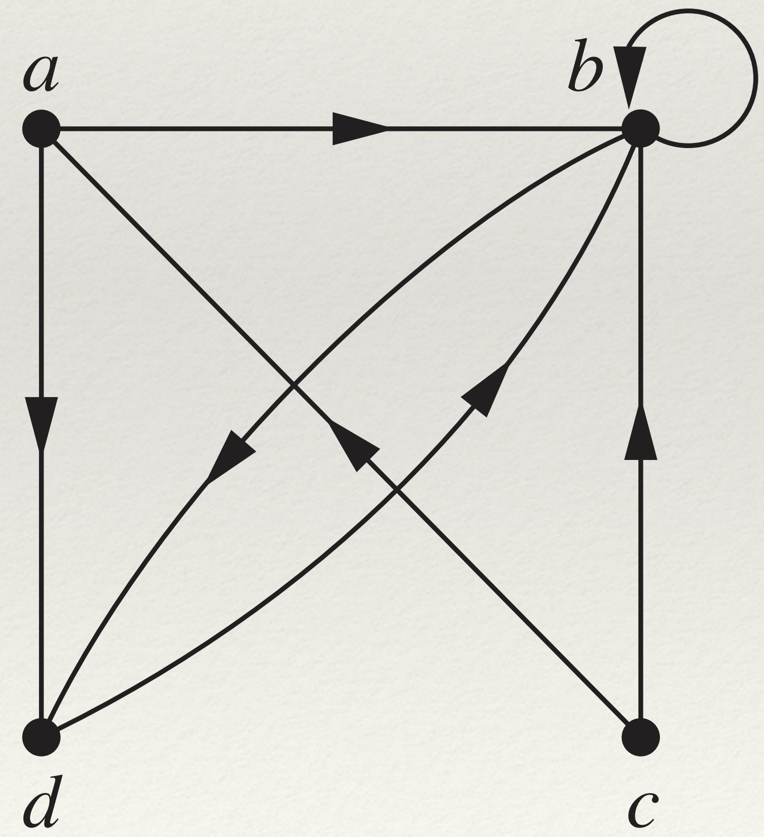
Representing Relations

- Assume $A=\{a_1,a_2,\ldots,a_m\}$ and $B=\{b_1,b_2,\ldots,b_n\}$.
- A relation from A to B can be represented with an $m\times n$ matrix M where:

$$m_{ij} = \begin{cases} 1 \text{ if } (a_i, b_j) \in R, \\ 0 \text{ if } (a_i, b_j) \notin R. \end{cases}$$

Representing Relations

- Relations can also be represented using digraphs, but ones that allow self loops.



Closures of Relations

- Let R be a relation on set A.
- Let P be some property of relations (e.g., reflexivity).
- If there is a relation S with property P containing R such that S is a subset of every relation with property P containing R, then S is called the <u>closure of R with respect to P</u>.

Closures of Relations

- Let R be a relation on set A.
- Reflexive closure of R: $S = R \cup \{(a,a) : a \in A\}$
- Symmetric closure of R: $S = R \cup \{(a,b) : (b,a) \in R\}$

Transitive Closure

- Let R be a relation on set A.
- Recall that R can be represented as a digraph.
- R^n is a relation on set that satisfies: $(a,b) \in R^n$ iff there is a path of length n from a to b in the graph of relation R.
- The transitive closure of R, denoted by R^*, is

$$R^* = \bigcup_{n=1}^{\infty} R^n$$

Transitive Closure

- Can you devise an algorithm for computing the transitive closure of a relation?
- *Hint: Think of matrix multiplication!*

Equivalence Relations

A relation on a set A is called an <u>*equivalence*</u> *relation* if it is reflexive, symmetric, and transitive.

Two elements a and b that are related by an equivalence relation are called <u>*equivalent*</u>. The notation $a \sim b$ is often used to denote that a and b are equivalent elements with respect to a particular equivalence relation.

$$R = \{(a, b) | a \equiv_m b\}$$

Equivalence Relations

Let R be an equivalence relation on a set A. The set of all elements that are related to an element a of A is called the *equivalence class* of a. The equivalence class of a with respect to R is denoted by $[a]_R$. When only one relation is under consideration, we can delete the subscript R and write $[a]$ for this equivalence class.

$$[a]_R = \{s \mid (a, s) \in R\}$$

Let R be an equivalence relation on a set S. Then the equivalence classes of R form a partition of S. Conversely, given a partition $\{A_i \mid i \in I\}$ of the set S, there is an equivalence relation R that has the sets $A_i, i \in I$, as its equivalence classes.

Partial Orders

A relation R on a set S is called a *partial ordering* or *partial order* if it is reflexive, antisymmetric, and transitive. A set S together with a partial ordering R is called a *partially ordered set*, or *poset*, and is denoted by (S, R). Members of S are called *elements* of the poset.

$$S_1 = \mathbb{N} \qquad R_1 = \{(a,b) | a \leq b\}$$

$$S_2 = 2^{\{1,2,3,4,5,6\}} \qquad R_2 = \{(A,B) | A \subseteq B\}$$

The elements a and b of a poset $(S, \preccurlyeq)$ are called *comparable* if either $a \preccurlyeq b$ or $b \preccurlyeq a$. When a and b are elements of S such that neither $a \preccurlyeq b$ nor $b \preccurlyeq a$, a and b are called *incomparable*.

Total Order

If $(S, \preccurlyeq)$ is a poset and every two elements of S are comparable, S is called a *totally ordered* or *linearly ordered set*, and $\preccurlyeq$ is called a *total order* or a *linear order*. A totally ordered set is also called a *chain*.

$S_1 = \mathbb{N}$ $\quad$ $R_1 = \{(a, b) | a \leq b\}$ ✔

$S_2 = 2^{\{1,2,3,4,5,6\}}$ $\quad$ $R_2 = \{(A, B) | A \subseteq B\}$ ✘

$(S, \preccurlyeq)$ is a *well-ordered set* if it is a poset such that $\preccurlyeq$ is a total ordering and every nonempty subset of S has a least element.

$$S_1 = \mathbb{N} \qquad R_1 = \{(a, b) | a \leq b\}$$

$$S_2 = \mathbb{Z}^+ \times \mathbb{Z}^+$$

$$R_2 = \{((a_1, b_1), (a_2, b_2)) : (a_1 < a_2) \vee (a_1 = a_2 \wedge b_1 < b_2)\}$$

THE PRINCIPLE OF WELL-ORDERED INDUCTION Suppose that S is a well-ordered set. Then $P(x)$ is true for all $x \in S$, if

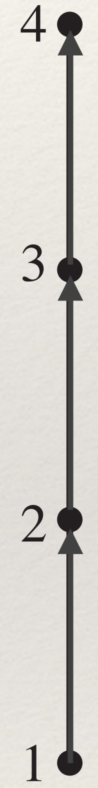
INDUCTIVE STEP: For every $y \in S$, if $P(x)$ is true for all $x \in S$ with $x \prec y$, then $P(y)$ is true.

Hasse Diagrams

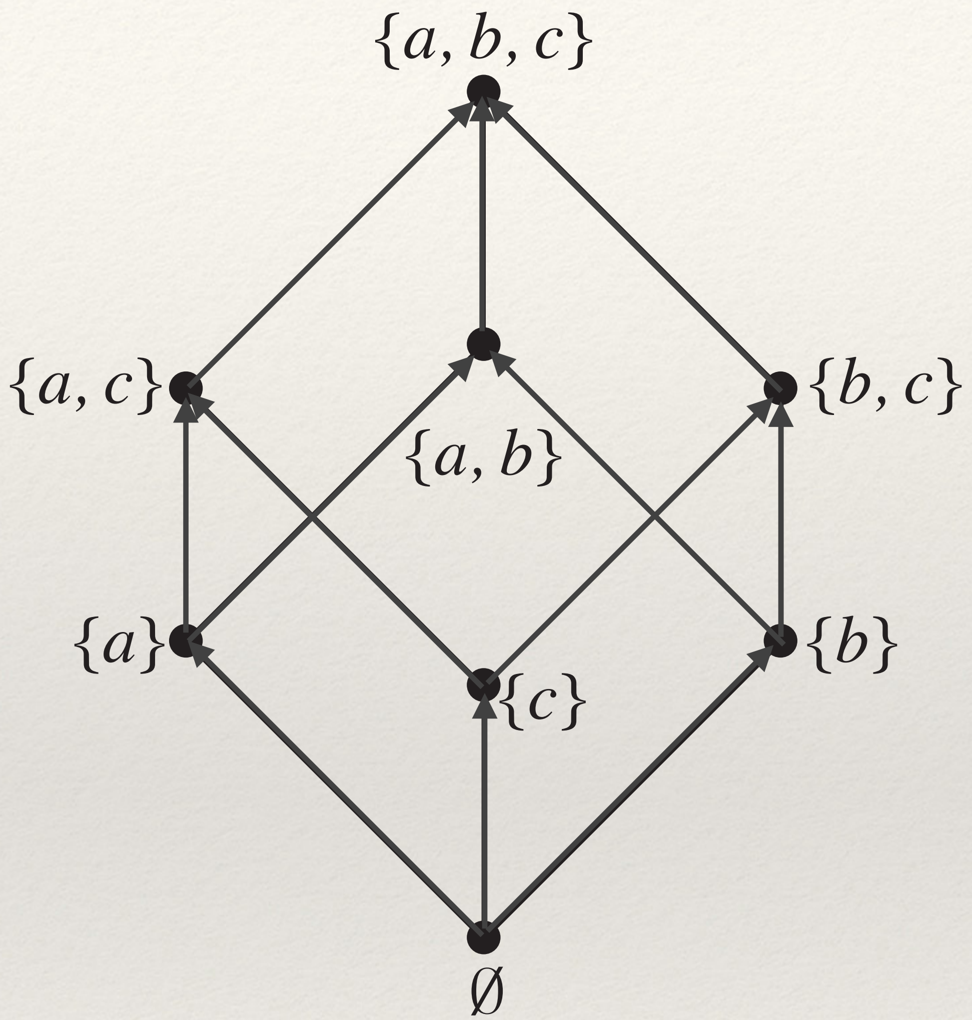
- Let (S,R) be a finite poset.
- The Hasse diagram of (S,R) is a digraph $g=(V,E)$ where
  - $V=S$
  - $R=\{(a,b) \mid aRb \text{ and } \nexists c\ (aRc \text{ and } cRb)\}$

Hasse Diagrams

$(\{1, 2, 3, 4\}, \leq)$



Hasse Diagrams



Hasse diagram of $(P(\{a,b,c\}),\subseteq)$

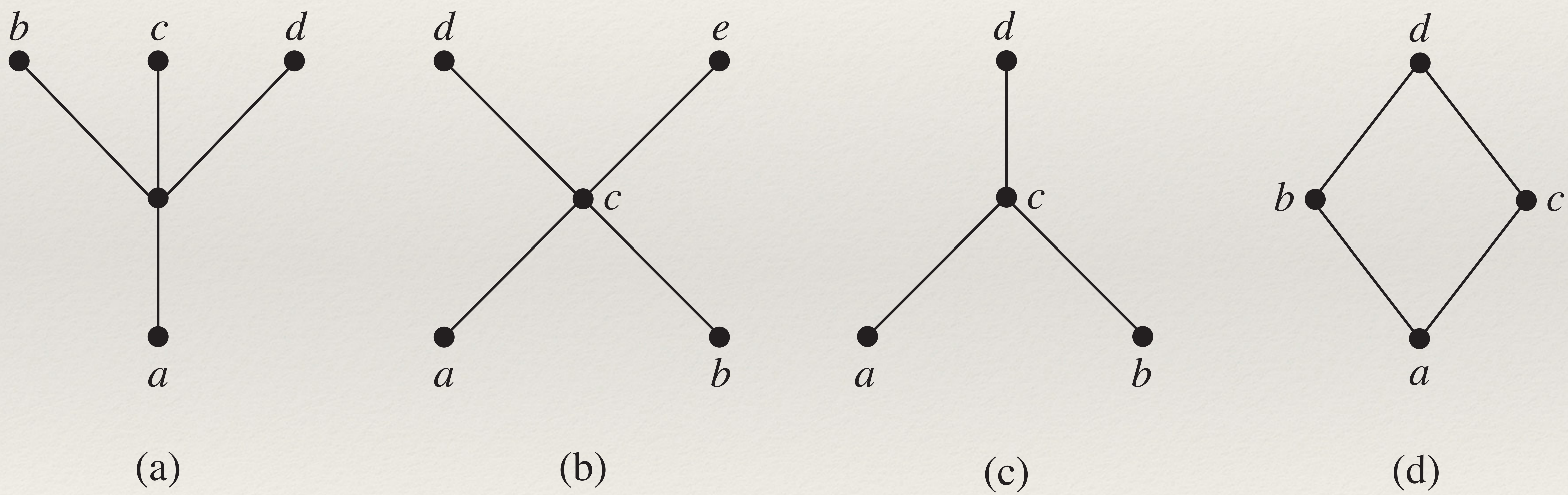
Maximal and Minimal Elements

- Let (S,R) be a poset and $a,b \in S$.
- Element a is <u>maximal</u> if there is no element $c \in S$ $(c \neq a)$ such that aRc.
- Element b is <u>minimal</u> if there is no element $c \in S$ $(c \neq b)$ such that cRb.

Greatest and Least Elements

- Let (S,R) be a poset and $a,b \in S$.
- Element a is the <u>greatest</u> element if for every element $c \in S$, cRa.
- Element b is the <u>least</u> element if for every element $c \in S$, bRc.
- The greatest and least elements are unique when they exist.

- What are the maximal, minimal, greatest, and least elements of the posets given by the following four Hasse diagrams?



Questions?