COSC 290 Discrete Structures

Lecture 27: Relations, IV

Prof. Michael Hay
Monday, Nov. 6, 2017
Colgate University

Plan for today

1. Equivalence relations and partial orders
2. Hasse diagram
3. Topological sort

Equivalence relations and partial orders

Recall: relation properties

For relation R on $A \times A$.

- **R** *reflexive*: for every $a \in A$, $\langle a, a\rangle \in R$.
- **IR** *irreflexive*: for every $a \in A$, $\langle a, a\rangle \notin R$.
- **S** *symmetric*: for every $a, b \in A$, if $\langle a, b\rangle \in R$, then $\langle b, a\rangle \in R$.
- **antiS** *antisymmetric*: for every $a, b \in A$, if $\langle a, b\rangle \in R$ and $\langle b, a\rangle \in R$, then $a = b$.
- **AS** *asymmetric*: for every $a, b \in A$, if $\langle a, b\rangle \in R$, then $\langle b, a\rangle \notin R$.
- **T** *transitive*: for every $a, b, c \in A$, if $\langle a, b\rangle \in R$ and $\langle b, c\rangle \in R$, then $\langle a, c\rangle \in R$.

Special relation: equivalence relation

Relation R on A is an equivalence relation if it is reflexive, symmetric, transitive.

Conventions: use $\equiv$ as the "name" of the relation (as opposed to a letter like R) and use *infix* notation: $a \equiv b$ instead of $\langle a, b\rangle \in \; \equiv$.

Intuition: equivalence relations behave like $=$.

Special relation: partial order

Relation R on A is a partial order if it is reflexive, antisymmetric, transitive.

Conventions: use $\preceq$ as the "name" of the relation (as opposed to a letter like R) and use *infix* notation: $a \preceq b$ instead of $\langle a, b\rangle \in \; \preceq$.

Intuition: partial order relations behave like $\leq$ except that some pairs may be *incomparable*.

Example: the prefixOf relation:

- "a" $\preceq$ "aa"
- "aa" $\preceq$ "aardvark"
- not all pairs comparable: "a" $\not\preceq$ "b" and "b" $\not\preceq$ "a"

Special relation: strict partial order

Relation R on A is a strict partial order if it is irreflexive, (antisymmetric), transitive.

Conventions: use $\prec$ as the "name" of the relation (as opposed to a letter like R) and use *infix* notation: $a \prec b$ instead of $\langle a, b\rangle \in \; \prec$.

Intuition: strict partial order relations behave like $<$ except that some pairs may be *incomparable*.

Example: the ancestorOf relation (ancestor is parent or (recursively) parent of ancestor):

- "DT" $\prec$ "Don Jr"
- "Hanns Drumpf" $\prec$ "DT" (#makedonalddrumpfagain)
- not all pairs comparable: "Harry Potter" $\not\prec$ "Aunt Petunia" and "Aunt Petunia" $\not\prec$ "Harry Potter"

Poll: partial order

Relation $\preceq$ is a partial order if it is reflexive, antisymmetric, transitive.

Consider two relations on a set of track runners:

- $a \preceq_1 b$ if the number of races in which a competed is no more than the number in which b competed.
- $a \preceq_2 b$ if the total amount of time (measured in nanoseconds with laser precision) that a ran is no more than the total amount of time that b ran.

Is $\preceq_1$ a partial order? Is $\preceq_2$ a partial order?

A) Yes, Yes
B) Yes, No
C) No, Yes
D) No, No

Hasse diagram

Hasse diagram

A partial order $\preceq$ on A can be drawn using a Hasse diagram.

- Draw nodes: one node for each A
- Draw edges: edge from a to b if $a \preceq b$, except...
- ... *omit* edges that can be inferred by reflexivity
- ... *omit* edges that can be inferred by transitivity
- ... and *layout* nodes "by level" if $a \preceq b$ for $a \neq b$, then a is placed *lower* than b

Example: isSubstringOf relation on the strings $\{ a, b, c, ab, bc, abc, cd \}$.

Exercise: draw Hasse diagram

Complete the following exercise: on a piece of paper, draw a Hasse diagram for the relation on $A := \{ 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60 \}$ for the relation $R \subseteq A \times A$ where

$$R :- \{\langle x, y \rangle \in A \times A : y \bmod x = 0\}$$

- Draw nodes: one node for each A
- Draw edges: edge from a to b if $a \preceq b$, except...
- ... *omit* edges that can be inferred by reflexivity
- ... *omit* edges that can be inferred by transitivity
- ... and *layout* nodes "by level" if $a \preceq b$ for $a \neq b$, then a is placed *lower* than b

Topological sort

Example

A to do list,

$$[attendClass, sleep, borrowBook, eat, brushTeeth, study]$$

with constraints:

- $borrowBook \preceq study$
- $study \preceq attendClass$
- $sleep \preceq attendClass$
- $eat \preceq brushTeeth$
- $brushTeeth \preceq sleep$

What should you do *first*? Brush teeth? Eat? Borrow book?

Topological ordering

Given a partial order $\preceq$, a topological ordering is a total order $\preceq_{total}$ that is *consistent* with $\preceq$.

Total order

Relation R is a total order if it is a partial order where every pair is comparable (either $\langle a, b\rangle \in R$ or $\langle b, a\rangle \in R$).

A total order can be written succinctly as an ordered list.

Exercise

Suppose you have a findMinimal(S) method that finds a minimal element among S.

x is minimal in S if $\forall y \in S - \{ x \} : y \not\preceq x$

How could you use this to compute the topological sort of a partial order?

Suppose findMinimal(S) had cost $f(n)$ where n is the size of the set. What is the runtime of your algorithm?