CMPSCI 250: Introduction to Computation

Lecture #10: Partial Orders
David Mix Barrington
13 February 2012

Partial Orders

- Definition of a Partial Order
- Total Orders
- The Division Relation
- More Examples of Partial Orders
- Hasse Diagrams
- The Hasse Diagram Theorem
- Proving the Hasse Diagram Theorem

Definition of a Partial Order

- A **partial order** is a particular kind of binary relation on a set. Remember that R is **a binary relation on** a set A if $R \subseteq A \times A$, that is, if R is a set of ordered pairs where both elements of every pair are from A.

- Last time we defined four properties of binary relations on a set. A relation R is **reflexive** if every element is related to itself -- in symbols, $\forall x: R(x, x)$. It is **symmetric** if the order of the two elements in the pair does not matter -- in symbols, $\forall x: \forall y: R(x, y) \rightarrow R(y, x)$. It is **antisymmetric** if the order of elements in a pair can *never* be reversed unless they are the same element -- in symbols, $\forall x: \forall y: (R(x, y) \wedge R(y, x)) \rightarrow (x = y)$. Finally, R is **transitive** if $\forall x: \forall y: \forall z: (R(x, y) \wedge R(y, z)) \rightarrow R(x, z)$. This says that a chains of pairs in the relation must be accompanied by a single pair whose elements are the start and end of the chain.

- A relation is a partial order if it is reflexive, antisymmetric, and transitive.

Diagrams of Binary Relations

- If A is a finite set and R is a binary relation on A, we can draw R in a diagram called a **graph**. We make a dot for each element of A, and draw an arrow from the dot for x to the dot for y whenever R(x, y) is true. If R(x, x), we draw a loop from the dot for x to itself.

- The properties are perhaps easier to see in one of these diagrams. A relation is reflexive if its diagram has a loop at every dot. It is symmetric if every arrow (except loops) has a matching opposite arrow. It is antisymmetric if there are never two arrows in opposite directions between two different nodes. It is transitive if for every path of arrows (a chain where the start of each arrow is the end of the previous one) there is a single arrow from the start of the chain to the end.

- Later in this lecture we'll see a different picture of a relation called a **Hasse diagram**. These are defined only for relations that are partial orders.

Total Orders

- When we studied **sorting** in CMPSCI 187, we assumed that the elements set to be sorted came from a type with a defined **comparison operation**. Given any two elements in the set, we can determine which is "smaller" according to the definition. (In Java the type would have a `compareTo` method or have an associated `Comparator` object.)

- The "smaller" relation is not normally reflexive, but the related "smaller or equal to" relation is. Both these relations are normally antisymmetric, *unless* it is possible for the comparison relations to have ties between different elements. And both relations are transitive, just as $\leq$ is on numbers.

- But ordered sets have an additional property called being **total**, which we write in symbols as $\forall x: \forall y: R(x, y) \vee R(y, x)$. In general a partial order need not have this property -- two distinct elements could be **incomparable**. For example, the **equality relation** E, defined by $E(x, y) \leftrightarrow (x = y)$, is reflexive, antisymmetric, and transitive, but any two distinct elements are incomparable.

The Division Relation

- Here's another example of a partial order that is not total. Our base set will be the natural numbers {0, 1, 2, 3,...}, and we define the **division relation** D so that D(x, y) means "x divides into y without remainder". In symbols, D(x, y) means $\exists z: x \cdot z = y$. (Here we use the dot operator $\cdot$ for multiplication.)

- Any natural divides 0, but 0 divides only itself. D(1, y) is always true. D(2, y) is true for even y's (including 0) but not for odd y's. D(100, x) is true if and only if the decimal for x ends in at least two 0's. In Friday's discussion we'll look at some tricks to test D(x, y) by hand for small values of x.

- It's easy to prove that D is a partial order. D(x, x) is always true because we can take z to be 1 and $x \cdot 1 = x$. If D(x, y) and D(y, x) are both true, x must equal y because D(x, y) implies that $x \leq y$. And if D(x, y) and D(y, z), then there exist naturals u and v such that $x \cdot u = y$ and $y \cdot v = z$, and then $x \cdot (u \cdot v) = z$.

More Examples of Partial Orders

- There are several easily defined partial orders on strings. We say that u is a **prefix** of v if ∃w: uw = v. (Here we write concatenation as algebraic multiplication.) We say u is a **suffix** of v if ∃w: wu = v. And u is a **substring** of v if ∃w: ∃z: wuz = v. It's easy to check that each of these relations is reflexive, antisymmetric, and transitive.

- **Inclusion** on sets is another partial order, as $X \subseteq X$, $X \subseteq Y$ and $Y \subseteq X$ imply $X = Y$, and $X \subseteq Y$ and $Y \subseteq Z$ imply $X \subseteq Z$.

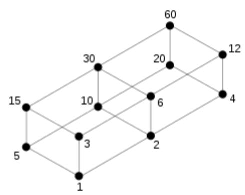
- The **subclass** relation on Java classes is a partial order, since every class is a subclass of itself, two different classes can never each be subclasses of the other, and a subclass of a subclass is a subclass.

- We represent this relation by an object hierarchy diagram in the form of a **tree**. One class is a subclass of another if we can trace a path of `extends` relationships in the diagram from the subclass up to the superclass.

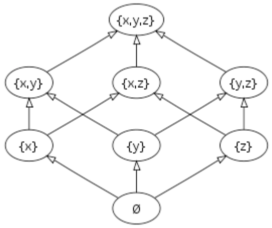
Hasse Diagrams

- We make a Hasse diagram by making a dot for each element of the set, and making lines so that R(x, y) is true if and only if there is a path from x *up to* y. (Relative position of points in a graph usually doesn't matter, but here it does.)

- We make a Hasse diagram from the graph of the partial order by deleting the loops, positioning the dots so all arrows go upward, and deleting arrows that are implied by transitivity from other arrows.



D on Divisors of 60



Inclusion on Sets

Diagrams from wikipedia.org: "Hasse Diagram"

The Hasse Diagram Theorem

- A Hasse diagram is a convenient way to represent a partial order if we can make one. But if I am just given R and told that it is a partial order, can I always make a Hasse diagram for it? The potential problem comes with the rule that the points must be arranged so that every arrow goes upward.

- The **Hasse Diagram Theorem** says that any finite partial order is the "path-below" relation of some Hasse diagram, and the "path-below" relation of any Hasse diagram is a partial order.

- The second of these two statements is easy to prove -- we just have to check that the path-below relation is reflexive, antisymmetric, and transitive.

- We'll sort of prove the first statement, though until we learn mathematical induction we won't have the tools to do it properly.

Proving the Hasse Diagram Theorem

- Given the relation R, we want an arrow from x up to y if R(x, y) is true and $\neg\exists z$: $(x \neq z) \wedge (z \neq y) \wedge R(x, z) \wedge R(z, y)$. (That z would make an x-y arrow redundant.)

- To start drawing the diagram, we need an element that we can safely put at the bottom, because it has no arrows into it. An element x is **minimal** for R if $\forall y$: $R(y, x) \rightarrow (x = y)$. A finite partial order must have at least one minimal element, because we can start somewhere and keep taking smaller elements until none exist. This process can't lead to a **cycle** by antisymmetry.

- We build the diagram **recursively** by finding a minimal element, making a Hasse diagram for the set without that element, and then putting the minimal element back at the bottom, with the arrows given by the rule above.

- We have to make sure that the path-below relation of this diagram is really R.