CMPSCI 250: Introduction to Computation

Lectures #10 and #11: Partial Orders and Equivalence Relations

David Mix Barrington

14 February 2014

Partial Orders, Eq. Relations

- Definition of Partial Orders and Total Orders
- The Division Relation and Other Examples
- Hasse Diagrams
- Definition of Equivalence Relations
- Examples and Their Graphs
- Partitions and the Partition Theorem
- Equivalence Classes Form a Partition

Definition of a Partial Order

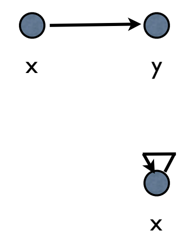
- A **partial order** is a particular kind of binary relation on a set. Remember that R is a **binary relation** on a set A if $R \subseteq A \times A$, that is, if R is a set of ordered pairs where both elements of every pair are from A.
- Last time we used quantifiers to define four particular properties that a binary relation on a set might have.
- A relation is a partial order if and only if it is reflexive, antisymmetric, and transitive.

Properties of a Partial Order

- A relation R is **reflexive** if every element is related to itself -- in symbols, $\forall x: R(x, x)$.
- 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 chain of pairs in the relation must be accompanied by a single pair whose elements are the start and end of the chain.

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.

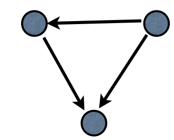


Seeing the Properties

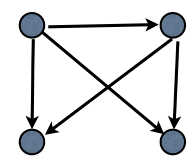
- The properties are perhaps easier to see in one of these diagrams.
- A relation is reflexive if its diagram has a loop at every dot.
- It is symmetric if every arrow (except loops) has a matching opposite arrow.

Seeing the Properties

- It is antisymmetric if there are never two arrows in opposite directions between two different nodes.

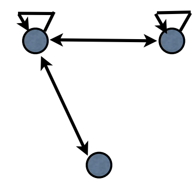


- It is transitive if for every path of arrows (a chain where the start of each arrow is the end of the previous one) there is a single arrow from the start of the chain to the end.



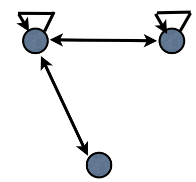
Clicker Question #1

- Which property does the diagrammed relation have?
- (a) reflexive
- (b) antireflexive
- (c) symmetric
- (d) transitive



Answer #1

- Which property does the diagrammed relation have?
- (a) reflexive
- (b) antireflexive
- *(c) symmetric*
- (d) transitive



Total Orders

- When we studied **sorting** in CMPSCI 187, we assumed that the elements 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.)

Total Orders

- 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.

Total Orders

- 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.)

The Division Relation

- 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 Excursion 3.2 the text looks at some tricks to determine whether D(k, y) is true for some particular small values of k.

Division is a Partial Order

- 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 $x \leq y$ (unless x or y is 0).
- 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 we see that $x \cdot (u \cdot v) = z$.

More Partial Order Examples

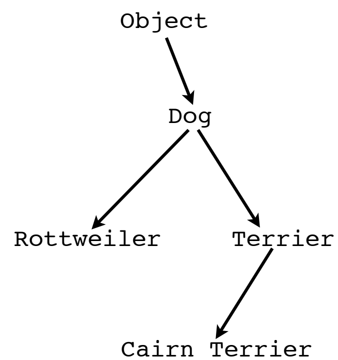
- There are several easily defined partial orders on strings.
- We say that u is a **prefix** of v if ∃w: uw = v. (Here we write concatenation as algebraic multiplication.) We say u is a **suffix** of v if ∃w: wu = v. And u is a **substring** of v if ∃w: ∃z: wuz = v.
- It's easy to check that each of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

More Partial Order Examples

- **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.

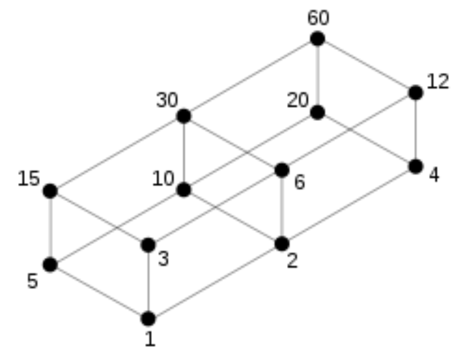
More Partial Order Examples

- 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

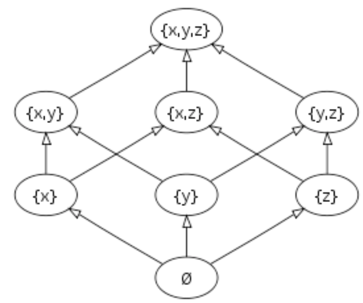
- We make a Hasse diagram by making a dot for each element of the set, and making lines so that R(x, y) is true if and only if there is a path from x up to y.
- (Relative position of points in a graph usually doesn't matter, but here it does.)



Relation D on Divisors of 60
(wikipedia.org)

Hasse Diagram

- Starting from the graph of a partial order, we make a Hasse diagram as follows.
- We first delete the loops.
- We then position the does so the all arrows go upward.
- Finally we delete arrows that are implied by transitivity from other arrows.



Inclusion on Sets
(wikipedia.org)

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

- 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.
- The text proves the first statement -- we'll prove it later using mathematical induction.

Defining an Equivalence Relation

- We have been looking at partial orders, which are reflexive, antisymmetric, and transitive. Now we look at **equivalence relations**: binary relations on a set that are reflexive, symmetric, and transitive.
- Recall the definitions: R is **reflexive** if $\forall x$: $R(x, x)$, R is **symmetric** if $\forall x$: $\forall y$: $R(x, y) \rightarrow R(y, x)$), and R is **transitive** if $\forall x$: $\forall y$: $\forall z$: $(R(x, y) \wedge R(y, z)) \rightarrow R(x, z)$.

Defining an Equivalence Relation

- You should be familiar with these properties of the equality relation: "x = x" is always true, from "x = y" we can get "y = x", and we know that if x = y and y = z, then x = z. The idea of equivalence relations is to formalize the property of acting like equality in this way.
- To prove that a relation is an equivalence relation, we formally need to use the Rule of Generalization, though we often skip steps if they are obvious.

Some Equivalence Relations

- If A is any set, we can define the **universal relation** U on A to *always be true*. Formally, U is the entire set A × A consisting of all possible ordered pairs.
- Of course U(x, x) is always true, and the implications in the definitions of symmetry and transitivity are always true because their conclusions are true.
- The **always false** relation ¬U (or ∅) is symmetric and transitive but not reflexive.

More Equivalence Relations

- The **parity relation** on naturals is perhaps more interesting. We define P(i, j) to be true if i and j are either both even or both odd. Later we'll call this "being congruent modulo 2" and we'll define "being congruent modulo n" in general.
- Any relation of the form "x and y are the same in this respect" will normally be reflexive, symmetric, and transitive, and thus be an equivalence relation.

Clicker Question #2

- Let S be the set of the fifty states in the United States. Which of the following is not an equivalence relation?
- (a) A = {(x, y): states x and y became states in the same year}
- (b) B = {(x, y): states x and y are both states}
- (c) C = {(x, y): states x and y are either equal or share a land border, or both}
- (d) D = {(x, x): state x is a state}

Answer #2

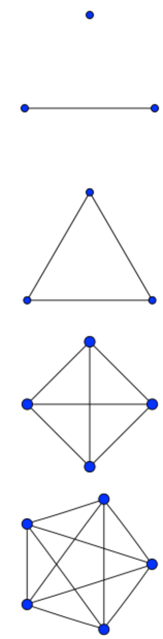
- Let S be the set of the fifty states in the United States. Which of the following is not an equivalence relation?
- (a) A = {(x, y): states x and y became states in the same year}
- (b) B = {(x, y): states x and y are both states}
- *(c) C = {(x, y): states x and y are either equal or share a land border, or both} (not transitive)*
- (d) D = {(x, x): state x is a state}

Graphs of Equivalence Relations

- What happens when we draw the diagram of an equivalence relation?
- Because it is reflexive, we have a loop on every vertex, but we can leave those out for clarity. The arrows are bidirectional because the relation is symmetric.
- The effect of transitivity on the diagram is a bit harder to see.

Complete Graphs

- If we have a set of points that have some connection from each point to each other point, transitivity forces us to have all possible direct connections among those points.
- A graph with all possible undirected edges is called a **complete graph** on its points. The graph of an equivalence relation has a complete graph for each **connected component**.



Partitions

- We've claimed a characterization of the graph of any equivalence relation in terms of complete graphs. Let's now prove that this characterization is correct -- we will need a new definition.
- If A is any set, a **partition** of A is a set of subsets of A -- a set $P = \{S_1, S_2, \ldots, S_k\}$ where (1) each S_i is a subset of A, (2) the union of all the S_i's is A, and (3) the sets are **pairwise disjoint** -- $\forall i: \forall j: (i \neq j) \rightarrow (S_i \cap S_j = \varnothing)$.

Clicker Question #3

- Let D be the set {Cardie, Duncan, Jack, Nala}. Which of these sets of sets is *not* a partition of D?
- (a) {{Nala, Jack}, {Cardie}, {Nala, Duncan}}
- (b) {{Nala}, {Jack}, {Duncan, Cardie}}
- (c) {{Nala, Duncan, Cardie, Jack}
- (d) {{Cardie, Nala, Jack}, {Duncan}}

Answer #3

- Let D be the set {Cardie, Duncan, Jack, Nala}. Which of these sets of sets is *not* a partition of D?
- *(a) {{Nala, Jack}, {Cardie}, {Nala, Duncan}}*
- (b) {{Nala}, {Jack}, {Duncan, Cardie}}
- (c) {{Nala, Duncan, Cardie, Jack}
- (d) {{Cardie, Nala, Jack}, {Duncan}}

The Partition Theorem

- The **Partition Theorem** relates equivalence relations to partitions. It says that a relation is an equivalence relation if and only if it is the "same-set" relation of some partition. In symbols, the same-set relation of P is given by the predicate SS(x, y) defined to be true if $\exists i: (x \in S_i) \wedge (y \in S_i)$.
- So we need to get a partition from any equivalence relation, and an equivalence relation from any partition.

“Same-Set” is an E.R.

- Let P = {S_1, S_2,..., S_k} be a partition of A and let SS be its same set relation. We need to show that SS is an equivalence relation.
- It is easy to check that SS is reflexive, symmetric, and transitive by working with the definition. We’ll look at this in Discussion #4 on Monday.

Equivalence Classes

- If R is an equivalence relation on A, and x is any element of A, we define the **equivalence class** of x, written [x], as the set {y: R(x, y)}, that is, the set of elements of A that are related to x by R.
- The universal relation U has a single equivalence class consisting of all the elements. The equality relation has a separate equivalence class for each element.

Equivalence Classes

- In the parity relation, the set of even numbers forms one equivalence class and the set of odd numbers forms another.
- If we let A be the set of people in the USA, and define R(x, y) to mean "x and y are legal residents of the same state", we get fifty equivalence classes, one for each state. One of them is {x: x is a legal resident of Massachusetts}.

The Classes Form a Partition

- To finish the proof of the Partition Theorem, we must prove that if R is any equivalence relation on A, the set of equivalence classes forms a partition.
- We'll do this with quantifier proof rules in Discussion #4 on Tuesday.