Order Relations

Class 28

Introduction

- our lives are filled with ordered things
- e.g., $<$ on numbers
- what are the properties of an ordered bunch of things?

- if $a < b$ and $b < c$, then we want $a < c$ to be true, so order must be transitive

Introduction

- our lives are filled with ordered things
- e.g., $<$ on numbers
- what are the properties of an ordered bunch of things?

- if $a < b$ and $b < c$, then we want $a < c$ to be true, so order must be transitive
- if $a < b$, we do not want $b < a$, so order must not be symmetric

Introduction

- our lives are filled with ordered things
- e.g., $<$ on numbers
- what are the properties of an ordered bunch of things?

- if $a < b$ and $b < c$, then we want $a < c$ to be true, so order must be transitive
- if $a < b$, we do not want $b < a$, so order must not be symmetric
- but even stronger, if $a \leq b$ and $b \leq a$, then we want $a = b$ to be true, which is the precise definition of antisymmetric

Introduction

- our lives are filled with ordered things
- e.g., $<$ on numbers
- what are the properties of an ordered bunch of things?

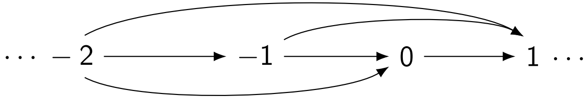
- if $a < b$ and $b < c$, then we want $a < c$ to be true, so order must be transitive
- if $a < b$, we do not want $b < a$, so order must not be symmetric
- but even stronger, if $a \leq b$ and $b \leq a$, then we want $a = b$ to be true, which is the precise definition of antisymmetric

- a relation that is transitive and antisymmetric is an order relation

Total Order

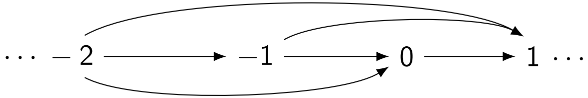
- the canonical order relation is less-than on the set of integers
- for every pair of integers m and n that exists, it is always true that either $m < n$ or $n < m$ but not both
- since every pair of integers can be related by $<$, we call less-than a total order, or a total ordering
- less-than totally orders the integers
- the integers are a totally ordered set with respect to less-than

Less Than



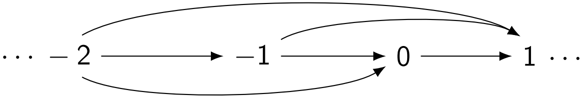
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive?

Less Than



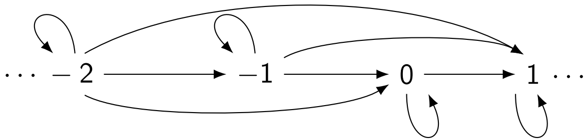
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive? no
- is it irreflexive?

Less Than



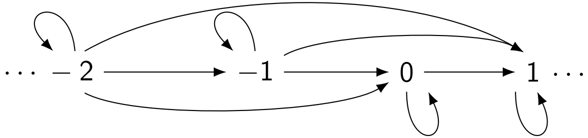
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive? no
- is it irreflexive? yes

Less Than or Equal To



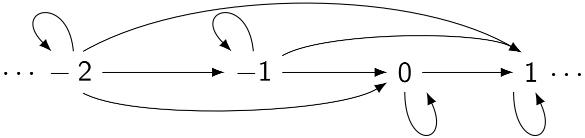
- another canonical total order
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive?

Less Than or Equal To



- another canonical total order
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive? yes
- is it irreflexive?

Less Than or Equal To



- another canonical total order
- since it's an order, we already know that it's transitive and antisymmetric
- is it reflexive? yes
- is it irreflexive? no

Total Order

- a total order is a relation that is
  - transitive
  - antisymmetric
  - either reflexive or irreflexive
- in which every element is related to every other element

Pancakes

1. mix the dry ingredients (flour, sugar, baking powder) in a bowl
2. mix the wet ingredients (milk, eggs) in a bowl
3. mix the mixed dry and wet ingredients together in a bowl to form batter
4. oil the pan
5. heat the pan
6. pour some batter into the pan to cook a first pancake
7. feed the first pancake to the chickens
8. repeatedly pour batter into the pan to cook pancakes

Recipe

- recipe steps are typically presented in a list numbered 1 to n
- you could make pancakes by following the steps in exactly this order
- but you can also make pancakes using different orders
- consider items 1 and 2
  - can 1 come before 2? yes
  - can 2 come before 1? yes
  - can 1 and 2 be done at the same time? yes
- now consider items 1 and 3
  - can 1 come before 3? yes
  - can 3 come before 1? no
  - can 1 and 3 be done at the same time? no
- some pairs of elements have a definite order
- other pairs of elements cannot be ordered

Partial Order

- a partial order is a relation that is
  - transitive
  - antisymmetric
  - either reflexive or irreflexive
- in which some pairs of elements may have no relative order (they cannot be ordered)

- we call a partially ordered set a poset

- just as $\sim$ is the general symbol for an equivalence relation
  $\prec$ is the general symbol for an irreflexive poset
  $\preceq$, a reflexive poset
  (LaTeX: `\prec` and `\preceq`)

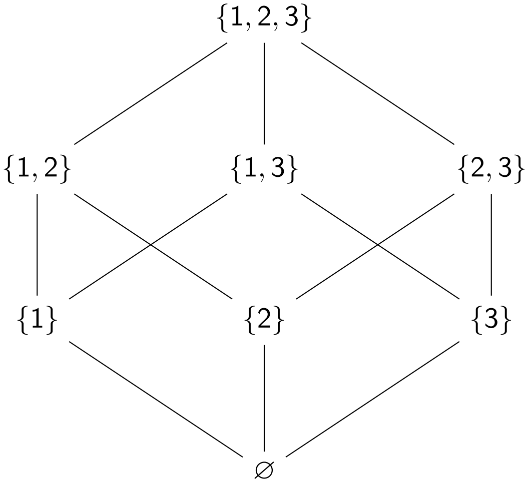
Divides

- consider the relation divides $|$ on N^+
- what are some of the tuples of divides?
- why is divides a poset, and exactly what kind of poset is it?

Poset Diagram

- it is common to diagram a poset by connecting most-closely related elements
- if $x \prec y$ and no element a exists so that $x \prec a \prec y$, then x is drawn lower than y and a line segment connects x and y
- the poset diagram, also known as a Hasse diagram, never shows transitively related tuples
- thus a poset diagram is the transitive reduction of the relation, with directional arrows indicated by position

Poset Diagram



$\langle \{1, 2, 3\}, \subset \rangle$

Pancakes

1. mix dry ingredients
2. mix wet ingredients
3. combine mixed ingredients
4. oil the pan
5. heat the pan
6. cook first pancake
7. feed first pancake to chickens
8. cook remaining batter

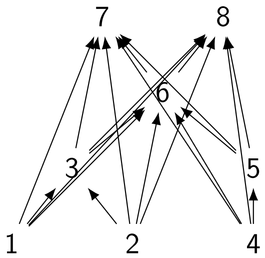
the immediate predecessors

$$\begin{aligned}\prec = \{&(1,3),(2,3),(4,5),(3,6),\\&(5,6),(6,7),(6,8)\}\end{aligned}$$

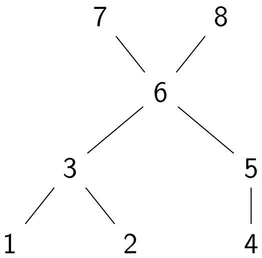
the full relation, with transitive tuples added

$$\begin{aligned}\prec = \{&(1,3),(1,6),(1,7),(1,8),\\&(2,3),(2,6),(2,7),(2,8),\\&(3,6),(3,7),(3,8),(4,5),\\&(4,6),(4,7),(4,8),(5,6),\\&(5,7),(5,8),(6,7),(6,8)\}\end{aligned}$$

Pancakes

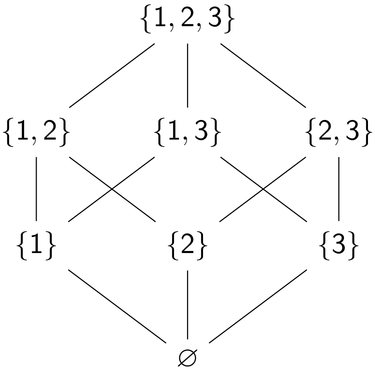


the full relation

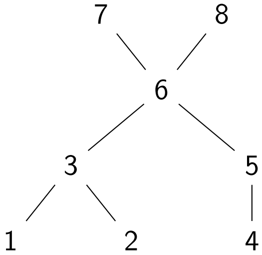


the poset diagram

Two Posets

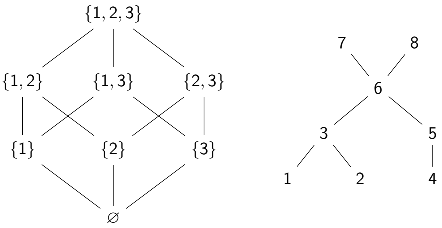


the powerset poset



the pancake poset

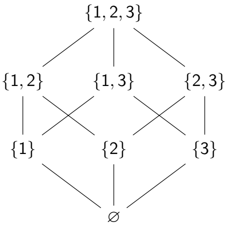
Minimal and Maximal

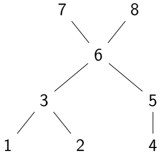


- a minimal element is one that has no predecessors
- $\varnothing$ in the powerset and 1, 2, and 4 in the pancake are minimal elements

- a maximal element has no successors
- $\{1, 2, 3\}$ and 7 and 8 are maximal elements

Least and Greatest



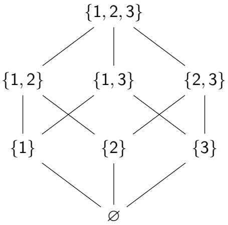


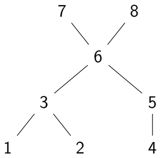
- a least element x is one for which $x \prec y$ for all other elements y in the set
- $\varnothing$ in the powerset is a least element; the pancake poset has no least element

- $\{1, 2, 3\}$ is a greatest element; the pancake poset has no greatest element

- the subset $\{3, 6, 7\}$ of the pancake poset has greatest element 7 and least element 3

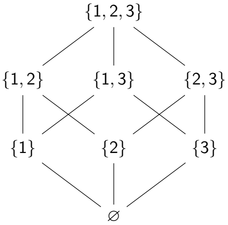
Bounds

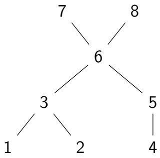




- an element x is a lower bound of a subset of a poset if $x \prec y$ for every element y in the subset
- x does not have to be an element of the poset or of the subset
- $\{2\}$ is a lower bound of the subset $\{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$
- $\varnothing$ is also a lower bound of the subset $\{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$
- 3 and 6 are both lower bounds of the subset $\{7, 8\}$
- 0 is a lower bound of divides on N^+ (even though it's not in the poset)

Least and Greatest Bounds

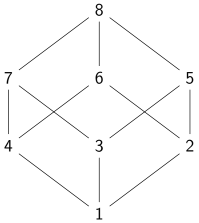




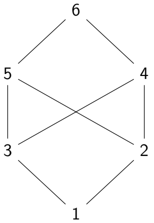
- an element x is a greatest lower bound (glb) of a subset of a poset if it is a lower bound and is greater than every other lower bound
- both $\varnothing$ and $\{2\}$ are lower bounds of the subset $\{\{1,2\},\{2,3\},\{1,2,3\}\}$
- $\{2\}$ is the glb of the subset

- 3 is the lub of $\{1,2\}$

Lattices



a lattice



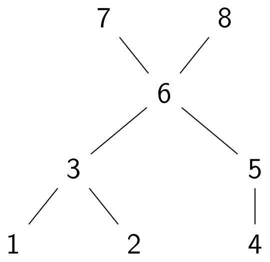
not a lattice

- a lattice is a poset in which every pair of elements has both a glb and a lub within the poset
- the diagram on the left is a lattice
- the diagram on the right is not: 4, 5, and 6 are all upper bounds of $\{2, 3\}$, but none of the is a lub
- no lub exists within the poset for $\{2, 3\}$

Sorting

- sorting is a big deal in CS
- dozens of sorting algorithms have been invented: bubble sort, insertion sort, merge sort, quick sort, Shell sort, heap sort, etc.
- they all apply to totally ordered sets

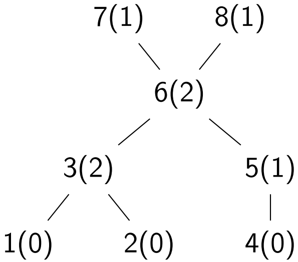
- can you sort a poset? how do you sort a bunch of values when some values cannot be compared to other?
- how can you put these recipe steps in sorted order?



Topological Sort

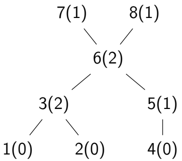
- a poset cannot be totally sorted
- instead we topologically sort a poset; topsort for short

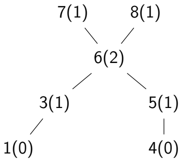
- start by labeling each element with its Hasse diagram in-degree



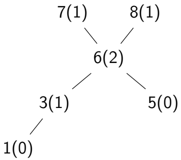
Topsort

```
repeat while elements remain
{
  pick an element of in-degree 0, print it
  remove the element from the diagram
  adjust the remaining elements' in-degrees
}
```





2

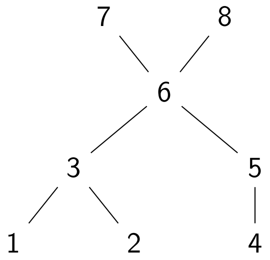


4

Topsort

- if you have helpers, you can do some of the steps of making pancakes in parallel
- this is obvious from the Hasse diagram

- but if you must do the steps in linear order with no parallelism, doing the steps in any valid topsort order will result in pancakes



Well-Founded Orders

- some posets have a special property
- a poset is well-founded if every descending chain of elements is finite

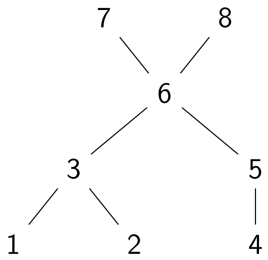
- every subset of integers is well-founded wrt less-than
- for example

$$S = \{x \mid |x| < 10\} = \{-9, -8, -7, \ldots, 7, 8, 9\}$$

- S is well-founded; there is only one descending chain of elements, its least element is -9, and so it is finite

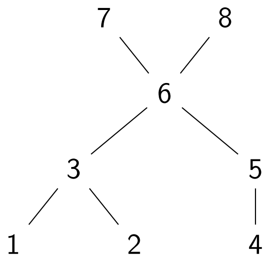
- the powerset of every finite set is well-founded wrt $\subset$
- the empty set is the least element of every descending chain, and the number of subsets is finite, so every descending chain of subsets is finite

Pancakes



- is the pancake poset well-founded?

Pancakes



- is the pancake poset well-founded?

- yes: every descending chain is finite
- different descending chains end at different places, but every one ends

Not Well-Founded

- the set of integers is not well-founded because there is no least element and so the (only) descending chain is infinite
- the set of positive rationals is not well-founded because it has the infinite descending chain

$$\frac{1}{2} > \frac{1}{3} > \frac{1}{4} > \frac{1}{5} > \dots$$

Minimal Element Property

- every non-empty subset of a well-founded poset has a minimal element
- if every non-empty subset of a poset has a minimal element, then the poset is well-founded

- this is fairly obvious
- every subset of the well-founded poset N has a minimal element (which also happens to be a least element, but that is not necessary)
- the subset of positive rationals smaller than 1/2 has no minimal element

Lexicographic Ordering of Tuples

$$\mathsf{N} \times \mathsf{N} = \{(0,0),(0,1),(0,2),\ldots,\\ (1,0),(1,1),(1,2),\ldots,\\ (2,0),(2,1),(2,2),\ldots\}$$

- is this a poset? is it well-founded?
- we can only answer these questions if we define an operation to compare tuples $(x_1, x_2) \prec (y_1, y_2)$
- we define lexicographic ordering so that $(x_1, x_2) \prec (y_1, y_2)$ is true if and only if
  1. $x_1 < y_1$ or
  2. $x_1 = y_1$ and $x_2 < y_2$
- is $(3,5) \prec (5,3)$? yes
- is $(2,4) \prec (2,1)$? no

Lexicographic Ordering of Tuples

- this is a total ordering of $\mathbb{N} \times \mathbb{N}$ and is also a well-founded poset
- $(0, 0)$ is the least element, and every subset has a minimal (and least) element
- do notice that $(1, 0)$ has infinitely many predecessors, but no immediate predecessor

- for tuples with larger cardinality, we extend the definition

$$(x_1, x_2, \ldots, x_k) \prec (y_1, y_2, \ldots, y_k)$$

is true iff zero or more x's and y's starting at the left, in order, are equal, followed by a pair $x_j < y_j$
what comes after the jth pair is irrelevant

Lexicographic Ordering of Strings

- at first glance, strings seem like tuples (using ASCII values for characters)

$$cat \prec con$$

- however, strings can be of arbitrary length
- in other words, for alphabet A, strings are members of A^*

$$cat \stackrel{?}{\prec} catalog$$

- so we need a different definition of $\prec$ for strings

- it turns out that there are multiple lexicographic orderings that can be defined on strings

Dictionary Ordering of Strings

- we define the dictionary ordering of strings
  1. if x and y are the same length, then use the same definition as for tuples
  2. if x is a proper prefix of y, then $x \prec y$
  3. if $x = uv$ and $y = uw$, then $x \prec y$ if $v \prec w$
- this is a total order, as every pair of strings can be compared
- this is not a well-founded ordering

$$\cdots \prec a^{\infty}b \prec \cdots \prec aaab \prec aab \prec ab \prec b$$

this is not a finite descending chain

Standard Ordering of Strings

- sometimes we need a well-founded ordering for strings
  1. $x \prec y$ if length(x) $<$ length(y)
  2. $x \prec y$ if they are the same length and $x \prec y$ based on their dictionary ordering
- this is a total ordering and a well-founded ordering
- the first few strings over $\{a, b\}$ are

$$\Lambda, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa$$