CS 220: Discrete Structures and their Applications

partial orders, DAGs and n-ary relations
Recap Binary Relations

The relation $R$ is **reflexive** if for every $x \in A$, $xRx$.

Example: the less-or-equal to relation on the positive integers

The relation $R$ is **anti-reflexive** if for every $x \in A$, it is not true that $xRx$.

Example: the less-than relation

The relation $R$ is **transitive** if for every $x, y, z \in A$, $xRy$ and $yRz$ imply that $xRz$.

Example: the ancestor relation

The relation $R$ is **symmetric** if for every $x, y \in A$, $xRy$ implies that $yRx$.

Example: $R = \{(a, b) : a, b$ are actors that have played in the same movie}$

The relation $R$ is **anti-symmetric** if for every $x, y \in A$, $xRy$ and $yRx$ imply that $x = y$.

Example: less-or-equal $\quad a \leq b$ and $b \leq a \Rightarrow a = b$
Let's look at the graphs for the following relations:

\( x \leq y \)  
\( x \) evenly divides \( y \)

What properties do these relations have (symmetric, anti-symmetric, reflexive, anti-reflexive, transitive).
A relation $R$ on a set $A$ is a partial order if it is reflexive, transitive, and anti-symmetric.

The notation $a \preceq b$ is used to reflect the fact that a partial order acts like the $\leq$ operator on the elements of $A$.

The domain along with a partial order defined on it is denoted $(A, \preceq)$ and is called a partially ordered set or poset.

Example: The $\leq$ operator acting on the set of integers is a partial order, denoted by $(\mathbb{Z}, \leq)$. 
Two elements of a partially ordered set, $x$ and $y$, are said to be comparable if $x \leq y$ or $y \leq x$.

Otherwise they are said to be incomparable.

A partial order is a total order if every two elements in the domain are comparable. The partial order $(\mathbb{Z}, \leq)$ is an example of a total order.
An element $x$ is a **minimal** element if there is no $y \neq x$ such that $y \leq x$.

An element $x$ is a **maximal** element if there is no $y \neq x$ such that $x \leq y$. 

$x \leq y$

$x$ evenly divides $y$
partial orders

Example:
Is the following a partial order?

Properties? reflexive, transitive, anti-symmetric?

What are the minimal/maximal elements?
Is the following a partial order?

The domain is a set of students at a school. $x \preceq y$ if $x$ has the same birthday as $y$.

Is it transitive?
Is it reflexive?
Is it anti-symmetric?
strict orders

A relation \( R \) is a **strict order** if \( R \) is transitive and anti-reflexive.

The notation \( a \prec b \) is used to express that "a is less than b".

The domain along with the strict order defined on it is called a **strictly ordered set** and is denoted by \((A, \prec)\).

The definitions of comparable, incomparable, minimal, maximal are the same as for partial orders.
strict orders

Example:
Is the following a strict order?

Properties? anti-reflexive, transitive, anti-symmetric?
Example:
Is the following a strict order?

Properties? anti-reflexive, transitive, anti-symmetric?

A relation $R$ that is transitive and anti-reflexive is also anti-symmetric
strict orders

Given a finite set $A$, let's check if $(P(A), \subset)$ is a strict order.

The domain is $P(A)$, the set of all subsets of $A$.

Two subsets of $A$, $X$ and $Y$, are in the relation if $X \subset Y$.

Is it transitive? anti-reflexive?
Question: Is there a cycle in this graph?
graphs describing precedence

Examples:
- prerequisites for a set of courses
- dependencies between programs (for installation and compilation)

Edge from \(a\) to \(b\) indicates \(a\) should come before \(b\)

Batman images are from the book “Introduction to bioinformatics algorithms”
A directed acyclic graph (DAG) is a directed graph that has no cycles.
A directed acyclic graph (DAG) is a directed graph that has no cycles.

Is it a strict order?
A directed acyclic graph (DAG) is a directed graph that has no cycles.

Is it a strict order?

No, it's not transitive. But its transitive closure is!
graphs describing precedence

Want an ordering of the vertices of the graph that respects the precedence relation

- Example: An ordering of CS courses

**Topological sort:** listing of nodes such that if \((a,b)\) is an edge, \(a\) appears before \(b\) in the list
Question: Which course needs to be first? second? third?

order for CS/ACT courses
graphs describing precedence

Want an ordering of the vertices of the graph that respects the precedence relation

**Topological sort:** listing of nodes such that if \((a,b)\) is an edge, \(a\) appears before \(b\) in the list

Is a topological sort unique?
topological sort

- Pick a vertex $x$ with in-degree 0 and remove $x$ from $G$, including all its outgoing edges.
- Then pick another vertex with in-degree 0 from the remaining vertices.
- Keep selecting vertices until no vertices left.
IDEA: nodes with no successors can be added to the back of the list

A, D, E, B, G, C, F, H, I
directed acyclic graphs

DAGs are an important class of graphs

Used for representing probabilistic relationships between variables (Bayesian networks)
Are at the core of dataflow programming (TensorFlow)

Many computational problems that are NP-hard on general graphs can be solved efficiently on DAGs
$n$-ary relations

Definition: Let $A_1, A_2, \ldots, A_n$ be sets.
An $n$-ary relation on these sets is a subset of

$$A_1 \times A_2 \times \ldots \times A_n.$$  

The sets $A_1, A_2, \ldots, A_n$ are called the *domains* of the relation, and $n$ is called its *degree*.

Example: The *between* relation consisting of triples $(a, b, c)$ where $a, b, c$ are integers such that $a < b < c$. 
**x thinks that y likes z**

<table>
<thead>
<tr>
<th>Person x</th>
<th>Person y</th>
<th>Person z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>Bob</td>
<td>Denise</td>
</tr>
<tr>
<td>Charles</td>
<td>Alice</td>
<td>Bob</td>
</tr>
<tr>
<td>Charles</td>
<td>Charles</td>
<td>Charles</td>
</tr>
<tr>
<td>Denise</td>
<td>Denise</td>
<td>Denise</td>
</tr>
</tbody>
</table>
Databases defined by relations are called *relational databases*.

<table>
<thead>
<tr>
<th>Students</th>
<th>StudentName</th>
<th>IDnumber</th>
<th>Major</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ackermann</td>
<td>231455</td>
<td>Computer Science</td>
<td>3.88</td>
<td></td>
</tr>
<tr>
<td>Adams</td>
<td>888323</td>
<td>Physics</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>Chou</td>
<td>102147</td>
<td>Computer Science</td>
<td>3.49</td>
<td></td>
</tr>
<tr>
<td>Goodfriend</td>
<td>453876</td>
<td>Mathematics</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>Rao</td>
<td>678543</td>
<td>Mathematics</td>
<td>3.90</td>
<td></td>
</tr>
<tr>
<td>Stevens</td>
<td>786576</td>
<td>Psychology</td>
<td>2.99</td>
<td></td>
</tr>
</tbody>
</table>