Lecture 11: relations

- relations are like tables of data
- we'll give def's, examples
- properties of rel's
- equivalence classes

Applications

- relations are central to the theory of database systems
- Many important algorithms (e.g. page rank) operate on relations
Def.: A relation $R$ on sets $A, B, C, \ldots$ is a subset of $A \times B \times C \times \ldots$

Def.: A binary rela. $R$ on a set $A$ is a subset of $A \times A$

Ex.: equality, "$\leq", "\geq$"

Ex.: isFriend is a relation on the set of people. isSaying, ... $R$

$\left( (1, 2) \in \"\leq\" \right)$  $\left( 1 \leq 2 \right)$

$\text{(Alice, Bob)} \in \text{Friends}$  Alex is a friend of Bob

Notation: if $R$ is a bin. rela. on $A$, and $x, y \in A$ then "$x R y$" means $(x, y) \in R$

Notation: I usually use "$R^n$" or "$\cdot \cdot \cdot \cdot \cdot \cdot ^{\ldots}$" to represent rela.'s.

\[ F : A \rightarrow B \]

\[ R \text{ on set } A \]

\[ a \rightarrow 1 \]
\[ b \rightarrow 2 \]
\[ c \rightarrow 3 \]

\[ a R b, b R a \]
\[ c R b \]

\[ \text{Set } = \{ (ab), (b, a), (cb) \} \]

\[ c \]

(Sometimes called a "directed graph")
Properties that relations can have:

- R is a relation on set A
  - R is reflexive if \( \forall x \in A, xRx \).

- R is symmetric if \( \forall x, y \in A, xRy \implies yRx \).

- R is transitive if \( \forall x, y, z \in A, xRy \land yRz \implies xRz \).

R is an equivalence relation if:
- R is reflexive, symmetric, and transitive.

≤ relation on \( \mathbb{N} \):
- Is \( \leq \) reflexive?
  - Yes, \( \forall x, x \leq x \)

- Is \( \leq \) symmetric?
  - No: example, \( 1 \leq 2 \) but \( 2 \not\leq 1 \).

- Is \( \leq \) transitive?
  - Yes, if \( a \leq b \) and \( b \leq c \), then \( a \leq c \).

- Is \( \leq \) an equivalence relation?

"a \rightarrow b" is symmetric
(Not sensible; symmetry is a property of \( R \), not \( \rightarrow \)).
$R'$ is transitive closure of $R$. (Smallest extension of $R$ that is transitive)

is transitive not reflexive.

you can have any combination of reflexive, symmetric, transitive.

$x = a, y = b, z = c$

$xRy, yRz, xRz$

$x = c, y = a, z = d$

$xRa, xRb, xRc, xRd.$

$x = b, y = d, z = b$

$xRy, yRx, xRz$

$bRb.$
Often redefine equality for new types of objects.

\[(\text{sets}) \quad A = B \quad \text{if} \quad \forall x \in A, x \in B \quad \text{and} \quad \forall x \in B, x \in A\]

\[(\text{funs}) \quad f = g \quad \text{if} \quad \forall x \in A, f(x) = g(x)\]

\[(\text{pairs}) \quad (a,b) = (c,d) \quad \text{if} \quad a = c, b = d.\]

Many prog. langs let you redefine `==`, `equals()`

--eq--

if you define equality, check that
your def is equivalence relation.

```cpp
if (a == b) {
    ...
}
```

```cpp
x = 17
```

```cpp
if (x == 17) {
    ...
}
```