Representation of Relations

INTRODUCTION TO RELATIONS

- When (a, b) belongs to R, a is said to be **related** to b by R.
- **Example:** Let P be a set of people, C be a set of cars, and D be the relation describing which person drives which car(s).
- P = {Carl, Suzanne, Peter, Carla},
- C = {Mercedes, BMW, tricycle}
- D = {(Carl, Mercedes), (Suzanne, Mercedes), (Suzanne, BMW), (Peter, tricycle)}
- This means that Carl drives a Mercedes, Suzanne drives a Mercedes and a BMW, Peter drives a tricycle, and Carla does not drive any of these vehicles.

RELATIONS

- If we want to describe a relationship between elements of two sets A and B, we can use **ordered pairs** with their first element taken from A and their second element taken from B.
- Since this is a relation between **two sets**, it is called a **binary relation**.
- **Definition:** Let A and B be sets. A binary relation from A to B is a subset of $A \times B$.
- In other words, for a binary relation R we have $R \subseteq A \times B$. We use the notation aRb to denote that $(a, b) \in R$ and $a\underline{R}b$ to denote that $(a, b) \notin R$.

RELATIONS ON A SET

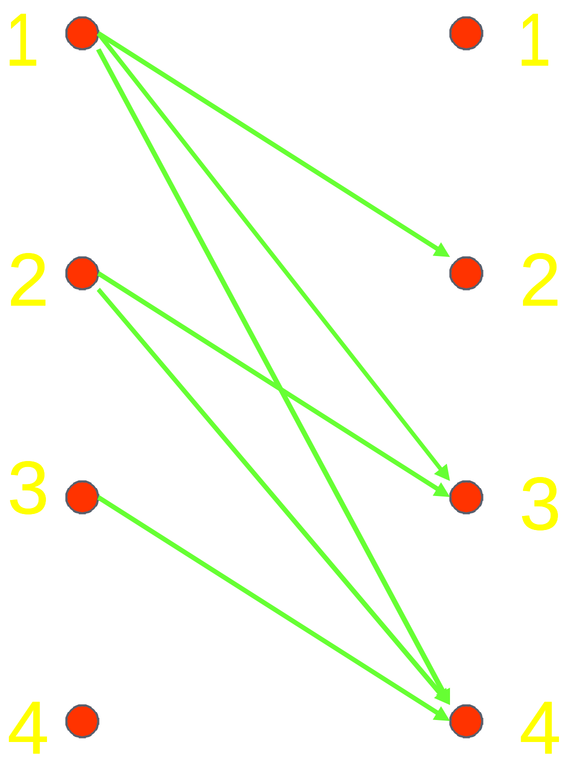
- **Definition:** A relation on the set A is a relation from A to A.

- In other words, a relation on the set A is a subset of $A \times A$.

- **Example:** Let $A = \{1, 2, 3, 4\}$. Which ordered pairs are in the relation $R = \{(a, b) \mid a < b\}$?

RELATIONS ON A SET

- **Solution:** R = {(1, 2)(1, 3), (1, 4), (2, 3),(2, 4),(3, 4)}



R	1	2	3	4
1		X	X	X
2			X	X
3				X
4				

Relations on a Set

- **How many different relations can we define on a set A with n elements?**

- A relation on a set A is a subset of $A \times A$.
- How many elements are in $A \times A$?

- There are n^2 elements in $A \times A$, so how many subsets (= relations on A) does $A \times A$ have?

- The number of subsets that we can form out of a set with m elements is 2^m. Therefore, 2^{n^2} subsets can be formed out of $A \times A$.

- **Answer:** We can define 2^{n^2} different relations on A.

Properties of Relations

- We will now look at some useful ways to classify relations.
- **Definition:** A relation R on a set A is called **reflexive** if $(a, a) \in R$ for every element $a \in A$.
- Are the following relations on {1, 2, 3, 4} reflexive?

R = {(1, 1), (1, 2), (2, 3), (3, 3), (4, 4)} No.

R = {(1, 1), (2, 2), (2, 3), (3, 3), (4, 4)} Yes.

R = {(1, 1), (2, 2), (3, 3)} No.

Definition: A relation on a set A is called **irreflexive** if $(a, a) \notin R$ for every element $a \in A$.

Properties of Relations

- **Definitions:**
- A relation R on a set A is called **symmetric** if $(b, a) \in R$ whenever $(a, b) \in R$ for all $a, b \in A$.
- A relation R on a set A is called **antisymmetric** if $a = b$ whenever $(a, b) \in R$ and $(b, a) \in R$.
- A relation R on a set A is called **asymmetric** if $(a, b) \in R$ implies that $(b, a) \notin R$ for all $a, b \in A$.

Properties of Relations

- Are the following relations on {1, 2, 3, 4} symmetric, antisymmetric, or asymmetric?

R = {(1, 1), (1, 2), (2, 1), (3, 3), (4, 4)} symmetric

R = {(1, 1)} sym. and antisym.

R = {(1, 3), (3, 2), (2, 1)} antisym. and asym.

R = {(4, 4), (3, 3), (1, 4)} antisym.

PROPERTIES OF RELATIONS

- **Definition:** A relation R on a set A is called **transitive** if whenever (a, b)$\in$R and (b, c)$\in$R, then (a, c)$\in$R for a, b, c$\in$A.

- Are the following relations on {1, 2, 3, 4} transitive?

R = {(1, 1), (1, 2), (2, 2), (2, 1), (3, 3)} Yes.

R = {(1, 3), (3, 2), (2, 1)} No.

R = {(2, 4), (4, 3), (2, 3), (4, 1)} No.

Counting Relations

- **Example:** How many different reflexive relations can be defined on a set A containing n elements?

- **Solution:** Relations on R are subsets of $A \times A$, which contains n^2 elements.
- Therefore, different relations on A can be generated by choosing different subsets out of these n^2 elements, so there are 2^{n^2} relations.
- A **reflexive** relation, however, **must** contain the n elements (a, a) for every $a \in A$.
- Consequently, we can only choose among $n^2 - n = n(n - 1)$ elements to generate reflexive relations, so there are $2^{n(n-1)}$ of them.

N-ARY RELATIONS

- In order to study an interesting application of relations, namely **databases**, we first need to generalize the concept of binary relations to **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**.

N-ARY RELATIONS

- **Example:**

  Let $R = \{(a, b, c) \mid a = 2b \wedge b = 2c \text{ with } a, b, c \in \mathbf{N}\}$
- What is the degree of R?

  The degree of R is 3, so its elements are triples.
- What are its domains?

  Its domains are all equal to the set of integers.
- Is (2, 4, 8) in R?

  No.
- Is (4, 2, 1) in R?

  Yes.

Databases and Relations

- Let us take a look at a type of database representation that is based on relations, namely the **relational data model.**
- A database consists of n-tuples called **records**, which are made up of **fields**.
- These fields are the **entries** of the n-tuples.
- The relational data model represents a database as an n-ary relation, that is, a set of records.

Databases and Relations

- **Example:** Consider a database of students, whose records are represented as 4-tuples with the fields **Student Name**, **ID Number**, **Major**, and **GPA**:

- R = {(Ackermann, 231455, CS, 3.88),
  (Adams, 888323, Physics, 3.45),
  (Chou, 102147, CS, 3.79),
  (Goodfriend, 453876, Math, 3.45),
  (Rao, 678543, Math, 3.90),
  (Stevens, 786576, Psych, 2.99)}

- Relations that represent databases are also called **tables**, since they are often displayed as tables.

Databases and Relations

- A domain of an n-ary relation is called a **primary key** if the n-tuples are uniquely determined by their values from this domain.
- This means that no two records have the same value from the same primary key.
- In our example, which of the fields **Student Name**, **ID Number**, **Major**, and **GPA** are primary keys?
- **Student Name** and **ID Number** are primary keys, because no two students have identical values in these fields.
- In a real student database, only **ID Number** would be a primary key.

DATABASES AND RELATIONS

- In a database, a primary key should remain one even if new records are added.
- Therefore, we should use a primary key of the **intension** of the database, containing all the n-tuples that can ever be included in our database.

- **Combinations of domains** can also uniquely identify n-tuples in an n-ary relation.
- When the values of a **set of domains** determine an n-tuple in a relation, the **Cartesian product** of these domains is called a **composite key**.

DATABASES AND RELATIONS

- We can apply a variety of **operations** on n-ary relations to form new relations.

- **Definition:** The **projection** $P_{i_1, i_2, \ldots, i_m}$ maps the n-tuple $(a_1, a_2, \ldots, a_n)$ to the m-tuple $(a_{i_1}, a_{i_2}, \ldots, a_{i_m})$, where $m \leq n$.

- In other words, a projection $P_{i_1, i_2, \ldots, i_m}$ keeps the m components $a_{i_1}, a_{i_2}, \ldots, a_{i_m}$ of an n-tuple and deletes its (n – m) other components.

- **Example:** What is the result when we apply the projection $P_{2,4}$ to the student record (Stevens, 786576, Psych, 2.99) ?

- **Solution:** It is the pair (786576, 2.99).

DATABASES AND RELATIONS

- In some cases, applying a projection to an entire table may not only result in fewer columns, but also in **fewer rows**.
- Why is that?
- Some records may only have differed in those fields that were deleted, so they become **identical**, and there is no need to list identical records more than once.

DATABASES AND RELATIONS

- We can use the **join** operation to combine two tables into one if they share some identical fields.

- **Definition:** Let R be a relation of degree m and S a relation of degree n. The **join** $J_p(R, S)$, where $p \leq m$ and $p \leq n$, is a relation of degree $m + n - p$ that consists of all $(m + n - p)$-tuples
$(a_1, a_2, \ldots, a_{m-p}, c_1, c_2, \ldots, c_p, b_1, b_2, \ldots, b_{n-p})$,
where the m-tuple $(a_1, a_2, \ldots, a_{m-p}, c_1, c_2, \ldots, c_p)$ belongs to R and the n-tuple $(c_1, c_2, \ldots, c_p, b_1, b_2, \ldots, b_{n-p})$ belongs to S.

DATABASES AND RELATIONS

- In other words, to generate Jp(R, S), we have to find all the elements in R whose p last components match the p first components of an element in S.

- The new relation contains exactly these matches, which are combined to tuples that contain each matching field only once.

DATABASES AND RELATIONS

- **Example:** What is $J_1(Y, R)$, where Y contains the fields **Student Name** and **Year of Birth**,
- Y = {(1978, Ackermann),
  (1972, Adams),
  (1917, Chou),
  (1984, Goodfriend),
  (1982, Rao),
  (1970, Stevens)},
- and R contains the student records as defined before ?

DATABASES AND RELATIONS

- **Solution:** The resulting relation is:
- {(1978, Ackermann, 231455, CS, 3.88),
  (1972, Adams, 888323, Physics, 3.45),
  (1917, Chou, 102147, CS, 3.79),
  (1984, Goodfriend, 453876, Math, 3.45),
  (1982, Rao, 678543, Math, 3.90),
  (1970, Stevens, 786576, Psych, 2.99)}

- Since Y has two fields and R has four, the relation $J_1(Y, R)$ has 2 + 4 – 1 = 5 fields.

REPRESENTING RELATIONS

- We already know different ways of representing relations. We will now take a closer look at two ways of representation: **Zero-one matrices** and **directed graphs**.
- If R is a relation from $A = \{a_1, a_2, \ldots, a_m\}$ to $B = \{b_1, b_2, \ldots, b_n\}$, then R can be represented by the zero-one matrix $M_R = [m_{ij}]$ with
- $m_{ij} = 1$, if $(a_i, b_j) \in R$, and
- $m_{ij} = 0$, if $(a_i, b_j) \notin R$.
- Note that for creating this matrix we first need to list the elements in A and B in a **particular, but arbitrary order**.

Equivalence Relations

- **Equivalence relations** are used to relate objects that are similar in some way.
- **Definition:** A relation on a set A is called an equivalence relation if it is reflexive, symmetric, and transitive.
- Two elements that are related by an equivalence relation R are called **equivalent**.

EQUIVALENCE RELATIONS

- Since R is **symmetric**, a is equivalent to b whenever b is equivalent to a.
- Since R is **reflexive**, every element is equivalent to itself.
- Since R is **transitive,** if a and b are equivalent and b and c are equivalent, then a and c are equivalent.
- Obviously, these three properties are necessary for a reasonable definition of equivalence.

EQUIVALENCE RELATIONS

- **Example:** Suppose that R is the relation on the set of strings that consist of English letters such that aRb if and only if l(a) = l(b), where l(x) is the length of the string x. Is R an equivalence relation?
- **Solution:**
  - R is reflexive, because l(a) = l(a) and therefore aRa for any string a.
  - R is symmetric, because if l(a) = l(b) then l(b) = l(a), so if aRb then bRa.
  - R is transitive, because if l(a) = l(b) and l(b) = l(c), then l(a) = l(c), so aRb and bRc implies aRc.
- R is an equivalence relation.

Partitions

- Consider the relation $R = \{ (a,b) \mid a \equiv b \bmod 2\}$
- This splits the integers into two equivalence classes: even numbers and odd numbers
- Those two sets together form a partition of the integers
- Formally, a <u>partition of a set *S*</u> is a collection of non-empty disjoint subsets of S whose union is *S*
- In this example, the partition is { [0], [1] }
  - Or { {…, -3, -1, 1, 3, …}, {…, -4, -2, 0, 2, 4, …} }

Application & Scope of research

- Database system