CHAPTER 8

RELATIONS

SECTION 8.1

Relations on Sets

Relations on Sets

A more formal way to refer to the kind of relation defined on sets is to call it a **binary relation** because it is a subset of a Cartesian product of two sets.

At the end of this section we define an *n-ary relation* to be a subset of a Cartesian product of *n* sets, where *n* is any integer greater than or equal to two.

Such a relation is the fundamental structure used in relational databases. However, because we focus on binary relations in this text, when we use the term *relation* by itself, we will mean binary relation.

Example 2 – *The Congruence Modulo 2 Relation*

Define a relation E from **Z** to **Z** as follows: For all $(m, n) \in \mathbf{Z} \times \mathbf{Z}$,

$$m\ E\ n \quad \Leftrightarrow \quad m - n \text{ is even.}$$

a. Is $4\ E\ 0$? Is $2\ E\ 6$? Is $3\ E\ (–3)$? Is $5\ E\ 2$?

b. List five integers that are related by E to 1.

c. Prove that if n is any odd integer, then $n\ E\ 1$.

Solution:

a. Yes, $4\ E\ 0$ because $4 – 0 = 4$ and 4 is even.

Yes, $2\ E\ 6$ because $2 – 6 = –4$ and $–4$ is even.

Example 2 – *Solution*

cont'd

Yes, 3 E (–3) because 3 – (–3) = 6 and 6 is even.

No, 5 $\not{E}$ 2 because 5 – 2 = 3 and 3 is not even.

b. There are many such lists. One is

1 because 1 – 1 = 0 is even,

3 because 3 – 1 = 2 is even,

5 because 5 – 1 = 4 is even,

–1 because –1 – 1 = –2 is even,

–3 because –3 – 1 = –4 is even.

Example 1 – *Solution*

cont'd

c. Proof:

Suppose n is any odd integer.

Then $n = 2k + 1$ for some integer k. Now by definition of E, $n\ E\ 1$ if, and only if, $n - 1$ is even.

But by substitution,

$$n - 1 = (2k + 1) - 1 = 2k,$$

and since k is an integer, $2k$ is even.

Hence $n\ E\ 1$ *[as was to be shown]*.

Example 1 – *Solution*

cont'd

It can be shown that integers *m* and *n* are related by *E* if, and only if, *m mod* 2 = *n mod* 2 (that is, both are even or both are odd).

When this occurs *m* and *n* are said to be **congruent modulo 2**.

The Inverse of a Relation

The Inverse of a Relation

If R is a relation from A to B, then a relation R^{-1} from B to A can be defined by interchanging the elements of all the ordered pairs of R.

• Definition

Let R be a relation from A to B. Define the inverse relation R^{-1} from B to A as follows:

$$R^{-1} = \{(y, x) \in B \times A \mid (x, y) \in R\}.$$

This definition can be written operationally as follows:

For all $x \in A$ and $y \in B$, $\quad (y, x) \in R^{-1} \quad \Leftrightarrow \quad (x, y) \in R.$

Example 4 – *The Inverse of a Finite Relation*

Let $A = \{2, 3, 4\}$ and $B = \{2, 6, 8\}$ and let R be the "divides" relation from A to B: For all $(x, y) \in A \times B$,

$$x \; R \; y \quad \Leftrightarrow \quad x \mid y \qquad x \text{ divides } y.$$

a. State explicitly which ordered pairs are in R and R^{-1}, and draw arrow diagrams for R and R^{-1}.

b. Describe R^{-1} in words.

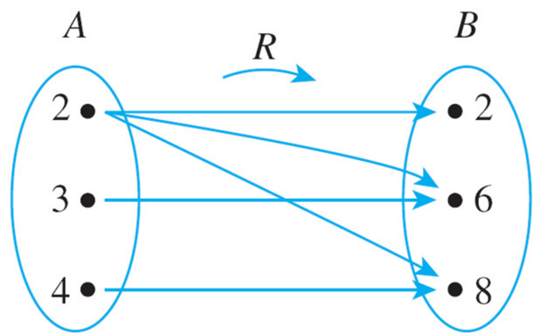
Solution:

a. $R = \{(2, 2), (2, 6), (2, 8), (3, 6), (4, 8)\}$

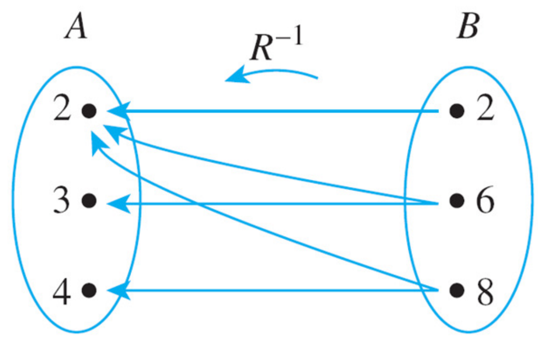
$R^{-1} = \{(2, 2), (6, 2), (8, 2), (6, 3), (8, 4)\}$

Example 4 – *Solution*

cont'd



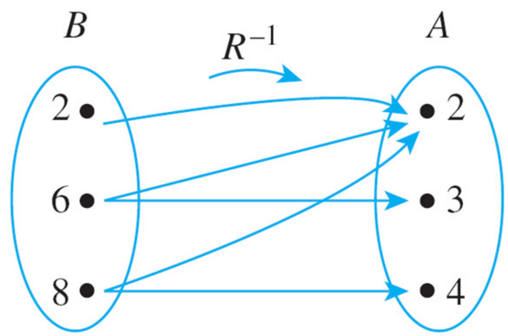
To draw the arrow diagram for R^{-1}, you can copy the arrow diagram for *R* but reverse the directions of the arrows.



Example 4 – *Solution*

cont'd

Or you can redraw the diagram so that *B* is on the left.



b. R^{-1} can be described in words as follows:
For all $(y, x) \in B \times A$,

$$y\ R^{-1}\ x \Leftrightarrow y \text{ is a multiple of } x.$$

Directed Graph of a Relation

Directed Graph of a Relation

• **Definition**

A **relation on a set** ***A*** is a relation from *A* to *A*.

When a relation *R* is defined *on* a set *A*, the arrow diagram of the relation can be modified so that it becomes a **directed graph**.

Instead of representing *A* as two separate sets of points, represent *A* only once, and draw an arrow from each point of *A* to each related point.

Directed Graph of a Relation

As with an ordinary arrow diagram,

For all points x and y in A,

$$\text{there is an arrow from } x \text{ to } y \quad \Leftrightarrow \quad x\ R\ y \quad \Leftrightarrow \quad (x, y) \in R.$$

If a point is related to itself, a loop is drawn that extends out from the point and goes back to it.

Example 6 – *Directed Graph of a Relation*

Let $A = \{3, 4, 5, 6, 7, 8\}$ and define a relation R on A as follows: For all $x, y \in A$,

$$x \; R \; y \Leftrightarrow 2 \mid (x - y).$$

Draw the directed graph of R.

Solution:

Note that $3 \; R \; 3$ because $3 - 3 = 0$ and $2 \mid 0$ since $0 = 2 \cdot 0$. Thus there is a loop from 3 to itself.

Similarly, there is a loop from 4 to itself, from 5 to itself, and so forth, since the difference of each integer with itself is 0, and $2 \mid 0$.

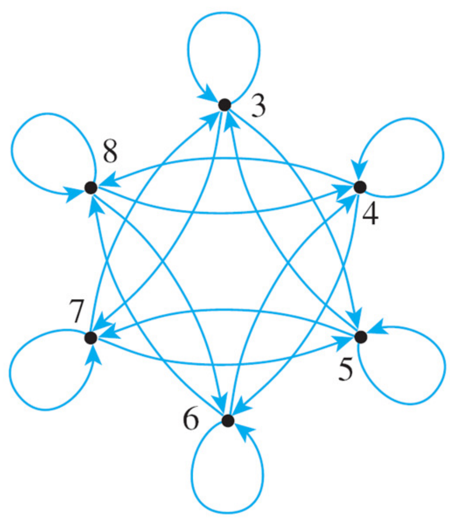
Example 6 – *Solution*

cont'd

Note also that 3 R 5 because 3 – 5 = –2 = 2 · (–1). And 5 R 3 because 5 – 3 = 2 = 2 · 1.

Hence there is an arrow from 3 to 5 and also an arrow from 5 to 3.

The other arrows in the directed graph, as shown below, are obtained by similar reasoning.



N-ary Relations and Relational Databases

N-ary Relations and Relational Databases

N-ary relations form the mathematical foundation for relational database theory.

A binary relation is a subset of a Cartesian product of two sets, similarly, an *n-ary* relation is a subset of a Cartesian product of *n* sets.

- **Definition**

Given sets $A_1, A_2, \ldots, A_n$, an ***n*-ary relation** R on $A_1 \times A_2 \times \cdots \times A_n$ is a subset of $A_1 \times A_2 \times \cdots \times A_n$. The special cases of 2-ary, 3-ary, and 4-ary relations are called **binary, ternary,** and **quaternary relations,** respectively.

Example 7 – *A Simple Database*

The following is a radically simplified version of a database that might be used in a hospital.

Let A_1 be a set of positive integers, A_2 a set of alphabetic character strings, A_3 a set of numeric character strings, and A_4 a set of alphabetic character strings.

Define a quaternary relation R on $A_1 \times A_2 \times A_3 \times A_4$ as follows:

$(a_1, a_2, a_3, a_4) \in R \Leftrightarrow$ a patient with patient ID number a_1, named a_2, was admitted on date a_3, with primary diagnosis a_4.

Example 7 – *A Simple Database* cont'd

At a particular hospital, this relation might contain the following 4-tuples:

(011985, John Schmidt, 020710, asthma)

(574329, Tak Kurosawa, 0114910, pneumonia)

(466581, Mary Lazars, 0103910, appendicitis)

(008352, Joan Kaplan, 112409, gastritis)

(011985, John Schmidt, 021710, pneumonia)

(244388, Sarah Wu, 010310, broken leg)

(778400, Jamal Baskers, 122709, appendicitis)

Example 7 – *A Simple Database* cont'd

In discussions of relational databases, the tuples are normally thought of as being written in tables.

Each row of the table corresponds to one tuple, and the header for each column gives the descriptive attribute for the elements in the column.

Operations within a database allow the data to be manipulated in many different ways.

Example 7 – *A Simple Database* cont’d

For example, in the database language SQL, if the above database is denoted *S*, the result of the query

SELECT Patient_ID#, Name FROM *S* WHERE
Admission_Date = 010310

would be a list of the ID numbers and names of all patients admitted on 01-03-10:

466581 Mary Lazars,
244388 Sarah Wu.

Example 7 – *A Simple Database* cont'd

This is obtained by taking the intersection of the set $A_1 \times A_2 \times \{010310\} \times A_4$ with the database and then projecting onto the first two coordinates.

Similarly, SELECT can be used to obtain a list of all admission dates of a given patient.

For John Schmidt this list is

02-07-10 and

02-17-10

Example 7 – *A Simple Database* cont'd

Individual entries in a database can be added, deleted, or updated, and most databases can sort data entries in various ways.

In addition, entire databases can be merged, and the entries common to two databases can be moved to a new database.