Chapter 7

Relations

The invention of the symbol ≡ by Gauss affords a striking example of the advantage which may be derived from an appropriate notation, and marks an epoch in the development of the science of arithmetic.

— G. B. MATHEWS

Functions are a special case of relations, which are also used in everyday life. Relations have applications to many disciplines, including biology, computer science, psychology, and sociology. The EQUIVALENCE statement in FORTRAN, for example, is based on the relation *has the same location as* (see Example 7.42). Graphs, digraphs, formal languages, finite state machines — all to be discussed in the next four chapters — are closely related to the theory of relations.

In this chapter we will examine the concept of a relation, its computer representations and properties, and different ways to construct new relations from known ones.

We will deal with the following problems, as well as others:

- Is it possible to arrange all n-bit words around a circle in such a way that any two adjacent words differ by exactly one bit?
- Can we determine the day corresponding to a given date $m/d/y$, where $y > 1582$, the year the Gregorian calendar was adopted?
- Five sailors and a monkey are marooned on a desert island. During the day they gather coconuts for food. They decide to divide them up in the morning and retire for the night. While the others sleep, one sailor gets up and divides them into equal piles, with one left over that he throws out for the monkey. He hides his share, puts the remaining coconuts together, and goes back to sleep. Later a second sailor gets up and divides the pile into five equal shares with one coconut left over, which he discards for the monkey. He also hides his share, puts the remaining coconuts together, and goes back to sleep. Later the remaining sailors repeat the process. Find the smallest possible number of coconuts in the original pile.

- The computer science courses required for a computer science major at a college are given in Table 7.1. In which order can a student take them?

Table 7.1

Number	Course	Prerequisite
CS 100	Computer Science I	None
CS 150	Computer Science II	CS 100
CS 200	Computer Organization	CS 150
CS 250	Data Structures	CS 150
CS 300	Computer Architecture	CS 200
CS 350	Programming Languages	CS 250
CS 400	Software Engineering	CS 250
CS 450	Operating Systems	CS 250, CS 300

A special class of matrices called boolean matrices is used to study relations, so we begin with a brief discussion of such matrices.

7.1 Boolean Matrices

(This section is closely related to Section 3.7 on matrices; you will probably find that section useful to review before reading further.)

A **boolean matrix** is a matrix with bits as its entries. Thus $A = (a_{ij})_{m\times n}$ is a boolean matrix if $a_{ij} = 0$ or 1 for every i and j. For instance, $\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 0\end{bmatrix}$ is a boolean matrix, whereas $\begin{bmatrix}1 & -3\\ 0 & 2\end{bmatrix}$ is *not*.

Boolean Operations and and or

The boolean operations *and* ($\wedge$) and *or* ($\vee$), defined by Table 2.1, signal the combining of boolean matrices to construct new ones. Listed below are several properties of these bit operations. They can be verified easily, so try a few.

THEOREM 7.1 Let a and b be arbitrary bits. Then:

- $a \wedge a = a$
- $a \vee a = a$
- $a \wedge (b \wedge c) = (a \wedge b) \wedge c$
- $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
- $a \wedge b = b \wedge a$
- $a \vee b = b \vee a$
- $a \vee (b \vee c) = (a \vee b) \vee c$
- $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ ∎

Using the two bit-operations, we now define two operations on boolean matrices.

Join and Meet

The **join** of the boolean matrices $A = (a_{ij})_{m\times n}$ and $B = (b_{ij})_{m\times n}$, denoted by $A \vee B$, is defined by $A \vee B = (a_{ij} \vee b_{ij})_{m\times n}$. Each element of $A \vee B$ is obtained by *oring* the corresponding elements of A and B. The **meet** of A and B, denoted by $A \wedge B$, is defined by $A \wedge B = (a_{ij} \wedge b_{ij})_{m\times n}$. Every element of $A \wedge B$ is obtained by *anding* the corresponding elements of A and B.

The following example illustrates these two definitions.

EXAMPLE 7.1 Let

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Find $A \vee B$ and $A \wedge B$.

SOLUTION:

- $$A \vee B = \begin{bmatrix} 1\vee 0 & 0\vee 0 & 1\vee 1 \\ 0\vee 1 & 1\vee 0 & 0\vee 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$
- $$A \wedge B = \begin{bmatrix} 1\wedge 0 & 0\wedge 0 & 1\wedge 1 \\ 0\wedge 1 & 1\wedge 0 & 0\wedge 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

■

Boolean Product

The **boolean product** of the boolean matrices $A = (a_{ij})_{m\times p}$ and $B = (b_{jk})_{p\times n}$, denoted by $A \odot B$, is the matrix $C = (c_{ij})_{m\times n}$, where $c_{ij} = (a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \cdots \vee (a_{ip} \wedge b_{pj})$. (See Figure 7.1).

Figure 7.1

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ \cdot & \cdot & \dots & \cdot \\ a_{i1} & a_{i2} & \dots & a_{ip} \\ \cdot & \cdot & \dots & \cdot \\ a_{m1} & a_{m2} & \dots & a_{mp} \end{bmatrix} \odot \begin{bmatrix} b_{11} & \dots & b_{1j} & \dots & b_{1n} \\ \cdot & \dots & \cdot & \dots & \cdot \\ b_{i1} & \dots & b_{ij} & \dots & b_{in} \\ \cdot & \dots & \cdot & \dots & \cdot \\ b_{p1} & \dots & b_{pj} & \dots & b_{pn} \end{bmatrix} = \begin{bmatrix} c_{11} & \dots & c_{1j} & \dots & c_{1n} \\ \cdot & \dots & \cdot & \dots & \cdot \\ c_{i1} & \dots & c_{ij} & \dots & c_{in} \\ \cdot & \dots & \cdot & \dots & \cdot \\ c_{m1} & \dots & c_{mj} & \dots & c_{mn} \end{bmatrix}$$

Notice the similarity between this definition and that of the usual product of matrices.

The next example clarifies this definition.

EXAMPLE 7.2 Let

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Find $A \odot B$ and $B \odot A$, if defined.

SOLUTION:

(1) Since the number of columns in A equals the number of rows in B, $A \odot B$ is defined:

$$A \odot B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} (1 \wedge 1) \vee (0 \wedge 1) \vee (1 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) \\ (0 \wedge 1) \vee (1 \wedge 1) \vee (0 \wedge 0) & (0 \wedge 0) \vee (1 \wedge 1) \vee (0 \wedge 0) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

(2) Number of columns in $B = 2 =$ Number of rows in A. Therefore, $B \odot A$ is also defined:

$$B \odot A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) & (1 \wedge 1) \vee (0 \wedge 0) \\ (1 \wedge 1) \vee (1 \wedge 0) & (1 \wedge 0) \vee (1 \wedge 1) & (1 \wedge 1) \vee (1 \wedge 0) \\ (0 \wedge 1) \vee (0 \wedge 0) & (0 \wedge 0) \vee (0 \wedge 1) & (0 \wedge 1) \vee (0 \wedge 0) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

■

The fundamental properties of the boolean matrix operations are listed in the following theorem. Their proofs being fairly straightforward, appear as routine exercises (see Exercises 36–43).

THEOREM 7.2 Let A, B, and C be three boolean matrices. Then:

- $A \vee A = A$
- $A \wedge A = A$
- $A \vee B = B \vee A$
- $A \wedge B = B \wedge A$
- $A \vee (B \vee C) = (A \vee B) \vee C$
- $A \wedge (B \wedge C) = (A \wedge B) \wedge C$
- $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$
- $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$
- $A \odot (B \odot C) = (A \odot B) \odot C$

The sizes of the matrices are assumed compatible for the corresponding matrix operations. ■

Boolean Power of a Boolean Matrix

Let A be an $m \times m$ boolean matrix and n any positive integer. The nth **boolean power** of A, denoted by $A^{[n]}$, is defined recursively as follows:

$$\begin{aligned} A^{[0]} &= I_m && \text{(the identity matrix)} \\ A^{[n]} &= A^{[n-1]} \odot A && \text{if } n \geq 1 \end{aligned}$$

The following example illustrates this definition.

EXAMPLE 7.3 Let

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Compute $A^{[2]}$ and $A^{[3]}$.

SOLUTION:

$$A^{[2]} = A^{[1]} \odot A = A \odot A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A^{[3]} = A^{[2]} \odot A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

(You can verify that in this case, $A^{[n]} = A$ for every $n \geq 1$.) ■

You will find boolean matrices and their properties useful in the next few sections, so review them as needed.

Exercises 7.1

Using the boolean matrices

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \text{ and } C = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

find each.

1. $A \vee B$ **2.** $A \wedge B$ **3.** $A \odot C$ **4.** $C \odot A$

5. $A \vee (B \vee C)$ **6.** $A \wedge (B \wedge C)$ **7.** $A \odot (B \odot C)$ **8.** $(A \odot B) \odot C$

Using the boolean matrices

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \text{ and } C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

find each.

9. $A \wedge (B \vee C)$ **10.** $A \vee (B \wedge C)$ **11.** $(A \wedge B) \vee (A \wedge C)$

12. $(A \vee B) \wedge (A \vee C)$ **13.** $(A \wedge B) \vee (A \vee C)$ **14.** $A \odot (B \odot C)$

15. $(A \odot B) \odot C$ **16.** $B \odot C \odot A$ **17.** $A \odot A \odot A$

18. Using the boolean matrix

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

find $A^{[3]}$ and $A^{[5]}$.

Let A and B be any two $n \times n$ boolean matrices. Find the number of boolean operations needed to compute each.

19. $A \vee B$ **20.** $A \wedge B$ **21.** $A \odot B$

22. Find the number of $m \times n$ boolean matrices that can be defined.

23. Let A be an $m \times p$ boolean matrix and B a $p \times n$ boolean matrix. Find the number of boolean operations needed to compute $A \odot B$.

24. For the boolean matrix A in Example 7.3, prove that $A^{[n]} = A$ for every $n \geq 1$.

The **complement** of a boolean matrix A, denoted by A', is obtained by taking the one's complement of each element in A, that is, by replacing 0's with 1's and 1's with 0's. Use the boolean matrices A, B, and C in Exercises 1–8 to compute each.

25. A' **26.** B' **27.** $(A \vee B)'$ **28.** $A' \wedge B'$

29. $(A \wedge B)'$ **30.** $A' \vee B'$ **31.** $A \wedge (B' \vee C')$ **32.** $(A \odot B) \odot C'$

Let A and O be two $m \times n$ boolean matrices such that every entry of A is 1 and every entry of O is 0. Let B be any $m \times n$ boolean matrix. What can you say about each?

33. $A \vee B$ **34.** $A \wedge B$ **35.** A'

Let A, B, and C be any $n \times n$ boolean matrices. Prove each.

36. $A \vee A = A$ **37.** $A \wedge A = A$ **38.** $A \vee B = B \vee A$ **39.** $A \wedge B = B \wedge A$

40. $A \vee (B \vee C) = (A \vee B) \vee C$ **41.** $A \wedge (B \wedge C) = (A \wedge B) \wedge C$

42. $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$ **43.** $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$

Write an algorithm to find each.

44. The join of two boolean matrices A and B.

45. The meet of two boolean matrices A and B.

46. The complement of a boolean matrix A.

47. The boolean product of two boolean matrices A and B.

48. The nth boolean power of an $m \times m$ boolean matrix A.

7.2 Relations and Digraphs

Clearly many relationships exist in the world around us. On the human level, they are parent–child, husband–wife, student–teacher, doctor–patient, and so on. Relationships exist between numbers also; the equality relation ($=$) and the less-than relation ($<$) are two such relationships. In fact, relationships can exist between any two sets; they are known as relations.

This section presents the concept of a relation and discusses how relations can be represented using matrices and graphs.

Before formally defining a binary relation, let us study an example.

EXAMPLE 7.4 Consider the sets $A = \{\text{Tom, Dick, Harry}\}$ and $B = \{\text{Amy, Betsy, Carol, Daisy}\}$. Suppose Tom is married to Daisy, Dick to Carol, and Harry to Amy. Let $R = \{(\text{Tom, Daisy}), (\text{Dick, Carol}), (\text{Harry, Amy})\}$. Using the set-builder notation, it can also be defined as

$$R = \{(a, b) \in A \times B \mid a \text{ is married to } b\}$$

Notice that $R \subseteq A \times B$. It is defined using the relation *is married to.* The set R is a **binary relation** from A to B. ∎

More generally, we make the following definition.

Binary Relation

A **binary relation** R from a set A to a set B is a subset of $A \times B$. The **domain** of the relation consists of the first elements in R and the **range** consists of the second elements; they are denoted by **dom(R)** and **range(R)**, respectively. A binary relation from A to itself is a binary relation **on** A.

The following example illustrates these terms.

EXAMPLE 7.5 Let $A = \{2, 3, 5\}$ and $B = \{2, 3, 4, 6, 7\}$. Define a relation R from A to B as follows:

$$R = \{(a, b) \mid a \text{ is a factor of } b\}$$

Then $R = \{(2, 2), (2, 4), (2, 6), (3, 3), (3, 6)\}$, $\text{dom}(R) = \{2, 3\}$, and $\text{range}(R) = \{2, 3, 4, 6\}$. ∎

Let R be a relation from A to B. If $(a,b) \in R$, we say a **is related to** b by the relation R; in symbols, we write aRb. If a is *not* related to b, we write $a \not R b$. For instance, $3 < 5$, but $7 \not< 6$. (Here the relation is $<$.) The next example illustrates this further.

EXAMPLE 7.6 Let A be the set of cities and B the countries in the world. Define a relation R from A to B, using the phrase *is the capital of.* So $R = \{(a,b) \in A \times B \mid a$ is the capital of $b\}$. Then Paris R France, but Toronto $\not R$ Canada. ■

Relations from a finite set to a finite set can be represented by boolean matrices, as defined below.

Adjacency Matrix of a Relation

A relation R from a set $\{a_1, a_2, \ldots a_m\}$ to a set $\{b_1, b_2, \ldots b_n\}$ can be represented by the $m \times n$ boolean matrix $M_R = (m_{ij})$, where

$$m_{ij} = \begin{cases} 1 & \text{if } a_i R b_j \\ 0 & \text{otherwise} \end{cases}$$

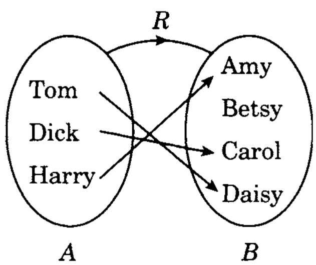
M_R is the **adjacency matrix** of the relation R.

EXAMPLE 7.7 Define a relation R from A = {chicken, dog, cat} to B = {fish, rice, cotton} by $R = \{(a,b) \mid a \text{ eats } b\}$. Then R = {(chicken, fish), (chicken, rice), (dog, fish), (dog, rice), (cat, fish), (cat, rice)}. Its adjacency matrix is

$$M_R = \begin{array}{c} \\ \text{chicken} \\ \text{dog} \\ \text{cat} \end{array} \begin{array}{c} \begin{array}{ccc} \text{fish} & \text{rice} & \text{cotton} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

■

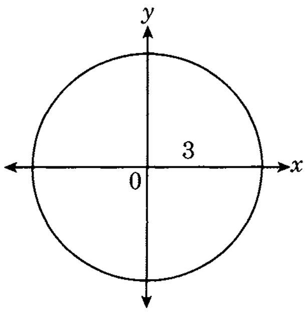
Figure 7.2



Relations can also be represented pictorially. For instance, the relation in Example 7.4 is displayed in Figure 7.2; an arrow from an element a in A to an element b in B indicates that a is related to b.

Figure 7.3

The circle $x^2 + y^2 = 9$.



Relations can be displayed using familiar **graphs** as well. For example, the graph of the relation $\{(x,y) \in \mathbb{R} \times \mathbb{R} \mid x^2 + y^2 = 9\}$ is the circle $x^2 + y^2 = 9$ with center at the origin and radius 3 (see Figure 7.3).

Digraphs

Relations R on a finite set A can be represented pictorially in yet another way. We denote every element of A by a point, called a **vertex** (or **node**), and each ordered pair (a, b) in R by a directed arc or a directed line segment, called an **edge**, from a to b. The resulting diagram is a **directed graph** or simply a **digraph**. If an edge (a, b) exists, we say that vertex b is **adjacent to** vertex a. (Notice the order of the vertices.)

The next two examples illustrate these definitions.

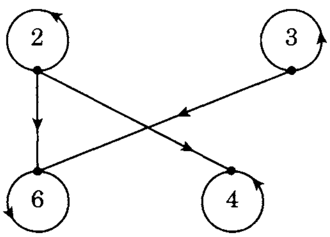
EXAMPLE 7.8 Represent the relation R defined on $A = \{2, 3, 4, 6\}$ by the phrase *is a factor of* in a digraph.

SOLUTION:
Notice that

$$\begin{aligned} R &= \{(a, b) \in A \times A \mid a \text{ is a factor of } b\} \\ &= \{(2,2), (2,4), (2,6), (3,3), (3,6), (4,4), (6,6)\} \end{aligned}$$

Figure 7.4 shows its digraph. It contains four vertices: 2, 3, 4, and 6. Since $3R6$, vertex 6 is adjacent to vertex 3.

Figure 7.4



■

Notice that the digraph in Figure 7.4 contains an edge (x,x) leaving and terminating at the same vertex x. Such an edge is a **loop**. The digraph in Figure 7.4 contains four loops.

We now turn to the concept of a path in a relation, and hence in a digraph.

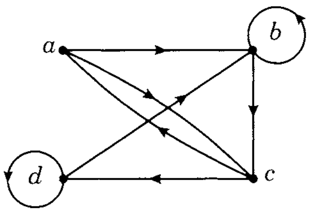
Paths in Digraphs and Relations

Let R be a relation on a set A, and let $a, b \in A$. A **path** in R, that is, in the digraph of R, from a to b is a finite sequence of edges $(a,x_1), (x_1,x_2), \ldots, (x_{n-1},b)$; the vertices x_i's need *not* be distinct. The path from a to b is also denoted by $a\text{-}x_1\text{-}x_2\text{-} \cdots \text{-}x_{n-1}\text{-}b$. The number of edges in the path is its **length**. A path that begins and terminates at the same vertex is a **cycle**. A cycle of length one is a **loop**.

The next example clarifies these terms.

EXAMPLE 7.9 Notice that the relation in Figure 7.5 contains a path of length three from a to b, namely, a-c-d-b. The path b-c-d-b is a cycle of length three. The cycle b-b is a loop.

Figure 7.5



■

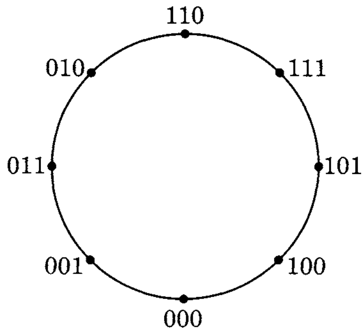
The next example presents an interesting relation in the language of binary words.

EXAMPLE 7.10 (**Gray Codes**) Suppose a switching network is composed of n switches a_i, where $1 \le i \le n$. Let $a_i = 1$ denote that switch a_i is closed and $a_i = 0$ denote that it is open. Every state of the network can be denoted by the n-bit word $a_1a_2 \ldots a_n$. Let Σ^n denote the set of n-bit words, that is, the set of all states of the network. For example, $\Sigma^3 = \{000, 001, 010, 100, 011, 101, 110, 111\}$. Naturally, we are tempted to ask: *Is it possible to test every state of the circuit by changing the state of exactly one switch? That is, is it possible to list every n-bit word by changing exactly one bit?*

Another definition can lead to rewording the question. Two n-bit words are **adjacent** if they differ in exactly one bit, that is, if the **Hamming distance** between them is one. For example, 010 and 011 are adjacent, whereas 001 and 110 are not.

Define a relation R on Σ^n as $\alpha R \beta$ if α and β are adjacent. We can rephrase this: Is it possible to arrange the elements α_i of Σ^n in such a way that $\alpha_i R \alpha_{i+1}$ where $1 \le i \le m - 1, \alpha_m R \alpha_1$, and $m = 2^n$? That is, is it possible to arrange the n-bit words around a circle in such a way that any two neighboring words are adjacent?

Figure 7.6



Interestingly enough, the elements of Σ^3 can be arranged in this fashion (see Figure 7.6): 000, 001, 011, 010, 110, 111, 101, 100. Such an ordering is called a **Gray code** for Σ^3. More generally, a **Gray code** for Σ^n is an arrangement of its elements $\alpha_1, \alpha_2, \ldots, \alpha_m$ such that $\alpha_i R \alpha_{i+1}$ and $\alpha_m R \alpha_1$, where $1 \leq i \leq m - 1$. Gray codes are named for Frank Gray, who invented them in the 1940s at what was then AT&T Bell Labs.

We can restate our original question again: *Is there a Gray code for Σ^n for every $n \geq 1$?* Induction leads to an affirmative answer.

PROOF (by induction):
Let P(n): There exists a Gray code for every Σ^n.

Basis step When $n = 1, \{0, 1\}$ is clearly a Gray code; so P(1) is true.

Induction step Assume P(k) is true; that is, there is a Gray code for Σ^k. Suppose $\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ is a Gray code, where $r = 2^k$.

To show that P(k+1) is true:

Consider the $(k + 1)$-bit words $0\alpha_1, 0\alpha_2, \ldots, 0\alpha_r, 1\alpha_r, 1\alpha_{r-1}, \ldots, 1\alpha_1$. Clearly they form the $2r = 2^{k+1}$ elements of Σ^{k+1}. Call them $\beta_1, \beta_2, \ldots, \beta_{2r}$, respectively, for convenience. Since $\alpha_i R \alpha_{i+1}$ and $\alpha_r R \alpha_1, \beta_j R \beta_{j+1}$ and $\beta_{2r} R \beta_1$, so $\{\beta_1, \beta_2, \ldots, \beta_{2r}\}$ is a Gray code; that is, P$(k + 1)$ is true.

Thus, by induction, a Gray code exists for every Σ^n.

(Notice that the induction step provides a smooth method for constructing a Gray code for Σ^{k+1} from that of Σ^k. This example will be taken a bit further in Chapter 8.) ■

> Finally, we will see how relations and functions are closely related, if we recall that a function $f : A \to B$ is a set of ordered pairs $(a, b) \in A \times B$ such that *every* element a in A is assigned a *unique* element b in B. Consequently, every function can be redefined as a relation, as follows.

An Alternate Definition of a Function

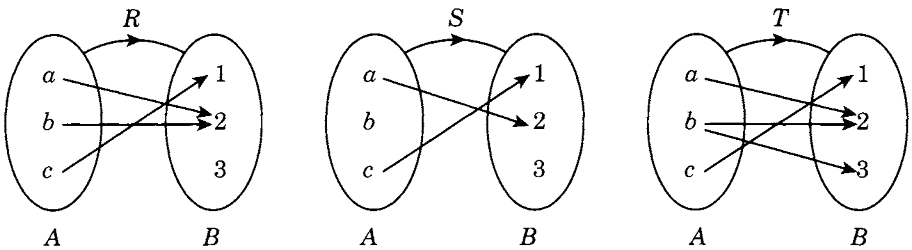
A **function** $f : A \to B$ is a relation from A to B such that:

- $\mathrm{Dom}(f) = A$; and
- If $(a,b) \in f$ and $(a,c) \in f$, then $b = c$.

We close this section with an example that illustrates this definition.

EXAMPLE 7.11 Which of the relations R, S, and T in Figure 7.7 are functions?

Figure 7.7



SOLUTION:
The relation R is a function, whereas S is *not* since $\mathrm{dom}(S) \neq A$. T is also *not* a function since the same element b in A is paired with two distinct elements in B, namely, 2 and 3. ■

Exercises 7.2

List the elements in each relation from $A = \{1,3,5\}$ to $B = \{2,4,8\}$.

1. $\{(a,b) \mid a < b\}$ **2.** $\{(a,b) \mid b = a+1\}$ **3.** $\{(a,b) \mid a+b = 5\}$

4. $\{(a,b) \mid a$ is a factor of $b\}$ **5.** $\{(a,b) \mid a+b \leq 3\}$ **6.** $\{(a,b) \mid a = b\}$

7–12. Find the domain and range of each relation in Exercises 1–6.

13–18. Find the adjacency matrix of each relation in Exercises 1–6.

Represent each relation R on the given set A in a digraph.

19. $\{(a,b) | a < b\}$, $\{2,3,5\}$ **20.** $\{(a,b) | a \leq b\}$, $\{2,3,5\}$

21. $\{(a,b) | a$ is a factor of $b\}$, $\{2,4,5,8\}$

22. $\{(a,b) | b = a+2\}$, $\{2,4,5,6\}$

Using the relation $R = \{(x,y) | 2x + 3y = 12\}$ on $\mathbb{R}$, determine whether or not each is true.

23. $3R2$ **24.** $2R3$ **25.** $-3R5$ **26.** $-5R6$

Using the relation $R = \{(x,y) | x^2 + y^2 = 4\}$ on $\mathbb{R}$, determine if each is true.

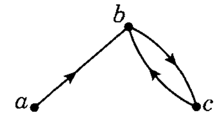
27. $2R0$ **28.** $2R2$ **29.** $-2R0$ **30.** $4R0$

Define a relation R on $\mathbf{Z}$ by xRy if and only if $x - y$ is divisible by 5. Determine if:

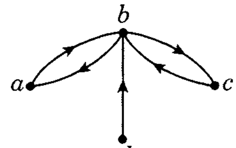
31. $9R4$ **32.** $13R6$ **33.** $3R8$ **34.** $23R3$

List the elements in the relation R represented by each digraph.

35.



36.



37–38. Find the adjacency matrix of each relation in Exercises 35 and 36.

39. Construct a Gray code for Σ^4, where $\Sigma = \{0, 1\}$.

Using the relation in Figure 7.5, find each.

40. Paths of length one starting at a.

41. Paths of length two starting at b.

42. Number of paths of length one.

43. Number of paths of length two.

44. Number of cycles of length three.

45. Number of loops.

Determine if each relation from $\{a, b, c, d\}$ to $\{0, 1, 2, 3, 4\}$ is a function.

46. $\{(a, 0), (b, 1), (c, 0), (d, 3)\}$

47. $\{(a, 3), (b, 3), (b, 4), (c, 1), (d, 0)\}$

48. $\{(a, 3), (b, 3), (c, 3), (d, 3)\}$

49. $\{(a, 1), (b, 2), (c, 3)\}$

Let A and B be finite sets with $|A| = m$ and $|B| = n$. Find the number of binary relations that can be defined:

50. From A to B.

51. On A.

52. A relation R on the set $\{1, 2, \ldots, n\}$ is given in terms of its elements. Write an algorithm to find its adjacency matrix A.

53. Write an algorithm to print the elements of a relation R on $\{1, 2, \ldots, n\}$ using its adjacency matrix A.

*7.3 Computer Representations of Relations (optional)

Since relations from a finite set to a finite set can be represented by boolean matrices, the most straightforward way of implementing a relation and its digraph in a computer is by its adjacency matrix.

The second method involves **linked lists**. Since some programming languages such as FORTRAN do not support dynamic linked lists, the array representation of linked lists serves well. (*Note*: Arrays are nothing but matrices.) For example, the digraph in Figure 7.8 contains seven edges, arbitrarily numbered 1 through 7. Store the tails and the corresponding heads of each edge in two parallel one-dimensional arrays, $T = (t_i)$ and $H = (h_i)$, respectively (see Figure 7.9). Notice that $t_3 = 1$ and $h_3 = 2$, so an edge exists from vertex 1 to vertex 2, namely, edge 1. Since $t_7 = 3$ and $h_7 = 2$, there is also an edge from vertex 3 to vertex 2, namely, edge 5. The other edges can be read similarly.

Figure 7.8

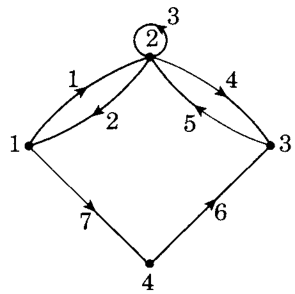


Figure 7.9

	TAIL	HEAD
1	2	2
2	4	3
3	1	2
4	2	3
5	2	1
6	1	4
7	3	2

The enumeration of the edges need not begin with edge 1. In this example, edge 1 is stored in t_3 and h_3. Accordingly, index 3 is stored in a variable called START (see Figure 7.10). Further, the edges can be stored in any order. To find the edge following each edge, an array N (for NEXT) is used. The element n_{i+1} locates the successor of edge n_i, $1 \le i \le 6$. We store 0 in n_6 to indicate the end of the linked list representation of the digraph, as in Figure 7.11.

Figure 7.10

START

3

	T	H
1	2	2
2	4	3
3	1	2
4	2	3
5	2	1
6	1	4
7	3	2

Figure 7.11

Static linked list representation of the relation.

START

3

	T	H	N
1	2	2	4
2	4	3	6
3	1	2	5
4	2	3	7
5	2	1	1
6	1	4	0
7	3	2	2

Most modern programming languages support dynamic data structures. In this type of language, a linked list consists of a set of **nodes** and each node contains (at least) two fields: a **data field** and a **link field** (or **pointer field**) (see Figure 7.12). The data field contains a data item, whereas the link field contains the address of the next node in the list. For instance, consider the linked list in Figure 7.13. HEADER contains the address of the first node in the list; it corresponds to START in the previous discussion. The link field of the last node contains a special pointer called the **nil pointer** that signals the end of the list. This pointer corresponds to 0 in the static representation; a slash (/) in the field signifies it.

Figure 7.12

A typical node.

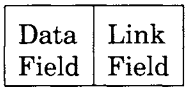
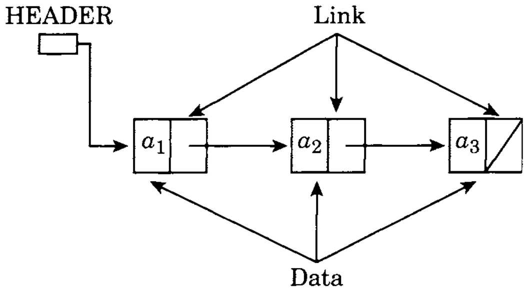
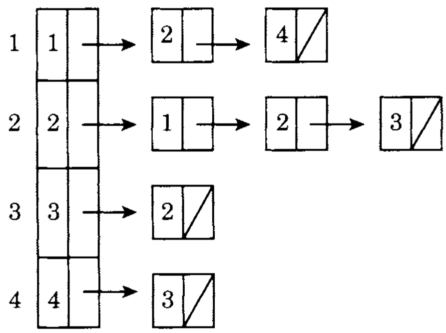


Figure 7.13



The relation in Figure 7.13 illustrates the dynamic linked list representation. First, for each vertex, create a linked list of vertices adjacent to it. Then store the header nodes in an array. The resulting linked representation appears in Figure 7.14.

Figure 7.14



We can abbreviate this representation by storing the header nodes in an array of pointers, as in Figure 7.15. This simplified version is the **adjacency list representation** of the digraph and hence of the relation.

The next example shows how to find the adjacency matrix of a relation from its adjacency list representation.

EXAMPLE 7.12 Using the adjacency list representation of the relation in Figure 7.15, find its adjacency matrix.

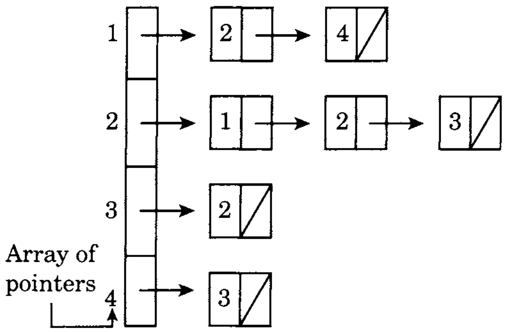
SOLUTION:

The figure indicates vertex 1 is related to 2 and 4; vertex 2 is related to 1, 2, and 3; vertex 3 is related to 2; and vertex 4 is related to 3. Thus, the adjacency matrix of the relation is

$$M_R = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

■

Figure 7.15

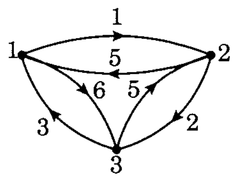


■

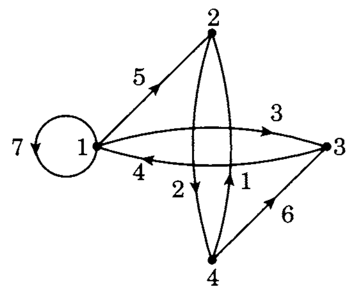
Exercise 7.3

Find the static linked list representation of each relation.

1.



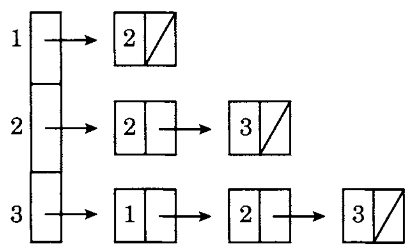
2.



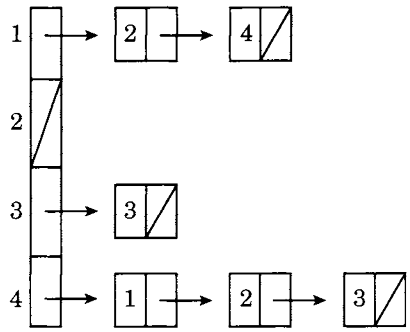
3–4. Find the adjacency list representation of the relations in Exercises 1 and 2.

Find the adjacency matrix of the relation with each adjacency list representation.

5.



6.



7–8. Draw the digraphs of the relations represented by the adjacency lists in Exercises 5 and 6.

Find the adjacency list representation of the relation with the given adjacency matrix.

9. $\begin{array}{c c} & \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{array}$

10. $\begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \end{array}$

Write an algorithm to find the adjacency list representation of a relation R on the set $\{1, 2, \dots, n\}$ using:

11. The relation, given in terms of ordered pairs.

12. Its adjacency matrix A.

13. Write an algorithm to find the adjacency matrix A of a relation on the set $\{1, 2, \dots, n\}$ from its adjacency list representation.

7.4 Properties of Relations

Since relations on finite sets can be represented by matrices, their properties can be identified from their adjacency matrices. In this section we will study the properties of reflexivity, symmetry, antisymmetry, and transitivity.

To begin with, consider the relation R, *is logically equivalent to*, on the set of propositions. Since every proposition is logically equivalent to itself, it has the property that xRx for every proposition x. Such a relation is reflexive.

Reflexive Relation

A relation R on a set A is **reflexive** if xRx for every element x in A, that is, if xRx for every $x \in A$.

Since every set A is a subset of itself, the relation *is a subset of* on its power set is reflexive. Similarly, the **equality relation** (=) is also reflexive; it is denoted by Δ. Thus, a relation is reflexive if and only if $\Delta \subseteq R$.

The next two examples illustrate additional reflexive relations.

EXAMPLE 7.13 Since $x \leq x$ for every real number x, the relation $\leq$ on $\mathbb{R}$ is reflexive. No number is less than itself, so the less than relation is *not* reflexive. ■

EXAMPLE 7.14 Which of the following relations on $A = \{x,y,z\}$ are reflexive?

- $R_1 = \{(x,x),(x,y),(y,y),(z,z)\}$
- $R_2 = \{(x,x),(y,y),(y,z),(z,y)\}$
- $R_3 = \emptyset$ [the **empty relation**]
- $R_4 = \{(x,x),(y,y),(z,z)\}$

SOLUTION:

For a relation R on A to be reflexive, every element in A must be related to itself, that is, $(a,a) \in R$ for every $a \in A$. The element a has three choices, namely, x, y, and z; therefore, the ordered pairs (x,x), (y,y), and (z,z) must be in the relation for it to be reflexive. Consequently, the relations R_1 and R_4 are reflexive, whereas R_2 and R_3 are *not*. ■

> How can we characterize the adjacency matrix $M = (m_{ij})$ of a reflexive relation on the set $A = \{a_1, a_2, \ldots a_n\}$? A relation R on A is reflexive if and only if a_iRa_i for every a_i in A. Thus, R is reflexive if and only if $m_{ii} = 1$ for every ij that is, if and only if the main diagonal elements of M_R are all 1's, as Figure 7.16 shows.

Figure 7.16

$$M_R = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & 1 \end{bmatrix}$$

> The digraph of a reflexive relation must contain a loop at each vertex, since every element of A is related to itself; see Figure 7.16.

Number of Reflexive Relations

We can use the adjacency matrix M_R of a relation R on a set A to compute the number of reflexive relations that can be defined on A, as the following example demonstrates.

EXAMPLE 7.15 Find the number of reflexive relations R that can be defined on a set with n elements.

SOLUTION:
Since R is reflexive, every element on the main diagonal of M_R is 1; there are n such elements. Since M_R contains n^2 elements, there are $n^2 - n = n(n-1)$ elements that do not lie on the main diagonal; each can be a 0 or 1; so each such element m_{ij} has two choices. Thus, by the multiplication principle, we can form $2^{n(n-1)}$ such adjacency matrices, that is, $2^{n(n-1)}$ reflexive relations on A. ■

For an exploration of symmetric and antisymmetric relations, again let R be the relation, *is logically equivalent to*, on the set of propositions. If x and y are any two propositions such that xRy, then yRx. Thus xRy implies yRx.

On the other hand, let x and y be any two real numbers such that $x \leq y$ and $y \leq x$. Then $x = y$. Thus the relation $R(\leq)$ has the property that if xRy and yRx, then $x = y$.

These two examples lead us to the next definitions.

Symmetric and Antisymmetric Relations

A relation R on a set A is **symmetric** if aRb implies bRa; that is, if $(a,b) \in R$, then $(b,a) \in R$. It is **antisymmetric** if aRb and bRa imply $a = b$.

By the law of the contrapositive, the definition of antisymmetry can be restated as follows: A relation R on A is **antisymmetric** if whenever $a \neq b$, either $a \not R b$ or $b \not R a$, that is, $\sim(aRb \wedge bRa)$. Thus R is antisymmetric if there are *no* pairs of distinct elements a and b such that aRb and bRa.

The next three examples demonstrate symmetric and antisymmetric relations.

EXAMPLE 7.16 Which of the following relations on $\{x,y,z\}$ are symmetric? Antisymmetric?

- $R_1 = \{(x,x),(y,y),(z,z)\}$
- $R_2 = \{(x,y)\}$
- $R_3 = \{(x,y),(y,x)\}$
- $R_4 = \{(x,x),(x,z),(z,x),(y,z)\}$

SOLUTION:
The relations R_1 and R_3 are symmetric. R_2 is *not* symmetric, since (y,x) is not in R_2. Similarly, R_4 is *not* symmetric. R_1 and R_2 are antisymmetric, but R_3 and R_4 are *not*. ■

EXAMPLE 7.17 The relation *is logically equivalent to* on the set of propositions is symmetric. Is it antisymmetric? Suppose $p \equiv q$ and $q \equiv p$; this does not imply that $p = q$, so the relation is *not* antisymmetric. ■

EXAMPLE 7.18 The relation $\leq$ on $\mathbb{R}$ is *not* symmetric, since $x \leq y$ does not imply that $y \leq x$. If, however, $x \leq y$ and $y \leq x$, then $x = y$, so the relation is antisymmetric. ■

These two examples demonstrate that a symmetric relation need not be antisymmetric and vice versa.

As for the adjacency matrix of a symmetric relation, a relation R on $\{a_1, a_2, \ldots, a_n\}$ is symmetric only if $a_i \not R a_j$ implies $a_j \not R a_i$; that is, only if, $m_{ij} = m_{ji}$. Thus, R is symmetric if and only if M_R is symmetric; see Figure 7.17.

Figure 7.17

$$M_R = \begin{bmatrix} & 0 & & \\ 0 & & 1 & \\ & 1 & & 0 \\ & & 0 & \end{bmatrix}$$

Graphically, this means if a directed edge runs from a_i to a_j, then one should run from a_j to a_i. In other words, every edge must be bidirectional.

For a relation R to be antisymmetric, if $a_i \neq a_j$ either $a_i \not R a_j$ or $a_j \not R a_i$. In other words, if $i \neq j$ and $m_{ij} = 1$, then $m_{ji} = 0$; that is, either $m_{ij} = 0$ or $m_{ji} = 0$; see Figure 7.18.

Figure 7.18

$$M_R = \begin{bmatrix} & 0 & & \\ 0 & & 0 & \\ & 1 & & 1 \\ & & 0 & \end{bmatrix}$$

Geometrically, if a directed edge runs from a_i to a_j, one should *not* run from a_j to a_i; that is, no edges are bidirectional.

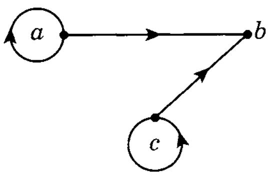
EXAMPLE 7.19 Determine if the relation R on $\{a, b, c\}$ defined by

$$M_R = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

is antisymmetric.

SOLUTION:
Consider the cases $i \neq j$ and $m_{ij} = 1$, where $1 \leq i,j \leq 3$. Clearly, $m_{12} = 1 \neq 0 = m_{21}$ and $m_{32} = 1 \neq 0 = m_{23}$. Thus, when $i \neq j$, either $m_{ij} = 0$ or $m_{ji} = 0$. Therefore, the relation is antisymmetric; see Figure 7.19. (Notice that $m_{11} = m_{33} = 1$ and $m_{22} = 0$, but this does not violate the condition for antisymmetry.)

Figure 7.19



■

Number of Symmetric Relations

Again, the adjacency matrix of a relation on a set A can be effectively used to determine the number of symmetric relations that can be defined on A. The following example demonstrates this.

EXAMPLE 7.20 Find the number of symmetric relations that can be defined on a set with n elements.

SOLUTION:
Let R be a relation on the set and let $M_R = (m_{ij})_{n\times n}$. Then $m_{ij} = 1$ if and only if $m_{ji} = 1$ for every i and j. So each element m_{ij} below the main diagonal determines uniquely the corresponding element m_{ji} above the main diagonal; in other words, each m_{ji} has one choice (see Figure 7.20).

Figure 7.20

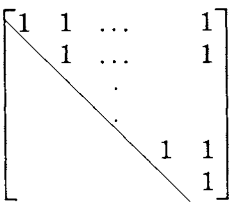
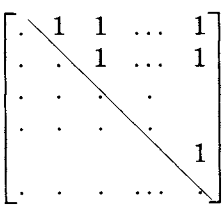


Figure 7.21

Each element has two choices.



Now, each element on or below the main diagonal has two choices: 0 or 1 (see Figure 7.21). There are $1 + 2 + \cdots + n = n(n + 1)/2$ such elements. So, by the multiplication principle, the number of such adjacency matrices equals $2^{n(n+1)/2}$; that is, we can define $2^{n(n+1)/2}$ symmetric relations on the set. ■

Notice that the less-than relation on $\mathbb{R}$ has the property that if $x < y$ and $y < z$, then $x < z$. Accordingly, the order relation $<$ is said to be transitive. More generally, we make the following definition.

Transitive Relation

A relation R on A is **transitive** if aRb and bRc imply aRc; that is, whenever a is related to b and b is related to c, a is related to c.

The next three examples illuminate this definition.

EXAMPLE 7.21 Once again, consider the relation *is logically equivalent to* on the set of propositions. If $p \equiv q$ and $q \equiv r$, then $p \equiv r$, so the relation $\equiv$ is transitive. ■

EXAMPLE 7.22 Let A be the set of courses offered by a mathematics department. Define a relation R on A as follows: xRy if course x is a prerequisite for course y. The relation R is transitive (Why?). (R is the **precedence relation**.) ■

> Determining if a relation R is transitive can be time-consuming, especially if the relation contains many elements. We must look at all possible ordered pairs of the form (a,b) and (b,c), then ascertain if the element (a,c) is also in R, as the next example illustrates.

EXAMPLE 7.23 Which of the following relations on $\{a, b, c\}$ are transitive?

- $R_1 = \{(a,b), (b,c), (a,c)\}$
- $R_2 = \{(a,a), (a,b), (a,c), (b,a), (b,c)\}$
- $R_3 = \{(a,a), (b,b), (c,c)\}$
- $R_4 = \{(a,b)\}$

SOLUTION:
The relation R_1 is transitive; so are R_3 and R_4 by default. In relation R_2, $(b,a) \in R_2$ and $(a,b) \in R_2$, but $(b,b) \notin R_2$. So, R_2 is *not* transitive. ■

> As for the digraph of a transitive relation R, whenever there is a directed edge from a to b and one from b to c, one also runs from a to c.

Transitive relations are explored further in Section 7.7.

Exercises 7.4

Determine if the given relation on $\{a, b, c, d\}$ is reflexive, symmetric, antisymmetric, or transitive.

1. $\{(a,a), (b,b)\}$

2. $\{(a,a), (a,b), (b,b), (c,c), (d,d)\}$

3. $\emptyset$

4. $\{(a,b), (a,c), (b,c)\}$

Is the relation *has the same color hair as* on the set of people:

5. Reflexive? **6.** Symmetric?

7. Antisymmetric? **8.** Transitive?

9–12. Redo Exercises 5–8 using the relation *lives within 5 miles of* on the set of people.

13–16. Let Σ^n denote the set of n-bit words. Define a relation R on Σ^n as xRy if the Hamming distance between x and y is one. Redo Exercises 5–8 using the relation R.

In Exercises 17–19, the adjacency matrices of three relations on $\{a, b, c\}$ are given. Determine if each relation is reflexive, symmetric, or antisymmetric.

17. $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ **18.** $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ **19.** $\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$

When is a relation on a set A *not*:

20. Reflexive? **21.** Symmetric? **22.** Transitive?

Give an example of a relation on $\{a, b, c\}$ that is:

23. Reflexive, symmetric, and transitive.

24. Reflexive, symmetric, but not transitive.

25. Reflexive, transitive, but not symmetric.

26. Symmetric, transitive, but not reflexive.

27. Reflexive, but neither symmetric nor transitive.

28. Symmetric, but neither transitive nor reflexive.

29. Transitive, but neither reflexive nor symmetric.

30. Neither reflexive, symmetric, nor transitive.

31. Symmetric, but not antisymmetric.

32. Antisymmetric, but not symmetric.

33. Symmetric and antisymmetric.

34. Neither symmetric nor antisymmetric.

In Exercise 35–38, complete each adjacency matrix of a relation on $\{a, b, c\}$ in such a way that the relation has the given property.

35. $\begin{bmatrix} - & 1 & 0 \\ 0 & - & 1 \\ 1 & 0 & - \end{bmatrix}$, reflexive **36.** $\begin{bmatrix} 1 & - & 0 \\ 1 & 0 & 1 \\ - & - & 1 \end{bmatrix}$, symmetric

37. $\begin{bmatrix} 0 & - & 1 \\ 1 & 1 & - \\ - & 1 & 0 \end{bmatrix}$, antisymmetric **38.** $\begin{bmatrix} - & 1 & - \\ - & 1 & 1 \\ 1 & - & - \end{bmatrix}$, transitive

39. When will a relation R on a set A be both symmetric and antisymmetric?

A relation R on a set A is **irreflexive** if no element of A is related to itself, that is, if $(a,a) \notin R$ for every $a \in A$. Determine if each relation is irreflexive.

40. The less-than relation on $\mathbb{R}$. **41.** The relation *is a factor of* on $\mathbb{N}$.

42. The relation *is a parent of* on the set of people.

Determine if each relation on $\{a, b, c\}$ is irreflexive.

43. $\{(a, a)\}$ **44.** $\{(a, b), (b, b), (a, c)\}$

45. $\{(b, a), (c, a)\}$ **46.** $\emptyset$

Characterize each for an irreflexive relation on a finite set:

47. Its adjacency matrix. **48.** Its digraph.

A relation R on a set A is **asymmetric** if whenever $aRb, b\not{R}a$. Determine if each relation is asymmetric.

49–51. The relations in Exercises 40–42.

52. $\{(a, a), (b, b), (c, c)\}$ on $\{a, b, c\}$ **53.** $\{(a, b), (a, c), (b, b)\}$ on $\{a, b, c\}$

54. $\{(a, b), (b, c), (c, a)\}$ on $\{a, b, c\}$

For an asymmetric relation on a finite set, characterize:

55. Its adjacency matrix. **56.** Its digraph.

Find the number of binary relations that can be defined on a set of two elements that are:

***57.** Reflexive. ***58.** Symmetric.

***59.** Reflexive and symmetric. ***60.** Antisymmetric.

***61.** Irreflexive. ***62.** Asymmetric.

***63.** Prove: A relation R on a finite set is transitive if $M_R^{[2]} \leq M_R$, where $(a_{ij}) \leq (b_{ij})$ means $a_{ij} \leq b_{ij}$ for every i and j.

7.5 Operations on Relations

Just as sets can be combined to construct new sets, relations can be combined to produce new relations. This section presents five such operations, three of which are analogous to the set operations of union, intersection, and complementation.

Union and Intersection

Let R and S be any two relations from A to B. Their **union** and **intersection**, denoted by $R \cup S$ and $R \cap S$, respectively, are defined as $R \cup S = \{(a,b) \mid aRb \vee aSb\}$ and $R \cap S = \{(a,b) \mid aRb \wedge aSb\}$. Thus $a(R \cup S)b$ if aRb or aSb. Likewise, $a(R \cap S)b$ if aRb and aSb.

The next two examples illustrate these definitions.

EXAMPLE 7.24 Consider the relations $R = \{(a,a), (a,b), (b,c)\}$ and $S = \{(a,a), (a,c), (b,b), (b,c), (c,c)\}$ on $\{a,b,c\}$ (see Figures 7.22 and 7.23). Then $R \cup S = \{(a,a), (a,b), (a,c), (b,b), (b,c), (c,c)\}$ and $R \cap S = \{(a,a), (b,c)\}$.

Figure 7.22

Digraph of R.

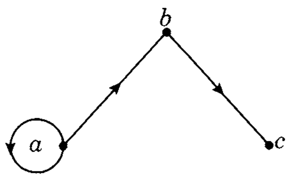


Figure 7.23

Digraph of S.

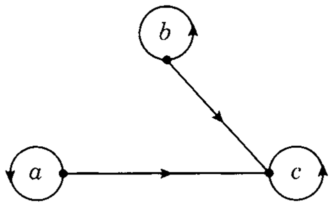


Figure 7.24

Digraph of $R \cup S$.

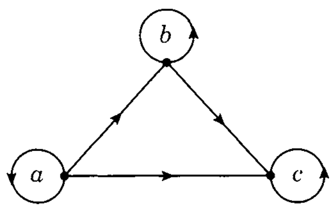
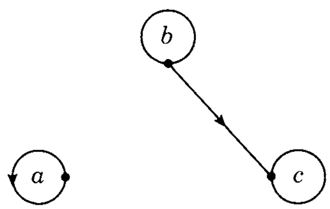


Figure 7.25

Digraph of $R \cap S$.



Graphically, $R \cup S$ consists of all edges in R together with those in S (see Figure 7.24), whereas $R \cap S$ consists of all common edges (see Figure 7.25). ■

EXAMPLE 7.25 Let R and S be the relations $\leq$ and $\geq$ on $\mathbb{R}$, respectively. Then $R \cup S$ consists of all possible ordered pairs $\mathbb{R} \times \mathbb{R}$ and $R \cap S$ is the equality relation. ∎

We can use the adjacency matrices of relations R and S to find those of their union and intersection. By definition, an entry in $M_{R \cup S}$ is 1 if and only if the corresponding element of M_R or M_S is 1; that is, if and only if the corresponding element of their join, $M_R \vee M_S$, is 1. Since $M_{R \cup S}$ and $M_R \vee M_S$ are of the same size, $M_{R \cup S} = M_R \vee M_S$. Similarly, an element of $M_{R \cap S}$ is 1 if and only if the corresponding element of $M_R \wedge M_S$ is 1, so $M_{R \cap S} = M_R \wedge M_S$.

Theorem 7.3 summarizes these conclusions. We leave a formal proof as an exercise (see Exercise 62).

THEOREM 7.3 Let R and S be relations on a finite set. Then $M_{R \cup S} = M_R \vee M_S$ and $M_{R \cap S} = M_R \wedge M_S$. ∎

The following example illustrates this theorem.

EXAMPLE 7.26 Using the adjacency matrices of the relations R and S in Example 7.24 find $M_{R \cup S} = M_R \vee M_S$ and $M_{R \cap S} = M_R \wedge M_S$.

SOLUTION:
We have

$$M_R = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad M_S = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

By Theorem 7.3,

$$M_{R \cup S} = M_R \vee M_S = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad M_{R \cap S} = M_R \wedge M_S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

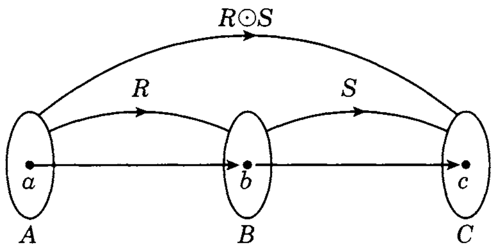
These matrices can recover the actual elements of $R \cup S$ and $R \cap S$ obtained in Example 7.24. ∎

Another way to combine two relations is quite similar to the composition of functions we studied in Section 3.5.

Composition of Relations

Let R be a relation from A to B, and S a relation from B to C. The **composition** of R and S, denoted by $R \odot S$, is defined as follows. Let $a \in A$ and $c \in C$. Then $a(R \odot S)c$ if there exists an element b in B such that aRb and bSc, as in Figure 7.26.

Figure 7.26



The next example illustrates this definition.

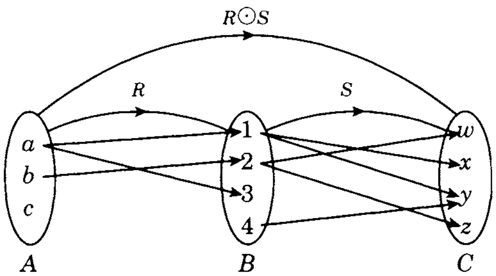
EXAMPLE 7.27 Let $A = \{a,b,c\}, B = \{1,2,3,4\}$, and $C = \{w,x,y,z\}$. Using the relations $R = \{(a,1),(a,3),(b,2)\}$ from A to B and $S = \{(1,x),(1,y),(2,w),(2,z),(4,y)\}$ from B to C (see Figure 7.27), find $R \odot S$.

SOLUTION:

Since $aR1$ and $1Sx$, $a(R \odot S)x$. Similarly, $a(R \odot S)y$, $b(R \odot S)w$, and $b(R \odot S)z$. Thus, $R \odot S = \{(a,x),(a,y),(b,w),(b,z)\}$.

Pictorially, all we need to do is simply follow the arrows from A to C in the figure. (Try this approach.)

Figure 7.27



■

Databases

The next example gives an interesting application of the composition operation to the theory of databases.

EXAMPLE 7.28 Suppose a database consists of two files F_1 and F_2, given by Tables 7.2 and 7.3, respectively. File F_1 can be considered a relation from the set of names to the set of telephone numbers and file F_2 a relation from the set of telephone numbers to the set of telephone bills. Then $F_1 \odot F_2$ is a relation from the set of names to the set of telephone bills. In other words, $F_1 \odot F_2$ is a file of names and their corresponding telephone bills (see Table 7.4).

Table 7.2

Name	Telephone number
Hall	123-4567
Berkowitz	225-5061
Chand	124-3987
Benson	239-3883
Scott	534-3434
Abrams	345-5678

Table 7.3

Telephone number	Bill
123-4567	39.45
123-0011	25.00
243-1111	47.50
124-3987	23.35
124-8958	73.30
534-3434	95.65
345-5678	51.95
128-9876	64.85

Table 7.4

Name	Bill
Hall	39.45
Chand	23.15
Scott	95.65
Abrams	51.95

The adjacency matrices of the relations R, S, and $R \odot S$ display an intriguing connection. To see this, from Example 7.27, we have:

$$M_R = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad M_S = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \text{and}$$

$$M_{R \odot S} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Then

$$M_R \odot M_S = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = M_{R \odot S}$$

■

More generally, we have the following result.

THEOREM 7.4 Let A, B, and C be finite sets. Let R be a relation from A to B, and S a relation from B to C. Then $M_{R \odot S} = M_R \odot M_S$.

PROOF:
Let $A = \{a_1, a_2, \ldots, a_m\}$, $B = \{b_1, b_2, \ldots, b_n\}$, and $C = \{c_1, c_2, \ldots, c_p\}$. Then the matrices $M_R, M_S, M_{R \odot S}$, and $M_R \odot M_S$ are of sizes $m \times n$, $n \times p$, $m \times p$, and $m \times p$, respectively.

Let $M_{R \odot S} = (x_{ij})$ and $M_R \odot M_S = (y_{ij})$. Then $x_{ij} = 1$ if and only if $a_i(R \odot S)c_j$. But $a_i(R \odot S)c_j$ if and only if $a_i R b_k$ and $b_k S c_j$ for some b_k in B. Thus, $x_{ij} = 1$ if and only if $y_{ij} = 1$, so $x_{ij} = y_{ij}$ for every i and j. Consequently, $M_{R \odot S} = M_R \odot M_S$. ■

The definition of composition can be extended to a finite number of relations. Accordingly, we now define the nth power of a relation using recursion.

Recursive Definition of R^n

Let R be a relation on a set A. The ***n*th power of *R***, denoted by R^n, is defined recursively as

$$R^n = \begin{cases} R & \text{if } n = 1 \\ R^{n-1} \odot R & \text{otherwise} \end{cases}$$

Geometrically, R^n consists of the endpoints of all possible paths of length n. Thus $aR^n b$ if a path of length n exists from a to b.

The next two examples illuminate this definition

EXAMPLE 7.29 Using the relation $R = \{(a,b),(b,b),(c,a),(c,c)\}$ on $\{a,b,c\}$, find R^2 and R^3.

SOLUTION:

- $R^2 = R \odot R = \{(a,b),(b,b),(c,a),(c,b),(c,c)\}$
- $R^3 = R^2 \odot R = \{(a,b),(b,b),(c,a),(c,b),(c,c)\} = R^2$

The digraphs of the relations R and R^2 are displayed in Figures 7.28 and 7.29, respectively.

Figure 7.28

Digraph of R.

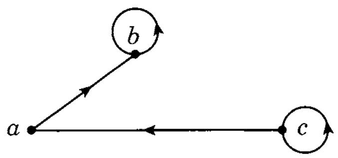
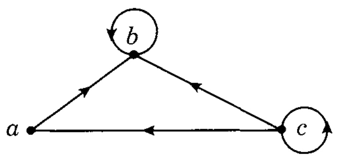


Figure 7.29

Digraph of $R^2 = R^3$.



■

EXAMPLE 7.30 Define a relation R on the set of all U.S. cities as follows: xRy if there is a direct flight from city x to city y. Then xR^2y if there is a direct flight from city x to some city z and a direct flight from city z to city y. Thus R^2 consists of the endpoints of all airline routes in R passing through exactly one city. More generally, R^n consists of the endpoints of all airline routes in R passing through exactly $n - 1$ cities. ■

Let R be a relation on a finite set. Then, by Theorem 7.4, $M_{R \odot R} = M_R \odot M_R$; that is, $M_{R^2} = (M_R)^{[2]}$.

More generally, we have the following result.

THEOREM 7.5 Let R be a relation on a finite set and n any positive integer. Then $M_{R^n} = (M_R)^{[n]}$. ■

EXAMPLE 7.31 For the relation R in Example 7.29, find M_{R^2} and M_{R^3}.

SOLUTION:
Notice that

$$M_R = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

$$M_{R^2} = (M_R)^{[2]} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$M_{R^3} = (M_R)^{[3]} = M_R^{[2]} \odot M_R$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Notice that M_{R^2} and M_{R^3} are the adjacency matrices of the relations R^2 and R^3, obtained in Example 7.29. ■

The next theorem tells us more about powers of transitive relations, and we will use it in Section 7.7.

THEOREM 7.6 Let R be a transitive relation on a set A. Then $R^n \subseteq R$ for every positive integer n.

PROOF (by PMI):
When $n = 1, R^1 \subseteq R$, which is true. Suppose $R^k \subseteq R$ for an arbitrary positive integer k.

To show that $R^{k+1} \subseteq R$:

Let $(x,y) \in R^{k+1}$. Since $R^{k+1} = R^k \odot R, (x,y) \in R^k \odot R$. Then, by definition, there is a z in A such that $(x,z) \in R^k$ and $(z,y) \in R$. But $R^k \subseteq R$, by the inductive hypothesis. Consequently, $(x,z) \in R$. Thus $(x,z) \in R$ and $(z,y) \in R$, so $(x,y) \in R$ by transitivity. Thus $R^{k+1} \subseteq R$.

Thus, by induction, $R^n \subseteq R$ for every $n \geq 1$. ■

We conclude this section with an example to illustrate this theorem.

EXAMPLE 7.32 Notice that the relation $R = \{(a,a),(a,b),(a,c),(b,c)\}$ on $\{a,b,c\}$ (see Figure 7.30) is transitive. You may verify that:

$$R^2 = R \odot R = \{(a,a),(a,b),(a,c)\} \subseteq R$$

$$R^3 = R^2 \odot R = \{(a,a),(a,b),(a,c)\} \subseteq R$$

$$R^4 = R^3 \odot R = \{(a,a),(a,b),(a,c)\} \subseteq R$$

(In fact, $R^n = R^2$ for every integer $n \geq 2$, so $R^n \subseteq R$ for every $n \geq 1$. See Exercise 7.38.)

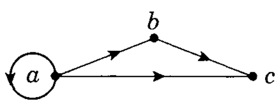


Figure 7.30

Digraph of R.

■

Exercises 7.5

1. Using the relations $R = \{(a,b),(a,c),(b,b),(b,c)\}$ and $S = \{(a,a),(a,b),(b,b),(c,a)\}$ on $\{a,b,c\}$, find $R \cup S$ and $R \cap S$.
2. Redo Exercise 1 using the relations $R = \{(a,a),(a,b),(b,c),(b,d)\}$ and $S = \{(a,b),(b,b),(b,c),(c,a),(d,a)\}$ on $\{a,b,c,d\}$.
3. Let R and S be the relations $<$ and $=$ on $\mathbb{R}$, respectively. Identify $R \cup S$ and $R \cap S$.
4. With the adjacency matrices of the relations R and S in Exercise 1, find those of the relations $R \cup S$ and $R \cap S$.
5. Redo Exercise 4 using the relations in Exercise 2.

Using the relations $R = \{(a,a),(a,b),(b,c),(c,c)\}$ and $S = \{(a,a),(b,b),(b,c),(c,a)\}$ on $\{a,b,c\}$, find each.

6. $R \odot S$
7. $S \odot R$
8. R^2
9. R^3

Let R be a relation from $\{a,b,c\}$ to $\{1,2,3,4\}$ and S a relation from $\{1,2,3,4\}$ to $\{x,y,z\}$. Find $R \odot S$ in each case.

10. $R = \{(a,2),(a,3),(b,1),(c,4)\}$ and $S = \{(1,x),(2,y),(4,y),(3,z)\}$
11. $R = \{(a,1),(b,2),(c,1)\}$ and $S = \{(3,x),(3,y),(4,z)\}$

Using the following adjacency matrices of relations R and S on $\{a,b,c\}$, find the adjacency matrices in Exercises 12–19.

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \qquad M_S = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

12. $M_{R \odot S}$
13. $M_{S \odot R}$
14. M_{R^4}
15. $(M_R)^{[4]}$
16. Define a relation R on the set of U.S. cities as follows: xRy if a direct communication link exists from city x to city y. How would you interpret R^2? R^n?
17. Redo Exercise 16 using the relation R on the set of all countries in the world, defined as follows: xRy if country x can communicate with country y directly.

The **complement** and **inverse** of a relation R from a set A to a set B, denoted by R' and R^{-1} respectively, are defined as follows: $R' = \{(a,b) \mid a \not R b\}$ and $R^{-1} = \{(a,b) \mid bRa\}$. So R' consists of all elements in $A \times B$ that are not in R, whereas R^{-1} consists of all elements (a,b), where $(b,a) \in R$. Using the relations $R = \{(a,a),(a,b),(b,c),(c,c)\}$ and $S = \{(a,a),(b,b),(b,c),(c,a)\}$ on $\{a,b,c\}$, find each.

18. R' **19.** R^{-1} **20.** M_{R^1} **21.** $(M_R)'$

22. $M_{R^{-1}}$ **23.** $(M_R)^T$ **24.** M_{S^1} **25.** $(M_S)^T$

Using the relations $R = \{(a,1),(b,2),(b,3)\}$ and $S = \{(a,2),(b,1),(b,2)\}$ from $\{a,b\}$ to $\{1,2,3\}$, find each.

26. R' **27.** R^{-1} **28.** $(R')'$ **29.** $(R^{-1})^{-1}$

30. $R' \cap S'$ **31.** $(R \cap S)'$ **32.** $R' \cup S'$ **33.** $(R \cup S)'$

34. $(R \cup S)^{-1}$ **35.** $R^{-1} \cup S^{-1}$ **36.** $R^{-1} \cap S^{-1}$ **37.** $(R \cap S)^{-1}$

38. For the relation R in Example 7.32, prove that $R^n = R^2$ for every $n \geq 2$.

Let R and S be relations on a finite set. Prove each.

39. $M_R = (M_{R^1})$ **40.** $M_{R^{-1}} = (M_R)^T$

Let R and S be relations from A to B. Prove each.

41. $(R^{-1})^{-1} = R$ **42.** If $R \subseteq S$, then $S' \subseteq R'$

43. If $R \subseteq S$, then $R^{-1} \subseteq S^{-1}$ **44.** $(R \cup S)' = R' \cap S'$

45. $(R \cap S)' = R' \cup S'$ **46.** $(R \cup S)^{-1} = R^{-1} \cup S^{-1}$

47. $(R \cap S)^{-1} = R^{-1} \cap S^{-1}$

Let R and S be relations on a set. Prove each.

48. R is reflexive if and only if R^{-1} is reflexive.

49. R is symmetric if and only if R' is symmetric.

50. R is symmetric if and only if R^{-1} is symmetric.

51. R is symmetric if and only if $R^{-1} = R$.

52. If R and S are symmetric, $R \cup S$ is symmetric.

53. If R and S are symmetric, $R \cap S$ is symmetric.

54. If R and S are transitive, $R \cap S$ is transitive.

55. Disprove: The union of two transitive relations on a set is transitive.

56. Let A, B, C, and D be any sets, R a relation from A to B, S a relation from B to C, and T a relation from C to D. Prove that $R \odot (S \odot T) = (R \odot S) \odot T$. (**associative property**)

Let R and S be two relations from A to B, where $|A| = m$ and $|B| = n$. Using their adjacency matrices, write an algorithm to find the adjacency matrix of each relation.

57. $R \cup S$ **58.** $R \cap S$ **59.** R' **60.** R^{-1}

61. Let $X = (x_{ij})$ be the adjacency matrix of a relation R from A to B and $Y = (y_{ij})$ that of a relation S from B to C, where $|A| = m$, $|B| = n$, and $|C| = p$. Write an algorithm to find the adjacency matrix $Z = (z_{ij})$ of the relation $R \odot S$.

***62.** Prove Theorem 7.3.

7.6 The Connectivity Relation (optional)

We can use the various powers R^n of a relation R to construct a new relation, called the connectivity relation. This section defines that new relation and then shows how to compute it.

Connectivity Relation

Let R be a relation on a set A. The **connectivity relation** of R, denoted by R^∞, is the union of all powers of R:

$$\begin{aligned} R^\infty &= R \cup R^2 \cup R^3 \cup R^4 \cup \ldots \cup R^n \cup \ldots \\ &= \bigcup_{n=1}^{\infty} R^n \end{aligned}$$

So $M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee \ldots$

> Geometrically, $aR^\infty b$ if there is a path of some length n from a to b. The connectivity relation consists of the endpoints of all possible paths in R.

The next two examples show how to find R^∞.

EXAMPLE 7.33 Find the connectivity relation R^∞ of the relation $R = \{(a,a),(a,b),(a,c),(b,c)\}$ on $\{a,b,c\}$.

SOLUTION:
From Example 7.32, $R^n = R^2$ for every integer $n \geq 2$. So

$$\begin{aligned} R^\infty &= R \cup R^2 \cup R^3 \cup R^4 \cup \ldots \\ &= R \cup R^2 \\ &= \{(a,a),(a,b),(a,c),(b,c)\} \end{aligned}$$

■

EXAMPLE 7.34 Find the connectivity relation R^∞ of the relation $R = \{(a,b),(b,a),(b,b),(c,b)\}$ on $\{a,b,c\}$.

SOLUTION:
$R^2 = R \odot R = \{(a,a),(a,b),(b,a),(b,b),(c,a),(c,b)\}$ (see Figures 7.31 and 7.32.).

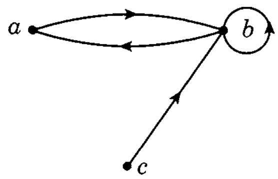


Figure 7.31
Digraph of R.

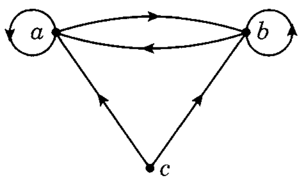


Figure 7.32
Digraph of R^2.

$$R^3 = R^2 \odot R = \{(a,a),(a,b),(b,a),(b,b),(c,a),(c,b)\} = R^2$$

$$R^4 = R^3 \odot R = \{(a,a),(a,b),(b,a),(b,b),(c,a),(c,b)\} = R^2$$

In fact $R^n = R^2$ for every $n \geq 2$. Thus,

$$R^{\infty} = R \cup R^2 = \{(a,a),(a,b),(b,a),(b,b),(c,a),(c,b)\}$$ ■

We can also determine connectivity using the adjacency matrix of a relation.

EXAMPLE 7.35 Using the adjacency matrix of the relation R in Example 7.34, find its connectivity relation.

SOLUTION:
Since $R = \{(a,b),(b,a),(b,b),(c,b)\}$,

$$M_R = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad M_{R^2} = M_R \odot M_R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

$$M_{R^3} = M_{R^2} \odot M_R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \qquad M_{R^4} = M_{R^3} \odot M_R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

Then

$$\begin{aligned} M_{R^\infty} &= M_R \vee M_{R^2} \vee M_{R^3} \vee \ldots \\ &= M_R \vee M_{R^2} \\ &= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \qquad \text{(Verify this.)} \end{aligned}$$

Thus $R^\infty = \{(a,a),(a,b),(b,a),(b,b),(c,a),(c,b)\}$, as in Example 7.34. ■

Theorem 7.7 comes in handy when computing R^∞. With the theorem, only the first n powers of R are needed to compute it, where $n = |A|$.

THEOREM 7.7 Let R be a relation on a set with size n. Then

$$\begin{aligned} R^\infty &= R \cup R^2 \cup R^3 \cup \ldots \cup R^n \\ M_{R^\infty} &= M_R \vee M_{R^2} \vee M_{R^3} \vee \ldots \vee M_{R^n} \\ &= M_R \vee (M_R)^{[2]} \vee (M_R)^{[3]} \vee \ldots \vee (M_R)^{[n]} \end{aligned}$$ ■

The next example illustrates this theorem.

EXAMPLE 7.36 Find R^∞ of the relation R on $\{a,b,c,d\}$ defined by

$$M_R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

SOLUTION:

$$M_{R^2} = M_R \odot M_R = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \qquad M_{R^3} = M_{R^2} \odot M_R = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

$$M_{R^4} = M_{R^3} \odot M_R = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

By Theorem 7.7,

$$M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee M_{R^4} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Thus $R^\infty = \{(a,a),(a,b),(a,c),(a,d),(b,a),(b,b),(b,c),(b,d),(c,a),(c,b),$ $(c,c),(c,d),(d,a),(d,b),(d,c),(d,d)\}$. (You may verify this using the digraph of R.) ■

We can use Theorem 7.7 to develop an algorithm for computing M_{R^∞}, which yields the connectivity relation of a relation R. It is given in Algorithm 7.1.

```
Algorithm Connectivity Relation (MR,MR∞)
(* This algorithm uses the adjacency matrix MR of a relation R on a set
   with size n and computes that of its connectivity relation, using
   Theorem 7.7. *)
0. Begin (* algorithm *)
      (* Initialize MR∞ and B, where B denotes the ith boolean
         power of MR. *)
1.    MR∞ ← MR
      B ← MR
2.    for i = 2 to n do (* find the ith boolean power of MR*)
3.      begin (* for *)
4.        B ← B ⊙ MR
5.        MR∞ ← MR∞ ∨ B (* update MR *)
6.      endfor (* for *)
7. End (* algorithm *)
```

Algorithm 7.1

We close this section with an analysis of the complexity of this algorithm. Let b_n denote the number of boolean operations needed to compute R^∞. Each element in line 4 takes n meets and $n-1$ joins, a total of $2n-1$ operations. Since the product contains n^2 elements, the total number of bit-operations in line 4 is $(2n-1)n^2$. The join of the two $n \times n$ matrices in line 5 takes n^2 boolean operations. Since the ***for*** loop is executed $n-1$ times, the total number of boolean operations is given by

$$\begin{aligned} b_n &= (n-1)[(2n-1)n^2 + n^2] \\ &= 2(n-1)n^3 \\ &= \Theta(n^4) \end{aligned}$$

Thus the connectivity algorithm takes $\Theta(n^4) =$ bit operations.

Exercises 7.6

Find the connectivity relation of each relation on $\{a, b, c\}$.

1. $\{(a,a)\}$ **2.** $\{(a,a),(b,b)\}$ **3.** $\{(a,a),(b,b),(c,c)\}$

4. $\{(a,a),(a,b),(c,a)\}$ **5.** $\emptyset$ **6.** $\{(a,b),(a,c),(b,a),(c,a)\}$

Find the connectivity relation of the relation on $\{a,b,c\}$ with each adjacency matrix.

7. $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ **8.** $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$ **9.** $\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$

Find the connectivity relation of the relation on $\{a,b,c,d\}$ with each adjacency matrix.

10. $\begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ **11.** $\begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$ **12.** $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$

*7.7 Transitive Closure (optional)

The connectivity relation of a relation R is closely associated with its transitive closure. First, we define the closure of R.

A relation R may not have a desired property, such as reflexivity, symmetry, or transitivity. Suppose it is possible to find a relation containing R and having the desired property. The smallest such relation is the **closure** of R with respect to the property. Accordingly, we make the next definition.

Transitive Closure

Suppose a relation R on A is *not* transitive. The smallest transitive relation that contains R is the **transitive closure** of R, denoted by R^*.

> How do we find R^*? If R is not transitive, it should have ordered pairs (a,b) and (b,c) such that $(a,c) \notin R$; so add (a,c) to R. We can continue this with every such pair in the new relation. The resulting relation is transitive, the transitive closure of R.

The next example illustrates this method.

EXAMPLE 7.37 Find the transitive closures of the relations $R = \{(a,b),(b,a),(b,c)\}$, $S = \{(a,a),(b,b),(c,c)\}$, and $T = \emptyset$ on $\{a,b,c\}$.

SOLUTION:

- $R = \{(a,b),(b,c),(b,a)\}$. Since $(a,b) \in R$ and $(b,c) \in R$, it needs (a,c) to be transitive. So add (a,c) to R. The new relation is $R_1 = \{(a,b),(a,c),(b,c),(b,a)\}$. It contains both (a,b) and (b,a), but not (a,a) or (b,b). Add them to R_1: $R_2 = \{(a,a),(a,b),(a,c),(b,a),(b,b),(b,c)\}$. It is transitive and contains R, so it is the transitive closure of R.
- The relation S is transitive, by default, so $S^* = S$.
- The transitive closure of $\emptyset$ is itself. ■

The transitive closure R^* of the relation R in Example 7.37 has practical applications. Suppose the relation indicates the communication links in a network of computers a, b, and c, as in Figure 7.33. The transitive closure R^* shows the possible ways one computer can communicate with another, perhaps through intermediaries. For instance, computer a cannot communicate directly with c, but it can through b. Figure 7.34 displays the transitive closure R^*.

Figure 7.33

Digraph of R.

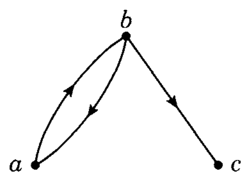
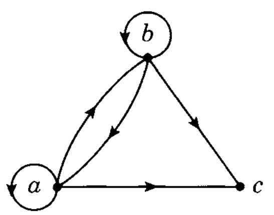


Figure 7.34

Digraph of R.*



The close link between the transitive closure of a relation and its connectivity relation can be illustrated as follows.

THEOREM 7.8 The connectivity relation R^∞ of a relation R is its transitive closure R^*.

PROOF:

The proof unfolds in two parts. First, we must show that R^∞ is transitive and then show it is the smallest transitive relation containing R.

- *To prove that $R^\infty$is transitive*: Let $(a,b) \in R^\infty$ and $(b,c) \in R^\infty$. Since $(a,b) \in R^\infty$, a path runs from a to b. Similarly, one runs from b to c.

Combining these two paths produces a path from a to c. So $(a,c) \in R^{\infty}$ and R^{∞} is transitive.

- *To prove that R^{∞} is the smallest transitive relation containing R:* Suppose there is a transitive relation S such that $R \subseteq S \subseteq R^{\infty}$. We will show that $S = R^{\infty}$. Since S is transitive, by Theorem 7.6, $S^n \subseteq S$ for every $n \geq 1$. So

$$S^{\infty} = \bigcup_{n=1}^{\infty} S^n \subseteq S$$

Thus

$$S^{\infty} \subseteq S$$

By assumption, $R \subseteq S$; so $R^{\infty} \subseteq S^{\infty}$, since every path in R is also a path in S. Therefore, $R^{\infty} \subseteq S$.

Consequently, $S \subseteq R^{\infty}$ and $R^{\infty} \subseteq S$. Therefore, $S = R^{\infty}$. In other words, there are no transitive relations in between R and R^{∞}. So R^{∞} is the smallest transitive relation containing R. ■

It follows by Theorems 7.7 and 7.8 that

$$R^* = R \cup R^2 \cup \cdots \cup R^n$$

and hence

$$M_{R^*} = M_R \vee M_{R^2} \vee \cdots \vee M_{R^n}$$

To illustrate this, using Example 7.36, the transitive closure of the relation $R = \{(a,b),(b,a),(b,c),(c,d),(d,a)\}$ on $\{a,b,c,d\}$ is $R^* = R^{\infty} = \{(a,a),$ $(a,b),(a,c),(a,d),(b,a),(b,b),(b,c),(b,d),(c,a),(c,b),(c,c),(c,d),(d,a),(d,b),$ $(d,c),(d,d)\}$.

Since $R^{\infty} = R^*$, the connectivity relation algorithm can be used to compute M_{R^*}, but it is not efficient, especially when M_{R^*} is fairly large. A better method to find R^* is **Warshall's algorithm**, named in honor of Stephen Warshall, who invented it in 1962.

Warshall's Algorithm

Let a-x_1-x_2-$\cdots$-x_m-b be a path in a relation R on a set $A = \{a_1, a_2, \ldots, a_n\}$. The vertices $x_1, x_2, \ldots, x_m$ are the **interior points** of the path. For instance, vertices c and d are the interior points on the path a-c-d-b of the digraph in Figure 7.5.

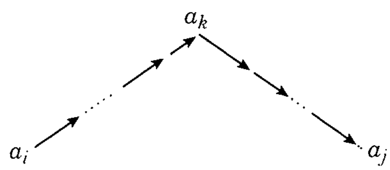
The essence of Warshall's algorithm lies in constructing a sequence of n boolean matrices $W_1, \ldots, W_n$, beginning with $W_0 = M_R$. Let $W_k = (w_{ij})$, where $1 \leq k \leq n$. Define $w_{ij} = 1$ if a path runs from a_i to a_j in R whose interior vertices, if any, belong to the set $\{a_1, a_2, \ldots, a_k\}$. Since the ijth element of W_n equals 1 if and only if a path exists from a_i to a_j whose interior points belong to the set $\{a_1, a_2, \ldots, a_n\}$, $W_n = W_{R^*}$.

In fact, the matrix $W_k = (w_{ij})$ can be constructed from its predecessor $W_{k-1} = (v_{ij})$ as follows. When can $w_{ij} = 1$? For $w_{ij} = 1$, there must be a path from a_i to a_j whose interior vertices belong to the set $\{a_1, a_2, \ldots, a_k\}$.

Case 1 If a_k is *not* an interior vertex, all interior vertices must belong to the set $\{a_1, a_2, \ldots, a_{k-1}\}$, so $v_{ij} = 1$.

Case 2 Suppose a_k is an interior vertex (see Figure 7.35). If a cycle exists at a_k, eliminate it to yield a shorter path. (This guarantees that the vertex a_k occurs exactly once in the path.) Therefore, all interior vertices of the paths a_i-$\cdots$-a_k and a_k-$\cdots$-a_j belong to the set $\{a_1, a_2, \ldots a_{k-1}\}$. In other words, $v_{ik} = 1$ and $v_{kj} = 1$.

Figure 7.35



Consequently, $w_{ij} = 1$ only if $v_{ij} = 1$, or $v_{ik} = 1$ and $v_{kj} = 1$. This is the crux of Warshall's algorithm. Thus the ijth element of W_k is 1 if:

- The corresponding element of W_{k-1} is 1 or
- Both the ikth element and the kjth element of W_{k-1} are 1; that is, the ith element in column k of W_{k-1} and the jth element in row k of W_{k-1} are 1.

Use this property to construct W_1 from $W_0 = M_R$, W_2 from $W_1, \ldots$, and W_n from W_{n-1}. Since $W_n = M_{R^*}$, the actual elements of R^* can be read from W_n.

The next two examples clarify this algorithm.

EXAMPLE 7.38 Using Warshall's algorithm, find the transitive closure of the relation $R = \{(a,b), (b,a), (b,c)\}$ on $A = \{a,b,c\}$.

SOLUTION:

Step 1 Find W_0.

$$W_0 = M_R = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 2 Find W_1.

If the ijth element of W_0 is 1, the ijth element of W_1 is also 1. In other words, every 1 in W_0 stays in W_1. To find the remaining 1's in W_1, locate the 1's in column $1(=k)$; there is just one 1; it occurs in position $i=2$. Now locate the 1's in row $1(=k)$. Again, there is just one 1, namely, in position $j=2$. Therefore, the ijth entry in W_1 should be 1, where $i=2$ and $j=2$. Thus

$$W_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 3 Find W_2.
Again, all the 1's in W_1 stay in W_2. To find the other 1's, if any, locate the 1's in column $2(=k)$ and row $2(=k)$. They occur in positions 1 and 2 of column 2 and in positions 1, 2, and 3 of row 2, so the ijth entry of W_2 must be 1, where $i=1,2$ and $j=1,2,3$. So change the 0's in such locations of W_1 to 1's. Thus

$$W_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 4 Find W_3.
All the 1's in W_2 remain in W_3. To find the remaining 1's, if any, locate the 1's in column 3 — namely, positions 1 and 2 — and the 1's in row 3. Because no 1's appear in row 3, we get no new 1's, so $W_3 = W_2$.

Since A contains three elements, $W_{R^*} = W_3$. Thus,

$$W_{R^*} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

which agrees with the transitive closure obtained in Example 7.37. ■

EXAMPLE 7.39 Using Warshall's algorithm, find the transitive closure of the relation $R=\{(a,a),(a,b),(a,d),(b,a),(c,b),(c,c),(d,b),(d,c),(d,d)\}$ on $\{a,b,c,d\}$.

SOLUTION:
Step 1 Find W_0.

$$W_0 = M_R = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Step 2 Find W_1.
Locate the 1's in column 1 and row 1; positions 1 and 2 in column 1; and positions 1, 2, and 4 in row 1. Therefore, W_1 should contain a 1 in locations (1,1), (1,2), (1,4), (2,1), (2,2), and (2,4):

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

(All the 1's in W_0 remain in W_1.)

Step 3 Find W_2.
Locate the 1's in column 2 and in row 2; positions 1, 2, 3, and 4 in column 2, and positions 1, 2, and 4 in row 2. So W_2 should contain a 1 in locations (1,1), (1,2), (1,4), (2,1), (2,2), (2,4), (3,1), (3,2), (3,4), (4,1), (4,2), and (4,4). Again, since all the 1's in W_1 stay in W_2,

$$W_2 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 4 Find W_3.
The 1's of column 3 occur in positions 3 and 4; those of row 3 in positions 1, 2, 3, and 4. Consequently, W_3 should contain a 1 in locations (i,j) where $i = 3, 4$ and $j = 1, 2, 3, 4$:

$$W_3 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 5 Find W_4.
The 1's of column 4 appear in positions 1, 2, 3, and 4; the 1's of row 4 in positions 1, 2, 3, and 4. So W_4 should contain a 1 in locations (i,j) where $i = 1, 2, 3, 4$ and $j = 1, 2, 3, 4$:

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Since $M_{R^*} = W_4$, this is the adjacency matrix of the transitive closure. (Finding the connectivity relation of R will verify this.) ■

Warshall's algorithm is presented in Algorithm 7.2. It is based on the discussion preceding Example 7.38.

```
Algorithm Warshall(MR,W)
(* This algorithm employs the adjacency matrix of a relation R on
   finite set with n elements to find the adjacency matrix MR* of its
   transitive closure. *)
0. Begin (* algorithm *)
         (* Initialize W = (wij) *)
1.   W ← MR
2.   for k = 1 to n do (* compute Wk *)
3.     for i = 1 to n do
4.       for j = 1 to n do
5.         wij ← wij ∨ (wik ∧ wkj) (*compute the ij-th element *)
6.   MR ← W
7. End (* algorithm *)
```

Algorithm 7.2

A Comparison of Warshall's Algorithm with the Connectivity Algorithm

Why is this algorithm far more efficient than the connectivity relation algorithm? Notice that the number of boolean operations in line 5 is 2, so the total number of boolean operations in lines 2 through 5 (and hence in the algorithm) is $2 \cdot n \cdot n \cdot n = 2n^3 = \Theta(n^3)$, whereas the connectivity algorithm takes $\Theta(n^4)$ bit operations.

Exercises 7.7

Find the transitive closure of each relation on $A = \{a, b, c\}$.

1. $\{(a,b),(b,a)\}$

2. $\{(a,b),(b,c),(c,a)\}$

3. $\{(b,a),(b,c),(c,b)\}$

4. $\{(a,a),(a,c),(b,c),(c,a)\}$

Find the transitive closure of each relation on $A = \{a, b, c, d\}$.

5. $\{(a,a),(a,b)\}$

6. $\{(a,b),(b,c),(c,a)\}$

In Exercises 7–9, find the adjacency matrix of the transitive closure of each relation R on $\{a, b, c\}$ with the given adjacency matrix.

7. $\begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

8. $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$

9. $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$

10–12. Using the connectivity relation algorithm, find the transitive closure R^* of each relation in Exercises 7–9.

13–15. For the relation R on $\{a,b,c\}$ with each adjacency matrix in Exercises 7–9, compute the boolean matrix W_1 in Warshall's algorithm.

In Exercises 16–18, the adjacency matrix of a relation R on $\{a,b,c,d\}$ is given. In each case, compute the boolean matrices W_1 and W_2 in Warshall's algorithm.

16. $\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ **17.** $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ **18.** $\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$

19–24. Using Warshall's algorithm, find the transitive closure of each relation in Exercises 7–9 and 16–18.

25–33. The **reflexive closure** of a relation on a set is the smallest reflexive relation that contains it. Find the reflexive closures of the relations in Exercises 1–9.

Find the reflexive closure of each relation on $\mathbb{R}$.

34. The less-than relation. **35.** The greater-than relation.

36–44. The **symmetric closure** of a relation on a set is the smallest symmetric relation that contains it. Find the symmetric closures of the relations in Exercises 1–9.

Let R be any relation on a set A. Prove each.

45. R is reflexive if and only if $\Delta \subseteq R$.

46. $R \cup \Delta$ is reflexive.

***47.** $R \cup \Delta$ is the smallest reflexive relation containing R.
(*Hint*: Assume there is a reflexive relation S such that $R \subseteq S \subseteq R \cup \Delta$. Prove that $S = R$ or $S = R \cup \Delta$.)

***48.** $R \cup R^{-1}$ is symmetric.
[*Hint*: Consider $(R \cup R^{-1})^{-1}$.]

***49.** $R \cup R^{-1}$ is the smallest symmetric relation that contains R.
(*Hint*: Suppose there is a symmetric relation S such that $R \subseteq S \subseteq R \cup R^{-1}$.)

7.8 Equivalence Relations

Section 7.4 introduced relations that are reflexive, symmetric, and transitive. Naturally we can now ask: Are there relations that simultaneously manifest all three properties? The answer is yes; for instance, the relation *is logically equivalent to* on the set of propositions has all these properties. Such a relation is an equivalence relation.

Equivalence Relation

A relation on a set is an **equivalence relation** if it is reflexive, symmetric, and transitive.

Examples 7.40–7.42 explore equivalence relations.

EXAMPLE 7.40 The relation *has the same color hair as* on the set of people is reflexive, symmetric, and transitive. So it is an equivalence relation. ■

EXAMPLE 7.41 Let Σ denote an alphabet. Define a relation R on Σ^* by xRy if $\|x\| = \|y\|$, where $\|w\|$ denotes the length of the word w. Is R an equivalence relation?

SOLUTION:

- Since every word has the same length as itself, R is reflexive.
- Suppose that xRy. Then $\|x\| = \|y\|$, so $\|y\| = \|x\|$. Consequently, yRx. Thus R is symmetric.
- If $xRy4$ and yRz, then $\|x\| = \|y\|$ and $\|y\| = \|z\|$. Therefore, $\|x\| = \|z\|$ and hence xRy. In other words, R is transitive.

Thus, R is an equivalence relation. ■

EXAMPLE 7.42 (optional) Is the relation *has the same memory location as* on the set of variables in a program an equivalence relation?

SOLUTION:

- Since every variable has the same location as itself, the relation is reflexive.
- If a variable x has the same location as a variable y, then y has the same location as x, so the relation is symmetric.
- Suppose x has the same location as y and y has the same location as z. Then x has the same location as z, so the relation is transitive.

Thus the relation is an equivalence relation. ■

FORTRAN provides an **equivalence statement**, so called since the relation *has the same location as* is an equivalence relation. We can see this in the following FORTRAN statement:

```
EQUIVALENCE (A,B),(C,D,E),(F,G,H)
```

It means the variables A and B share the same memory location; the variables C, D, and E share the same memory location; and so do the variables F, G, and H.

The congruence relation, an important relation in mathematics, is a classic example of an equivalence relation. It is closely related to the equality relation and partitions of a finite set, as will be seen shortly.

Karl Friedrich Gauss *(1777–1855), son of a laborer, was born in Brunswick, Germany. A child prodigy, he detected an error in his father's bookkeeping when he was 3. The Duke of Brunswick, recognizing his remarkable talents, sponsored his education. Gauss received his doctorate in 1799 from the University of Helmstedt. In his doctoral dissertation, he gave the first rigorous proof of the fundamental theorem of algebra, which states, "Every polynomial of degree n (≥ 1) with real coefficients has at least one zero." Newton and Euler, among other brilliant minds, had attempted to prove it, but failed.*

He made significant contributions to algebra, number theory, geometry, analysis, physics, and astronomy. His impressive work Disquisitiones Arithmeticae *of 1801 laid the foundation for modern number theory.*

From 1807 until his death, he was the director of the observatory and professor of mathematics at the University of Göttingen.

Called the "prince of mathematics" by his contemporary mathematicians, Gauss made the famous statement, "Mathematics is the queen of the sciences and the theory of numbers the queen of mathematics."

The congruence symbol $\equiv$ was invented around 1800 by Karl Friedrich Gauss, the greatest mathematician of the 19th century.

Congruence Relation

Let $a, b, m \in \mathbf{Z}$, where $m \geq 2$. Then a is **congruent to b modulo m**, denoted by $a \equiv b(\text{mod } m)$, if $a - b$ is divisible by m. The integer m is the **modulus** of the **congruence relation**. (This definition provides the basis of the **mod** operator we studied in Chapter 3.) If a is *not* congruent to b modulo m, we write $a \not\equiv b$ (mod m).

For example, since $5|(13-3)$, $13 \equiv 3$ (mod 5). Also, $-5 \equiv 3$ (mod 4) since $4|(-5-3)$. But $17 \not\equiv 4$ (mod 6), since $6 \nmid (17-4)$.

The congruence relation has several useful properties, some of which are given below.

THEOREM 7.9 Let $a, b, c, d, m \in \mathbf{Z}$ with $m \geq 2$. Then:

(1) $a \equiv a$ (mod m). (**reflexive property**)
(2) If $a \equiv b$ (mod m), then $b \equiv a$ (mod m). (**symmetric property**)
(3) If $a \equiv b$ (mod m) and $b \equiv c$ (mod m), then $a \equiv c$ (mod m). (**transitive property**)
(4) Let r be the remainder when a is divided by m. Then $a \equiv r(\text{mod } m)$.

PROOF:

We shall prove part 3 and leave the other parts as exercises.

(3) Suppose $a \equiv b$ (mod m) and $b \equiv c$ (mod m). Then $m|(a-b)$ and $m|(b-c)$. Consequently, $a - b = mq_1$ and $b - c = mq_2$ for some

integers q_1 and q_2. Then

$$\begin{aligned} a - c &= (a - b) + (b - c) \\ &= mq_1 + mq_2 \\ &= m(q_1 + q_2) \end{aligned}$$

Therefore, $m|(a - c)$ and $a \equiv c \pmod{m}$. ■

It follows by the theorem that the congruence relation is an equivalence relation.

The Congruence Relation and the Mod Operator

Suppose $a \equiv r \pmod{b}$, where $0 \le r \le b$. Then it can be shown that $r = a \bmod b$. Conversely, if $r = a \bmod b$, then $a \equiv r \pmod{b}$. Thus $a \equiv r \pmod{b}$ if and only if $r = a \bmod b$, where $0 \le r \le b$. See exercises 49 and 50.

For example, $43 \equiv 3 \pmod{5}$ and $0 \le 3 < 5$; clearly, $3 = 43 \bmod 5$. Let us digress briefly to look at an interesting application of congruences*.

Friday-the-13th

Congruences can be employed to find the number of Friday-the-13ths in a given year. Whether or not Friday-the-13th occurs in a given month depends on two factors: the day on which the 13th fell in the previous month and the number of days in the previous month.

Suppose that this is a non–leap year and that we would like to find the number of Friday-the-13ths in this year. Suppose also that we know the day the 13th occurred in December of last year. Let M_i denote each of the months December through November in that order and D_i the number of days in month M_i. The various values of D_i are $31, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31$, and 30, respectively.

We label the days Sunday through Saturday by 0 through 6 respectively; so day 5 is a Friday.

Let $D_i \equiv d_i \pmod{7}$, where $0 \le d_i < 7$. The corresponding values of d_i are $3, 3, 0, 3, 2, 3, 2, 3, 3, 2, 3$, and 2, respectively. Each value of d_i indicates the number of days the day of the 13th in month M_i must be advanced to find the day the 13th falls in month M_{i+1}.

For example, December 13, 2000, was a Wednesday. So January 13, 2001, fell on day $(3 + 3)$ = day 6, which was a Saturday.

*T. Koshy, *Elementary Number Theory with Applications*, Harcourt/Academic Press, Boston, MA, 2002.

Let $t_i \equiv \sum_{j=1}^{i} d_j \pmod 7$, where $1 \leq i \leq 12$. Then t_i represents the total number of days the day of December 13 must be moved forward to determine the day of the thirteenth in month M_i.

For example, $t_3 \equiv d_1 + d_2 + d_3 = 3 + 3 + 0 \equiv 6 \pmod 7$. So, the day of December 13, 2000 (Wednesday) must be advanced by six days to determine the day of March 13, 2001; it is given by day $(3 + 6)$ = day 2 = Tuesday.

Notice that the various values of t_i modulo 7 are 3, 6, 6, 2, 4, 0, 2, 5, 1, 3, 6, and 1, respectively; they include all the least residues modulo 7. Given the day of December 13, they can be used to determine the day of the thirteenth of each month M_i in a non–leap year.

Table 7.5 summarizes the day of the 13th of each month in a non-leap year, corresponding to every choice of the day of December 13 of the previous year. You may verify this. Notice from the table that there can be at most three Friday-the-13ths in a non–leap year.

Table 7.5

Day of the 13th in Each Month in a Non–leap Year.

t_i	**Jan.**	**Feb.**	**March**	**April**	**May**	**June**	**July**	**Aug.**	**Sept.**	**Oct.**	**Nov.**	**Dec.**
Dec. 13	**3**	**6**	**6**	**2**	**4**	**0**	**2**	**5**	**1**	**3**	**6**	**1**
Sun	3	6	6	2	4	0	2	5	1	3	6	1
Mon	4	0	0	3	5	1	3	6	2	4	0	2
Tue	5	1	1	4	6	2	4	0	3	5	1	3
Wed	6	2	2	5	0	3	5	1	4	6	2	4
Thu	0	3	3	6	1	4	6	2	5	0	3	5
Fri	1	4	4	0	2	5	0	3	6	1	4	6
Sat	2	5	5	1	3	6	1	4	0	2	5	0

For a leap year, the various values of d_i are 3, 3, 1, 3, 2, 3, 2, 3, 3, 2, 3, and 2; and the corresponding values of t_i are 3, 6, 0, 3, 5, 1, 3, 6, 2, 4, 0, and 2. Using these, we can construct a similar table for a leap year.

Returning to the congruence relation, we now explore a close relationship between equivalence relations and partitions; but first we make the following definition.

Equivalence Class

Let R be an equivalence relation on a set A and let $a \in A$. The **equivalence class** of a, denoted by $[a]$, is defined as $[a] = \{x \in A \mid xRa\}$. It consists of all elements in A that are linked to a by the relation R. If $x \in [a]$, then x is a **representative** of the class $[a]$.

The next two examples explore equivalence relations.

EXAMPLE 7.43 The relation $R = \{(\mathrm{a},\mathrm{a}), (\mathrm{a},\mathrm{b}), (\mathrm{b},\mathrm{a}), (\mathrm{b},\mathrm{b}), (\mathrm{c},\mathrm{c})\}$ on $A = \{\mathrm{a},\mathrm{b},\mathrm{c}\}$ is an equivalence relation. Find the equivalence class of each element in A.

SOLUTION:

(1) $[a] = \{x \in A \mid xRa\} = \{\mathrm{a}, \mathrm{b}\}$ (2) $[b] = \{x \in A \mid xRb\} = \{\mathrm{a}, \mathrm{b}\} = [a]$ (3) $[c] = \{x \in A \mid xRc\} = \{c\}$

Two distinct equivalence classes exist, $[a]$ and $[c]$. Class $[a]$ has two representatives and class $[c]$ one representative. ■

EXAMPLE 7.44 The relation R on the set of words over the alphabet $\{\mathrm{a}, \mathrm{b}\}$, defined by $x\mathrm{R}y$ if $\|x\| = \|y\|$, is an equivalence relation (see Example 7.41). Infinitely many equivalence classes exist for R, such as $\{\lambda\}$, $\{\mathrm{a}, \mathrm{b}\}$, and $\{\mathrm{aa}, \mathrm{ab}, \mathrm{ba}, \mathrm{bb}\}$. ■

EXAMPLE 7.45 Find all equivalence classes of the congruence relation mod 5 on the set of integers.

SOLUTION:

Let r be the remainder when an integer a is divided by 5. Then $a \equiv r(\text{mod } 5)$. Since the possible values of r, by the division algorithm, are $0, 1, 2, 3$, and 4, there are five distinct equivalence classes:

$$[0] = \{\ldots, -10, -5, 0, 5, 10, \ldots\}$$
$$[1] = \{\ldots, -9, -4, 1, 6, 11, \ldots\}$$
$$[2] = \{\ldots, -8, -3, 2, 7, 12, \ldots\}$$
$$[3] = \{\ldots, -7, -2, 3, 8, 13, \ldots\}$$
$$[4] = \{\ldots, -6, -1, 4, 9, 14, \ldots\}$$
■

These three examples lead us to the following observations:

- Every element belongs to an equivalence class.
- Any two distinct equivalence classes are disjoint.

These results can be stated more formally as follows.

THEOREM 7.10 Let R be an equivalence relation on a set A, with a and b any two elements in A. Then the following properties hold:

(1) $a \in [a]$. (2) $[a] = [b]$ if and only if $a\mathrm{R}b$.
(3) If $[a] \neq [b]$, then $[a] \cap [b] = \emptyset$.

PROOF:

(1) Since R is reflexive, aRa for every $a \in A$, so $a \in [a]$.
(2) Suppose $[a] = [b]$. Since $a \in [a]$ by part (1), $a \in [b]$. Therefore, by definition, aRb.

Conversely, let aRb. *To show that* $[a] \subseteq [b]$:
Let $x \in [a]$. Then xRa. Since xRa and aRb, xRb by transitivity. Therefore, $x \in [b]$ by definition. Thus $[a] \subseteq [b]$.
Similarly, $[b] \subseteq [a]$. Thus, $[a] = [b]$.

(3) We will prove the contrapositive of the given statement: If $[a] \cup [b] \neq \emptyset$, then $[a] = [b]$. Suppose $[a] \cup [b] \neq \emptyset$. Then an element x should be in $[a] \cap [b]$. Then $x \in [a]$ and $x \in [b]$. Since $x \in [a]$, xRa and hence aRx by symmetry. In addition, since $x \in [b]$, xRb. Thus aRx and xRb. Therefore, aRb by transitivity. Thus $[a] = [b]$, by part 2.

This concludes the proof. ■

It follows by Theorem 7.10 that any two equivalence classes are either identical or disjoint, but *not* both.

Notice that Example 7.43 has two disjoint equivalence classes, $[a]$ and $[c]$; their union is the whole set A. Therefore, $\{[a], [c]\}$ is a partition of A. In fact, every equivalence relation on a set induces a partition of the set, as given by the next theorem.

THEOREM 7.11 Let R be an equivalence relation on a set A. Then the set of distinct equivalence classes forms a partition of A. ■

The next four examples illuminate this theorem.

EXAMPLE 7.46 The relation *belongs to the same division as* is an equivalence relation on the set of teams in the American (National) League of major-league baseball. Let x denote a certain team in the American League. Then the class $[x]$ consists of all teams that belong to the same division as x. By Theorem 7.11, the set of teams in the league can be partitioned as {[Yankees], [White Sox], [Mariners]}. ■

EXAMPLE 7.47 By Example 7.41, the relation *has the same length as* on the set of words Σ^* over the alphabet $\Sigma = \{\text{a}, \text{b}\}$ is an equivalence relation. Then the set of equivalence classes formed is $\{[\lambda], [\text{a}], [\text{aa}], [\text{aaa}], \ldots\}$; it is a partition of Σ^*. ■

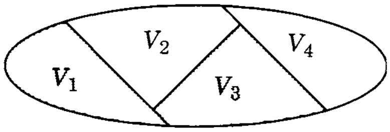
EXAMPLE 7.48 (optional) Suppose a FORTRAN program contains the variables A through J and the equivalence statement:

```
EQUIVALENCE (A,B),(C,D),(F,A,G),(C,J),(E,H)
```

By Example 7.42 the relation *shares the same memory location as* is an equivalence relation on the set of variables V. Let $V_1 = \{\text{A,B,F,G}\}$, $V_2 = \{\text{C,D,J}\}$, $V_3 = \{\text{E,H}\}$, and $V_4 = \{\text{I}\}$. The partition of V induced by this relation is $\{V_1, V_2, V_3, V_4\}$. See Figure 7.36.

Figure 7.36

Set of variables V.

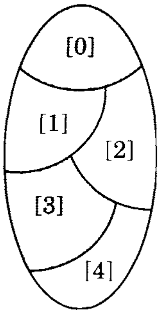


■

EXAMPLE 7.49 By Example 7.45, the distinct equivalence classes formed by the congruence relation modulo 5 on $\mathbf{Z}$ are [0], [1], [2], [3], and [4]. They form a partition of the set of integers, as shown in Figure 7.37.

Figure 7.37

A partition of the set of integers $\mathbf{Z}$.



■

Conversely, does every partition yield an equivalence relation? The next theorem shows that every partition does.

THEOREM 7.12 Every partition of a set induces an equivalence relation on it.

PROOF:

Let $P = \{A_1, A_2, \dots\}$ be a partition of a set A. Define a relation R on A as: aRb if a belongs to the same block as b. We shall show that R is indeed an equivalence relation.

- Since every element in A belongs to the same block as itself, R is reflexive.
- Let aRb. Then a belongs to the same block as b. So b belongs to the same block as a. Thus R is symmetric.
- Let aRb and bRc. Then a belongs to the same block as b and b to the same block as c. So a belongs to the same block as c. Therefore, R is transitive.

Thus R is an equivalence relation. ■

How can we find the equivalence relation corresponding to a partition of a set? The next example demonstrates how to accomplish this.

EXAMPLE 7.50 Find the equivalence relation on $A = \{a, b, c\}$ corresponding to the partition $\{\{a, b\}, \{c\}\}$.

SOLUTION:
Define a relation R on A as follows (see the above proof): xRy if x belongs to the same block as y. Since a and b belong to the same block, aRa, aRb, bRa, and bRb. Similarly, cRc. Thus $R = \{(a,a),(a,b,),(b,a),(b,b),(c,c)\}$. ■

Example 7.50 can serve to develop an algorithm for finding the equivalence relation corresponding to a partition P of a finite set A. It is given in Algorithm 7.3.

```
Algorithm Equivalence Relation (P,A,R)
(* This algorithm determines the equivalence relation R
   corresponding to a partition P of a finite set A. *)
Begin (* algorithm *)
    while P ≠ ∅ do
    begin (* while *)
    extract a block B
    pair each element in B with every element in B
    P ← P − B (* update P *)
    endwhile
  End (* algorithm *)
```

Algorithm 7.3

> Theorems 7.11 and 7.12 indicate a bijection between the family of partitions of a set and the family of equivalence relations on it.

Number of Partitions of a Finite Set

There is a delightful formula for computing the number of partitions (and hence the number of equivalence relations) of a set with size n. It is given by $\sum_{r=1}^{n} S(n,r)$, where $S(n,r)$ denotes a *Stirling number of the second kind*, defined by

$$S(n,1) = 1 = S(n,n)$$
$$S(n,r) = S(n-1,r-1) + rS(n-1,r), 1 < r < n$$

See Exercises 33–40.

Exercises 7.8

Determine if each is an equivalence relation.

1. The relation $\leq$ on $\mathbb{R}$.
2. The relation *is congruent to* on the set of triangles in a plane.
3. The relation *is similar to* on the set of triangles in a plane.

4. The relation *lives within 5 miles of* on the set of people.

5. The relation *takes a course with* on the set of students on campus.

Determine if each is an equivalence relation on $\{a, b, c\}$.

6. $\{(a,a),(b,b),(c,c)\}$

7. $\{(a,a),(a,c),(b,b),(c,a),(c,c)\}$

8. $\emptyset$

9. $\{(a,a),(b,b),(b,c),(c,b)\}$

Using the equivalence relation $\{(a,a),(a,b),(b,a),(b,b),(c,c),(d,d)\}$ on $\{a,b,c,d\}$, find each equivalence class.

10. $[a]$ **11.** $[b]$ **12.** $[c]$ **13.** $[d]$

A FORTRAN program contains 10 variables, A through J, and the following equivalence statement: EQUIVALENCE (A,B,C),(D,E),(F,B),(C,H). Find each class.

14. $[A]$ **15.** $[B]$ **16.** $[E]$ **17.** $[J]$

Using the equivalence relation in Example 7.47, find the equivalence class represented by:

18. a **19.** b **20.** aa **21.** aaa

Using the relation *has the same length as* on the set of words over the alphabet $\{a,b,c\}$, find the equivalence class with each representative.

22. λ **23.** a **24.** ab **25.** bc

26. Find the set of equivalence classes formed by the congruence relation modulo 4 on the set of integers.

Find the partition of the set $\{a,b,c\}$ induced by each equivalence relation.

27. $\{(a,a),(b,b),(c,c)\}$

28. $\{(a,a),(a,c),(b,b),(c,a),(c,c)\}$

A FORTRAN program contains the variables A through J. Find the partition of the set of variables induced by each equivalence statement.

29. EQUIVALENCE (A,B,C),(D,E),(F,B),(C,H)

30. EQUIVALENCE (A,B),(B,J),(C,J),(D,E,H)

Find the equivalence relation corresponding to each partition of the set $\{a,b,c,d\}$.

31. $\{\{a\},\{b,c\},\{d\}\}$

32. $\{\{a,b\},\{c,d\}\}$

The number of partitions of a set with size n is given by $\sum_{r=1}^{n} S(n,r)$, where $S(n,r)$ denotes a Stirling number of the second kind. Compute the number of partitions of a set with the given size.

33. Two **34.** Three **35.** Four **36.** Five

37–40. The number of partitions of a set with size n is also given by the **Bell number** B_n. Using Bell numbers, compute the number of partitions of a set with each of the sizes in Exercises 33–36.

Give a counterexample to disprove each.

41. The union of two equivalence relations is an equivalence relation.

42. The composition of two equivalence relations is an equivalence relation.

We can compute the day of the week corresponding to any date since 1582, the year the Gregorian calendar was adopted. The day d of the week for the rth day of month m in year y (> 1582) is given by

$$d = r + \lfloor 2.6m - 0.2 \rfloor - 2C + D + \lfloor C/4 \rfloor + \lfloor D/4 \rfloor \pmod 7$$

where $C = \lfloor y/100 \rfloor$ and $D = y \bmod 100$; $d = 0$ denotes Sunday; and $m = 1$ denotes March, $m = 11$ January, and $m = 12$ February. This formula is called **Zeller's formula**, after Christian Julius Johannes Zeller (1849–1899). Find the Christmas day of each year.

43. 2000 **44.** 2020 **45.** 2345 **46.** 3000

Let $a, b, c, d, m \in \mathbf{Z}$ with $m \geq 2$. Prove each.

47. If $a \equiv b \pmod m$ and $c \equiv d \pmod m$, then $a + c \equiv b + d \pmod m$.

48. If $a \equiv b \pmod m$ and $c \equiv d \pmod m$, then $ac \equiv bd \pmod m$.

49. Let r be the remainder when a is divided by m. Then $a \equiv r \pmod m$.

50. If $a \equiv r \pmod m$ and $0 \leq r < m$, r is the remainder when a is divided by m.

51. Let r_1 and r_2 be the remainders when a and b are divided by m, respectively. Then $a \equiv b \pmod m$ if and only if $r_1 \equiv r_2 \pmod m$.

52. A positive integer N is divisible by 3 if and only if the sum of its digits is divisible by 3. [*Hint*: $10 \equiv 1 \pmod 3$.]

53. A positive integer N is divisible by 9 if and only if the sum of its digits is divisible by 9. [*Hint*: $10 \equiv 1 \pmod 9$.]

Using the congruence relation, find the remainder when the first integer is divided by the second.

54. 256, 3 **55.** 657, 3 **56.** 1976, 9 **57.** 389, 276, 9

(*Hint*: Use Exercise 52 or 53.)

58. The United Parcel Service assigns to each parcel an identification number of nine digits and a check digit. The check digit is the remainder mod 9 of the 9-digit number. Compute the check digit for 359,876,015.

59. Every bank check has an 8-digit identification number $d_1d_2 \ldots d_8$ followed by a check digit d given by $d \equiv (d_1d_2, \ldots, d_8) \cdot (7, 3, 9, 7, 3, 9, 7, 3) \bmod 10$, where $(x_1, x_2, \ldots, x_n) \cdot (y_1, y_2, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i$. (It is the **dot product** of the two n-tuples.) Compute the check digit for 17,761,976.

60. Libraries use a sophisticated **code-a-bar system** to assign each book a 13-digit identification number $d_1, d_2 \ldots d_{13}$ and a check digit d. Let k denote the number of digits among d_1, d_3, d_5, d_7, d_9, d_{11}, and d_{13} greater than or equal to 5. Then d is computed as $d \equiv [-(d_1, d_2, \ldots, d_{13}) \cdot (2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2) - k] \bmod 10$, where the dot indicates the dot product. Compute the check digit for 2,035,798,008,938.

***61.** (**The coconuts and monkey problem**)* Five sailors and a monkey are marooned on a desert island. During the day they gather coconuts for food. They decide to divide them up in the morning and retire for the night. While the others are asleep, one sailor gets up and divides them into equal piles, with one left over that he throws out for the monkey. He hides his share, puts the remaining coconuts together, and goes back to sleep. Later a second sailor gets up and divides the pile into five equal shares with one coconut left over, which he discards for the monkey. Later the remaining sailors repeat the process. Find the smallest possible number of coconuts in the original pile.

7.9 Partial and Total Orderings

Just as we used the concepts of reflexivity, symmetry, and transitivity to define equivalence relations, we can use reflexivity, antisymmetry, and transitivity to introduce a new class of relations: partial orders. We begin this section with an example.

Building a house can be broken down into several tasks, as Table 7.6 shows. Define a relation R on the set of tasks as follows: Let x and y be any two tasks; then xRy if $x = y$ or must be done before y. This relation is reflexive, antisymmetric, and transitive (verify). Such a relation is a partial order.

*Writer Ben Ames Williams used this problem in a short story titled "Coconuts," which appeared in the October 9, 1926, issue of *The Saturday Evening Post*. The story concerned a contractor who wanted to bid on a large contract. Knowing of their competitor's strong passion for recreational mathematics, one of his employees gave him this problem. The competitor became so obsessed with solving the puzzle that he forgot to enter his bid before the deadline.

Table 7.6

Task	Requires the completion of
(1) Building the foundation (t_1)	None
(2) Framing (t_2)	t_1
(3) Subflooring (t_3)	t_1, t_2
(4) Partitioning into rooms (t_4)	t_1, t_2, t_3
(5) Roofing (t_5)	t_1, t_2
(6) Plumbing (t_6)	t_1, t_2
(7) Wiring (t_7)	t_1, t_2
(8) Siding (t_8)	t_1, t_2, t_5, t_9
(9) Flooring (t_9)	t_1, t_2, t_6
(10) Interior painting (t_{10})	t_1 through t_5, t_7
(11) Exterior painting (t_{11})	t_1, t_2, t_8, t_9
(12) Carpeting (t_{12})	t_1 through t_7, t_9, t_{10}
(13) Installing fixtures (t_{13})	t_1 through t_{11}

Partial Order

A relation R on a set A is a **partial order** if it is reflexive, antisymmetric, and transitive. The set A with its partial order R is a **partially ordered set** (or **poset**), denoted by (A, R). When the partial order is clear from the context, call the poset A.

The next three examples illustrate these definitions.

EXAMPLE 7.51 The relation $\le$ on $\mathbb{R}$ is reflexive, antisymmetric, and transitive, so $\le$ is a partial order on $\mathbb{R}$ and $(\mathbb{R}, \le)$ a poset. Similarly, the divisibility relation $|$ on $\mathbb{N}$ is a partial order, so $(\mathbb{N}, |)$ is also a poset. ■

EXAMPLE 7.52 Let $\Sigma = \{a, b\}$. Define a relation R on Σ^* as: xRy if x is a prefix of y. Is R a partial order?

- Every word is a prefix of itself, so R is reflexive.
- Let xRy and yRx. Then $y = sx$ and $x = ty$ for some $s, t \in \Sigma^*$, so $x = t(sx) = (ts)x$.
  Consequently, $ts = \lambda$ and hence $t = s = \lambda$. So $x = y$ and the relation is antisymmetric.
- Suppose xRy and yRz. Then $y = sx$ and $z = ty$ for some $s, t \in \Sigma^*$. Therefore, $z = t(sx) = (ts)x$. Consequently xRz, and the relation is transitive.

Thus, R is a partial order on Σ^* and (Σ^*, R) is a poset. ■

EXAMPLE 7.53 The relation *has the same color hair as* on the set of people is reflexive, but not antisymmetric. Therefore it is not a partial order. ■

Just as an equivalence relation generalizes the equality relation, a partial order generalizes the relation $\leq$. Accordingly, a partial order is denoted by $\preceq$. $x \preceq y$ means x **precedes or equals** y. If $x \preceq y$ and $x \neq y$, we write $x \prec y$, meaning x **precedes** y.

Comparable Elements

Two elements x and y in a poset are **comparable** if either $x \preceq y$ or $y \preceq x$; otherwise, they are **noncomparable**.

EXAMPLE 7.54 Let x and y be any two real numbers. Then either $x \leq y$ or $y \leq x$. So any two real numbers can be compared using the relation $\leq$: they are comparable.

Using the divisibility relation $|$ on $\mathbb{N}$, the positive integers 3 and 6 are comparable, since $3 \mid 6$. But 3 and 8 are not comparable, since $3 \nmid 8$ and $8 \nmid 3$. ■

> Example 7.54 indicates that a poset may contain noncomparable elements, which justifies the word *partial* in *partial order*. This leads us to the next definition.

Total Order

If any two elements in a poset are comparable, such a partial order is a **total order** or a **linear order**. The poset is then a **totally ordered set** or a **linearly ordered set**.

Notice that $\leq$ is a total order on $\mathbb{R}$, whereas the divisibility relation is *not* a total order on $\mathbb{N}$.

Just as sets can be used to construct new sets, posets can be combined to construct new posets. In order to do this, we first define a relation on the cartesian product of two posets.

Lexicographic Order

Let $(A, \preceq_1)$ and $(B, \preceq_2)$ be two posets. Define a relation $\preceq$ on $A \times B$ as $(a, b) \preceq (a', b')$ if $a \prec_1 a'$, or $a = a'$ and $b \preceq_2 b'$. The relation $\preceq$, an extension of the alphabetic order, is the **lexicographic order**.

The lexicographic order is a partial order on $A \times B$. If A and B are totally ordered sets, so is $A \times B$. The lexicographic order can be extended to the cartesian product $A_1 \times A_2 \times \cdots \times A_n$ of n posets and n totally ordered sets. The next two examples illustrate this.

EXAMPLE 7.55 Consider the cartesian product $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$, where the partial order is the usual $\leq$. Then $(2,5,3) \leq (3,2,1)$ since the first element in the triplet $(2,5,3)$ is less than that in the second triplet $(3,2,1)$. Also, $(2,4,5) \leq (2,4,7)$. This ordering mirrors the familiar sequencing of three-digit numbers. ■

EXAMPLE 7.56 Let Σ be a partially ordered alphabet with the partial order $\preceq$ and Σ^n denote the set of words of length n over Σ. Since every word in Σ^n can be considered an n-tuple, the lexicographic order on the cartesian product on n posets can be applied to Σ^n also.

Let $x = a_1a_2 \dots a_n$ and $y = b_1b_2 \dots b_n$ be any two elements in Σ^n. Then $x \prec y$ if:

- Either $a_1 \preceq b_1$, or
- An integer i exists such that $a_1 = b_1, a_2 = b_2, \dots, a_i = b_i$, and $a_{i+1} \prec b_{i+1}$.

In particular, let Σ denote the English alphabet, a totally ordered set: $a \prec b \prec c \prec \cdots \prec z$. Clearly, *computer* $\prec$ *demolish*, *compress* $\prec$ *computer*, *contend* $\prec$ *content*, and *content* $\prec$ *context*.

This lexicographic order can work for Σ^* in a familiar way. Let x and y be any two words over Σ. Then $x \prec y$ in lexicographic order if one of two conditions holds:

- $x = \lambda$, the empty word.
- If $x = su$ and $y = sv$, where s denotes the longest common prefix of x and y, the first symbol in u precedes that in v in alphabetic order.

For example, *marathon* $\prec$ *marble*, *margin* $\prec$ *market*, *limber* $\prec$ *timber*, and *creation* $\prec$ *discretion*. ■

Hasse Diagrams

We can simplify the digraph of a finite poset by omitting many of its edges. For instance, since a partial order is reflexive, each vertex has a loop, which we can delete. In addition, drop all edges implied by transitivity. For example, if the digraph contains the edges (a, b) and (b, c), it has the edge (a, c), which we can omit. Finally, draw the remaining edges upward and drop all arrows. The resulting is the **Hasse diagram**, named for the German mathematician Helmut Hasse.

Examples 7.57–7.60 generate Hasse diagrams.

EXAMPLE 7.57 Construct the Hasse diagram for the poset $(A, |)$, where $A = \{1, 2, 3, 6, 8, 12\}$ and $|$ denotes the divisibility relation.

SOLUTION:

The digraph of the poset is Figure 7.38.

Step 1 Delete the loop at each vertex. The result is Figure 7.39.

Step 2 Delete all edges implied by transitivity. Figure 7.40 shows the ensuing diagram.

Step 3 Omit all arrows and draw the edges "upward." The Hasse diagram appears in Figure 7.41.

Helmut Hasse *(1898–1979), a celebrated number theorist and dedicated teacher, was born in Kassel, Germany. His father was a judge. While studying at the gymnasiums in Kassel and later Berlin, he decided on a career in mathematics. After the gymnasiums, he entered the navy. While in the navy in the Baltic he studied number theory and then mathematics at the University of Kiel. Leaving the navy in December 1918, Hasse went to Göttingen to pursue his mathematical interest and then to Marburg, receiving his Ph.D. in 1921.*

His teaching career began in Kiel in 1922. Three years later, he became a professor at Halle, then moved to Marburg, Göttingen, Berlin, and finally Hamburg in 1950, where he remained until his retirement in 1966. Earlier he had been director of the Mathematics Institute at Göttingen. But he was dismissed by the British occupation authorities in September 1945.

Hasse was a member of several academies of science and author of numerous articles and books. Hasse received a number of awards including the German National prize for Science and Technology (1953) and the Cothenius Medal of the Academia Leopoldina (1968).

Figure 7.38

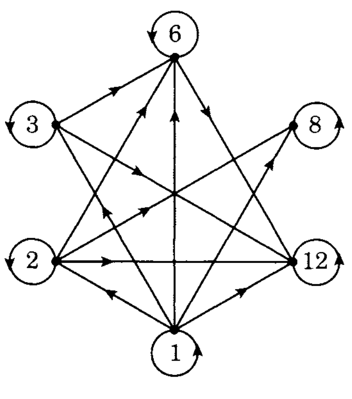


Figure 7.39

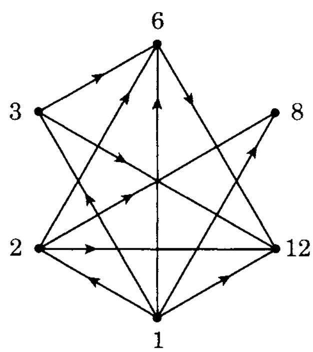


Figure 7.40

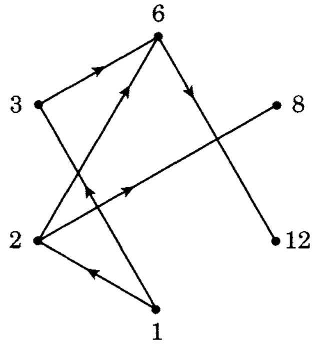
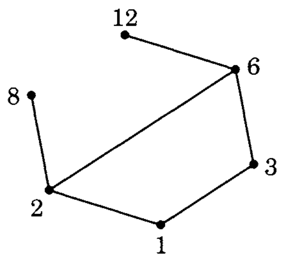


Figure 7.41

Hasse diagram for the poset.

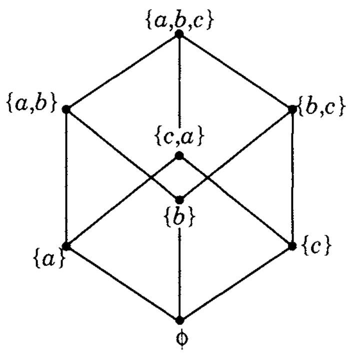


■

EXAMPLE 7.58 Draw the Hasse diagram for the poset $(A, \subseteq)$, where A denotes the power set of the set $\{a, b, c\}$.

SOLUTION:
The set $\{a, b, c\}$ has eight subsets: $\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{b, c\}, \{c, a\}$, and $\{a, b, c\}$. Following steps 1–3, as in Example 7.57, produces the Hasse diagram in Figure 7.42.

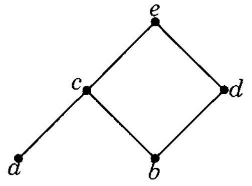
Figure 7.42



■

EXAMPLE 7.59 The relation $R = \{(a, a), (a, c), (a, e), (b, b), (b, c), (b, d), (b, e), (c, c), (c, e), (d, d), (d, e), (e, e)\}$ is a partial order on $\{a, b, c, d, e\}$. Figure 7.43 displays its Hasse diagram.

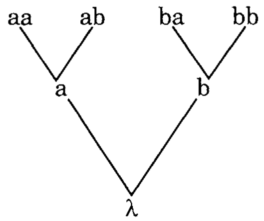
Figure 7.43



■

EXAMPLE 7.60 Consider the alphabet $\Sigma = \{a, b\}$. The relation $\preceq$ on Σ^*, defined by $x \preceq y$ if x is a prefix of y, is a partial order. The Hasse diagram for all words of maximum length two appears in Figure 7.44.

Figure 7.44



■

Extremal elements in a poset are important, especially in linear ordering.

Extremal Elements

An element a in a poset $(A, \preceq)$ is a **maximal element** if A has *no* element b such that $a \prec b$. Similarly, an element a in A is a **minimal element** if A has no element $b \prec a$.

The maximal and minimal elements in a finite poset can easily be read from its Hasse diagram, like the ones in Figures 7.41 and 7.42.

EXAMPLE 7.61 The poset in Figure 7.41 has two maximal elements, 8 and 12, and one minimal element, 1.

Figure 7.43 has one maximal element, e; it has two minimal elements, a and b. ■

A poset may exhibit the following properties:

- A poset may have more than one maximal element and more than one minimal element (see Example 7.61).
- A poset need not have any maximal or minimal elements. For instance, the poset $(\mathbf{Z}, \leq)$ has no maximal or minimal elements.
- A poset may have a maximal element but no minimal elements, or a minimal element but no maximal elements. For example, the poset $(\mathbf{Z}^-, \leq)$ has a maximal element but no minimal elements, whereas the poset $(\mathbf{Z}^+, \leq)$ has a minimal element but no maximal elements.

Two special extremal elements are the greatest and the least.

Greatest and Least Elements

If a poset A contains an element a such that $b \preceq a$ for every element b in A, a is the **greatest element** of the poset. If it contains an element a such that $a \preceq b$ for every b in A, a is the **least element**.

> The greatest element of a poset, if it exists, is unique; likewise, the least element. They are the topmost and the bottommost elements in the Hasse diagram.

For example, the poset in Figure 7.41 has no greatest element, but has a least element, 1. Figure 7.43, on the other hand, has a greatest element, e, but no least element.

Although an arbitrary poset need not have a minimal element, every nonempty finite poset has a minimal element, as Theorem 7.13 shows.

THEOREM 7.13 Every finite nonempty poset $(A, \preceq)$ has a minimal element.

PROOF:
Let a_1 be any element in A. If a_1 is not minimal, there must be an element a_2 in A such that $a_2 \prec a_1$. If a_2 is minimal, then we have finished. If a_2 is not minimal, A must have an element a_3 such that $a_3 \prec a_2$. If a_3 is not minimal, continue this procedure. Since A contains only a finite number of elements, it must terminate with some element a_n. Thus $a_n \prec a_{n-1} \prec \cdots \prec a_3 \prec a_2 \prec a_1$. Consequently, a_n is a minimal element. ■

This result forms the cornerstone of the topological sorting technique.

Topological Sorting

Study the tasks t_1 through t_{13} for building a house, given in Table 7.6. (Recall that the relation *precedes or is the same as* is a partial order on A). For these tasks to be entered in a computer, the elements of the poset must be arranged in a linear order consistent with the partial order. If $a \preceq b$, then enter task a before task b in linear order. This technique is called **topological sorting**.

To topologically sort a finite nonempty poset $(A, \preceq)$ with n elements, proceed as follows. By Theorem 7.13, the poset contains a minimal element, say, a_1. Exclude it from A. Then $A - \{a_1\}$ is also a finite poset. If it is nonempty, it contains a minimal element a_2. Delete a_2 from $A - \{a_1\}$. Then $A - \{a_1, a_2\}$ is a finite poset with minimal element a_3. Continue this procedure until the poset becomes null. This procedure yields the desired linear order, $a_1 \prec a_2 \prec a_3 \prec \cdots \prec a_n$.

A simple algorithm can handle this organizing (see Algorithm 7.4).

```
Algorithm Topological Sort (S)
(* This algorithm sorts a finite nonempty poset S into a linear order
   using topological sorting. *)
  Begin (* algorithm *)
    while S ≠ ∅ do
      begin (* while *)
        find a minimal element a in S
        S ← S − {a} (* delete a from S *)
    endwhile
  End (* algorithm *)
```

Algorithm 7.4

We can establish the validity of this algorithm using induction and Theorem 7.13. We leave its verification as an exercise.

EXAMPLE 7.62 Topologically sort the elements of the poset in Example 7.57.

SOLUTION:

The poset given by the Hasse diagram in Figure 7.41 has one minimal element, 1. Delete it from the poset and hence from the Hasse diagram. The diagram turns into Figure 7.45 with a poset of two minimal elements, 2 and 3. Delete one of them, say, 3. The resulting poset appears in Figure 7.46; it has two minimal elements, 2 and 6. Delete one of them, say, 2. The new poset in Figure 7.47 also has two minimal elements, 6 and 8. Extract, say, 8. The resulting poset is shown in Figure 7.48. Extract its minimal element, 6; this leaves just one element, 12 (see Figure 7.49). Deleting it yields the empty set, and the procedure terminates. Thus, we can sort the elements of the poset in a linear order compatible with the partial order: $1 \prec 3 \prec 2 \prec 8 \prec 6 \prec 12$.

Figure 7.45

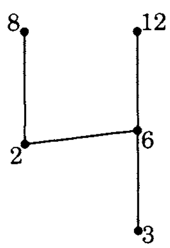


Figure 7.46

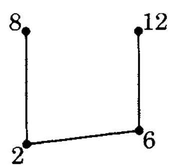


Figure 7.47

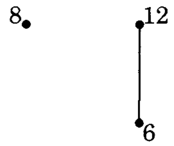


Figure 7.48

Figure 7.49

> In this example, we could have chosen a minimal element in more than one way on three occasions; in other words, the output from the topological sorting need not be unique.

We close this section with another sorting example.

EXAMPLE 7.63 Topologically sort the elements of the poset in Example 7.59.

SOLUTION:
Figures 7.50–7.54 track the steps of the sorting algorithm. The resulting output is $a \prec b \prec d \prec c \prec e$.

Figure 7.50

Extract a.

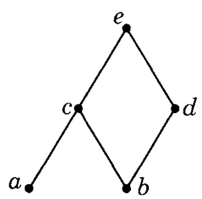


Figure 7.51

Extract b.

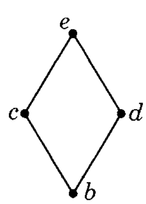


Figure 7.52

Extract c.

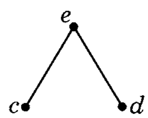


Figure 7.53

Extract d.

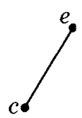


Figure 7.54

Extract e.

■

Exercises 7.9

Determine if each is a partial order.

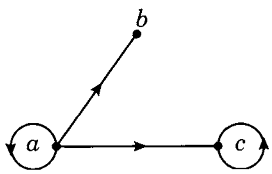
1. The relation $<$ on $\mathbb{R}$
2. The relation $\geq$ on $\mathbb{R}$
3. The relation $\geq$ on $\mathbf{Z}$
4. The relation $\mid$ on $\mathbf{Z}$

Determine if each is a partial order on $\{a, b, c\}$.

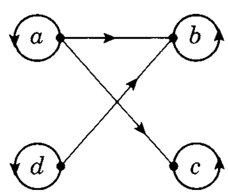
5. $\{(a,a),(b,b),(c,c)\}$
6. $\{(a,a),(a,b),(b,a),(b,b),(c,c)\}$
7. $\{(a,a),(b,b),(b,c),(c,c)\}$
8. $\{(a,a),(a,b),(b,b),(b,c),(c,c)\}$

Determine if each is a partial order.

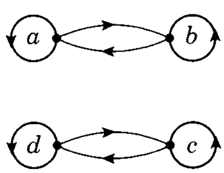
9.



10.



11.



Determine if the given elements are comparable in the poset $(A, \mid)$, where $A = \{1, 2, 3, 6, 9, 18\}$ and $\mid$ denotes the divisibility relation.

12. $2, 3$
13. $2, 6$
14. $2, 9$
15. $3, 18$

Determine if the given elements are comparable in the poset $(A, \subseteq)$, where A denotes the power set of $\{a, b, c\}$ (see Example 7.58).

16. $\{a, b\}, \{b, c\}$

17. $\{a, b\}, \{b\}$

Arrange the following pairs from the poset $\mathbb{N} \times \mathbb{N}$ in lexicographic order.

18. $(3, 5), (2, 3)$

19. $(3, 5), (2, 6)$

20. Find three ordered pairs of positive integers that precede the pair $(2, 3)$ in lexicographic order.

21. Find three triplets of positive integers that precede the triplet $(2, 3, 5)$.

Arrange the following words over the English alphabet in lexicographic order.

22. *mat, rat, bat, cat, eat, fat*

23. *neighbor, neophyte, neglect, moment, luxury, maximum*

24. *custom, custody, custard, cushion, curtain, culvert*

25. *discreet, discrete, discount, discourse, diskette, discretion*

26. Arrange all words of length ≤ 2 over the alphabet $\{a, b\}$ in lexicographic order. Construct a Hasse diagram for each poset.

27. $(A, |)$, where $A = \{1, 2, 3, 6, 9, 18\}$ and $|$ denotes the divisibility relation.

28. $(A, |)$, where $A = \{1, 2, 3, 6, 8, 24\}$ and $|$ is the divisibility relation.

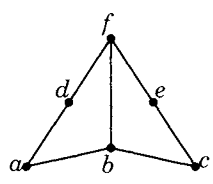
29. (A, R), where $A = \{a, b, c\}$ and $R = \{(a, a), (a, b), (b, b), (b, c), (c, c)\}$.

30. $(A, \subseteq)$, where A denotes the power set of the set $\{a, b\}$.

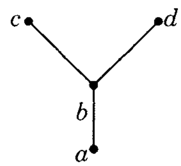
31. Let A denote the set of words of length ≤ 3 over the binary alphabet. The relation R, defined on A by xRy if x is a prefix of y, is a partial order. Draw a Hasse diagram for the poset (A, R).

Find the maximal and minimal elements in the poset with each Hasse diagram.

32.



33.



34.

Find the maximal and minimal elements, if they exist, in each poset.

35. $(A, \leq)$, where A denotes the set of positive even integers.

36. $(A, \leq)$, where A denotes the set of negative even integers.

37. $(A, |)$, where $A = \{1, 2, 3, 6, 9, 18\}$ **38.** $(A, |)$, where $A = \{1, 2, 3, 6, 8, 24\}$

39–42. Find the greatest and least elements, if they exist, in the posets of Exercises 35–38.

Mark each statement as true or false.

43. Every poset has a maximal element.

44. Every poset has a minimal element.

45. The maximal element in a poset, if it exists, is unique.

46. The minimal element in a poset, if it exists, is unique.

47. Every poset has a greatest element.

48. Every poset has a least element.

Give a counterexample to disprove each statement.

49. Every poset has a maximal element.

50. Every poset has a minimal element.

51. Every poset has a greatest element.

52. Every poset has a least element.

Topologically sort the elements of each poset.

53. The poset in Figure 7.43. **54.** The poset in Figure 7.44.

55. The poset in Exercise 32. **56.** The poset in Exercise 33.

57. $(A, |)$, where $A = \{1, 2, 3, 6, 9, 18\}$ **58.** $(A, |)$, where $A = \{1, 2, 3, 6, 8, 24\}$

59. Topologically sort the tasks t_1 through t_{13} in building a house, given by Table 7.6.

60. A project contains six subprojects, A through F. Results from some of the subprojects are needed by others, as Table 7.7 shows. Find the ways the subprojects can be sequentially arranged.

Table 7.7

Subproject	Requires results from
A	B, D
B	C
C	None
D	C, E
E	None
F	A

61. Seven tasks, A through G, comprise a project. Some of them can only be started after others are completed, as indicated by Table 7.8. How many ways can the tasks be arranged sequentially, so the prerequisites of each task will be completed before it is started? List one of them.

Table 7.8

Task	Requires the completion of
A	B, C
B	G
C	None
D	A, F
E	None
F	B, E
G	None

***62.** Let $(A, \preceq_1)$ and $(B, \preceq_2)$ be two posets. Define a relation $\preceq_3$ on $A \times B$ as follows: $(a, b) \preceq_3 (a', b')$ if $a \preceq_1 a'$ and $b \preceq_2 b'$. Prove that $\preceq_3$ is a partial order.

Prove each.

***63.** The greatest element of a poset $(A, \preceq)$, if it exists, is unique.

***64.** The least element of a poset $(A, \preceq)$, if it exists, is unique.

***65.** Every finite nonempty poset $(A, \preceq)$ contains a maximal element.

***66.** Establish the correctness of Algorithm 7.4.

Chapter Summary

We studied the fundamentals of the theory of relations and explored how relations on finite sets can be represented by graphs and boolean matrices.

Boolean Matrix

- A **boolean matrix** has bits for entries (page 438).
- The **join** $A \vee B$ and **meet** $A \vee B$ of two boolean matrices A and B are obtained by *oring* and *anding* the corresponding bits, respectively (page 439).
- The **boolean product** $A \odot B$ of two boolean matrices $A = (a_{ij})_{m \times p}$ and $B = (b_{jk})_{p \times n}$ is the matrix $C = (c_{ij})_{m \times n}$, where $c_{ij} = (a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \cdots \vee (a_{ip} \wedge b_{pj})$ (page 439).

- The **complement** A' of a boolean matrix A results from swapping 0's and 1's (page 442).

Binary Relation

- A **binary relation** R from A to B is a subset of $A \times B$. If $(a, b) \in R$, we write aRb; otherwise, we write ab (page 443).
- A relation R from a finite set to a finite set can be represented by its **adjacency matrix**, M_R (page 444).
- A relation on a finite set can be represented by a **digraph** (page 445).
- Every function $f : A \to B$ is a binary relation from A to B such that (1) $\text{dom}(f) = A$; and (2) if $(a, b) \in f$ and $(a, c) \in f$, then $b = c$ (page 448).

Properties of Relations

- A relation R on A is **reflexive** if aRa for every $a \in A$ (page 455).
- A relation R on A is **symmetric** if aRb implies bRa (page 456).
- A relation R on A is **antisymmetric** if $aRb \wedge bRa$ implies $a = b$ (page 456).
- A relation R on A is **transitive** if $aRb \wedge bRc$ implies aRc (page 459).

Constructing New Relations

- The **union** and **intersection** of two relations R and S from A to B are $R \cup S = \{(a, b)|aRb \vee aSb\}; R \wedge S = \{(a, b)|aRb \vee aSb\}$ (page 462).
- If R and S are relations on a finite set, $M_{R \cup S} = M_R \vee M_S$ and $M_{R \cap S} = M_R \wedge M_S$ (page 463).
- Let R be a relation from A to B and S a relation from B to C. Their **composition** is $R \odot S = \{(a, c) \in A \times C \mid aRb \wedge bRc \text{ for some } b \text{ in } B\}$ (page 463).
- In particular, if A, B, and C are finite sets, then $M_{R \odot S} = M_R \odot M_S$ (page 466).
- For a relation R on a finite set, $M_{R^n} = (M_R)^{[n]}$ (page 467).
- For a transitive relation $R, R^n \subseteq R$ for every $n \geq 1$ (page 468).
- The **connectivity relation** R^∞ is the union of all powers of R:

$$R^\infty = \bigcup_{n=1}^{\infty} R^n; M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee \ldots \quad \text{(page 471).}$$

- In particular, let R be a relation on a set with size n. Then

$$R^{\infty} = \bigcup_{i=1}^{n} R^i \text{ and } M_R = M_R \vee M_{R^2} \vee \cdots \vee M_{R^n} \quad \text{(page 473).}$$

Transitive Closure

- The **transitive closure** R^* of a relation R is the smallest transitive relation containing it (page 475).
- $R^* = R^{\infty}$ (page 477).
- **Warshall's algorithm** systematically finds M_{R^*} (page 477).

Equivalence Relations and Partitions

- An **equivalence relation** is reflexive, symmetric, and transitive (page 483).
- An equivalence relation on a set induces a partition of the set and vice versa (page 488).

Partial and Total Orders

- A **partial order** $\preceq$ is reflexive, antisymmetric, and transitive. A set together with a partial order is a **poset** (page 494).
- Two elements, x and y, in a poset are **comparable** if either $x \preceq y$ or $y \preceq x$ (page 495).
- If any two elements in a poset are comparable, the partial order is a **total order** or **linear order** (page 495).
- The **lexicographic order** is an extension of the alphabetical order to posets (page 495).
- The **Hasse diagram** of a finite poset contains no loops, edges implied by transitivity, or arrows; its edges are drawn upward (page 496).
- The elements of a finite nonempty poset can be **sorted topologically** (page 500).

Review Exercises

Determine if each relation on $\{a, b, c\}$ is reflexive, symmetric, antisymmetric, or transitive.

1. $\{(a,a),(b,c),(c,b),(c,c)\}$

2. $\{(a,b),(b,a),(b,c),(c,b)\}$

Using the relations $R = \{(1,1),(1,2),(2,2),(3,2)\}$ and $S = \{(1,1),(2,2),(2,3),(3,2)\}$ on $\{1,2,3\}$, find each.

3. $(R \cup S)'$ **4.** $R' \cap S'$ **5.** $(R \cap S)'$ **6.** $R' \cup S'$

7. $(R \cup S)^{-1}$ **8.** $R^{-1} \cup S^{-1}$ **9.** $(R \cap S)^{-1}$ **10.** $R^{-1} \cap S^{-1}$

11. $R \odot S$ **12.** R^2 **13.** R^3 **14.** R^∞

With the adjacency matrices of the relations $R = \{(1,1),(1,2),(2,2),(3,2)\}$ and $S = \{(1,1),(2,2),(2,3),(3,2)\}$ on $\{1,2,3\}$, find each.

15. $(M_R)'$ **16.** $(M_R)^T$ **17.** $M_{R \odot S}$ **18.** M_{R^2}

19. $R \odot S$ **20.** R^2 **21.** M_{R^*} **22.** R^*

Find the transitive closure of the relation on $A = \{a,b,c\}$ with each adjacency matrix.

23. $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$ **24.** $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$

Since 1972 every book published commercially has a 10-digit identification number, its *International Standard Book Number* (**ISBN**). The ISBN consists of four parts: a group code (one digit), a publisher code (two digits), a book code (six digits), and a check digit. For instance, the ISBN of an earlier text by this author is 0-12-421171-2. The group code 0 indicates that the book was published in an English-speaking country. The publisher code (12) identifies the publisher, Academic Press, and the book code (421171) is assigned by the publisher to the book. The check digit d, where $0 \le d \le 10$ and 10 is denoted by X, is used to detect errors and is computed as follows: Let $x_1, x_2, \ldots, x_9$ denote the first nine digits in the ISBN. Let s denote the dot product of the 9-tuples $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9)$ and $(10, 9, 8, 7, 6, 5, 4, 3, 2)$. Then $d \equiv -s \pmod{11}$. Compute the check digit if the first 9 digits of the ISBN are:

25. 0-12-421171 **26.** 0-87-620321

Determine if each is an equivalence relation on $\{a,b,c\}$.

27. $\{(a,a),(a,b),(b,a),(c,c)\}$ **28.** $\{(a,a),(a,c),(b,b),(c,a),(c,c)\}$

Using the equivalence relation $\{(a,a),(a,c),(b,b),(b,d),(c,a),(c,c),(d,b),(d,d)\}$ on $A = \{a,b,c,d\}$, find each equivalence class.

29. $[a]$ **30.** $[b]$ **31.** $[c]$ **32.** $[d]$

33. Find the partition of A induced by the above relation.

Find the equivalence relation corresponding to each partition of the set $\{2,3,4,7\}$.

34. $\{\{2,4,7\},\{3\}\}$ **35.** $\{\{2,4\},\{3\},\{7\}\}$

Find the number of partitions of a set with the given size.

36. Two **37.** Seven

Mark each statement as true or false, where A is an arbitrary set, R an arbitrary relation, and Δ the equality relation.

38. The null relation is reflexive.

39. The null relation is symmetric.

40. The null relation is transitive.

41. A relation R on A is reflexive if and only if $\Delta \subseteq R$.

42. The less-than relation on $\mathbb{R}$ is irreflexive.

43. The less-than relation on $\mathbb{R}$ is antisymmetric.

44. If R is transitive, $R^* = R$.

45. If $R^* = R$, R is transitive.

46. The less-than relation on $\mathbb{R}$ is a partial order.

47. The less-than relation on $\mathbb{R}$ is a total order.

48. Arrange all binary words of length 3 in lexicographic order, where $0 \prec 1$.

49. Arrange all binary words of length ≤ 3 in lexicographic order, where $0 \prec 1$.

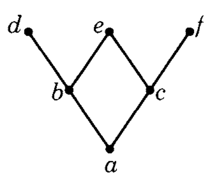
The relation $\preceq$ on the set A of required courses given in Table 7.1 by $x \preceq y$ if x is a prerequisite of or the same as y is a partial order on A.

50. Draw the Hasse diagram for the poset.

51. Topologically sort the required computer science courses.

Use the poset in Figure 7.55 to find the following.

Figure 7.55



52. The maximal and minimal elements, if they exist.

53. The greatest and least elements, if they exist.

54. Topologically sort the elements in the poset.

Let R and S be any two relations on a set A. Prove each.

55. $(R \cap S)^2 \subseteq R^2 \cap S^2$

56. $(R \cap S)^n \subseteq R^n \cap S^n, n \geq 1$

57. R is antisymmetric if and only if $R \cap R^{-1} \subseteq \Delta$.

58. The intersection of two equivalence relations is an equivalence relation.

Supplementary Exercises

Let D denote any day of the week, where $0 \leq D \leq 6$ and $D = 0$ denotes Sunday. The day of the week corresponding to any day $(m/d/y)$ in the Gregorian calendar is given by

$$D \equiv \begin{cases} \left\lfloor \frac{23m}{9} \right\rfloor + d + 4 + y + \left\lfloor \frac{y-1}{4} \right\rfloor - \left\lfloor \frac{y-1}{100} \right\rfloor + \left\lfloor \frac{y-1}{400} \right\rfloor \pmod 7 & \text{if } m < 3 \\ \left\lfloor \frac{23m}{9} \right\rfloor + d + 4 + y + \left\lfloor \frac{y}{4} \right\rfloor - \left\lfloor \frac{y}{100} \right\rfloor + \left\lfloor \frac{y}{400} \right\rfloor - 2 \pmod 7 & \text{otherwise} \end{cases}$$

(M. Keith and T. Carver, 1990)
Compute the day of each date.

1. July 4, 1776

2. December 25, 2076

Prove each.

3. Let p be a prime. Then $p \mid \binom{p}{k}$ for $0 < k < p$.

4. (**Fermat's theorem**) Let $a \in \mathbb{N}$ and p a prime. Then $a^p \equiv a \pmod p$. (*Hint*: Use induction.)

Let $a, b \in \mathbb{R}$ and p a prime. Prove that $(a+b)^p \equiv a^p + b^p \pmod p$ using:

5. The binomial theorem and Exercise 3.

6. Fermat's theorem.

Evaluate each.

7. $5^{1000} \pmod 7$

8. $12^{4000} \pmod 5$

9. Prove that the product of any three consecutive integers is divisible by 3.

10. Let $n \in \mathbf{W}$. Prove that the number formed by concatenating the decimal values of 2^n and 2^{n+1} is divisible by 3. (For example, when $n = 5$, both 3264 and 6432 are divisible by 3.) (D. Burns, 1977)

11. Around 1760, John Wilson (1741–1793), an English mathematician, proved that $(p-1)! \equiv -1 \pmod p$; that is, the quotient

$$W(p) = \frac{(p-1)!+1}{p}$$

is an integer. This is known as **Wilson's theorem.**) p is a **Wilson prime** if $W(p) \equiv 0 \pmod p$; that is, if $(p-1)! \equiv -1 \pmod{p^2}$. Find the two Wilson's primes < 20. (The third and largest known Wilson prime is 563. It is not known whether or not there are infinitely many Wilson primes.)

12. (**Lucas' Theorem**) Let p be a prime, $n = (a_t a_{t-1} \ldots a_0)_p$ and $k = (b_t b_{t-1} \ldots b_0)_p$. Then $\binom{n}{k} \equiv \binom{a_t}{b_t}\binom{a_{t-1}}{b_{t-1}} \ldots \binom{a_0}{b_0} \pmod p$. Using Lucas' theorem, find the remainders when the binomial coefficients $C(234,19)$ and $C(3456,297)$ are divided by 5.

***13.** Let a and b be relatively prime integers. Prove that $a^{\varphi(b)} + b^{\varphi(a)} \equiv 1 \pmod{ab}$.
(*Hint*: Let $n \in \mathbb{N}$ and a an integer relatively prime to n. Then $a^{\varphi(n)} \equiv 1 \pmod n$. This is **Euler's theorem.**) (M. Charosh, 1983)

***14.** Show that a set with n elements must have at least 2^n relations with the same reflexive closure.
(*Hint*: Use the pigeonhole principle.)

***15.** Show that a set with size n must have at least $2^{n(n-1)/2}$ relations with the same symmetric closure.
(*Hint*: Use the pigeonhole principle.)

Computer Exercises

Write a program to perform each task, where n denotes a positive integer ≤ 20 and $A = \{1, 2, \ldots, n\}$.

1. Read in two boolean matrices. Print their join, meet, complement, and boolean product, if defined.

2. Read in the elements of a relation R on A. Print its adjacency matrix M_R. Use M_R to enumerate the elements in the relation.

3. Read in the adjacency matrix of a relation on A. Determine if the relation is reflexive, symmetric, antisymmetric, or transitive.

4. Read in the adjacency matrices of two relations on A. Print the adjacency matrices of their union, intersection, complements, and inverses.

5. Read in the adjacency matrix of a relation R from A to B and that of a relation S from B to C. Print $M_{R \odot S}$.

6. Read in the adjacency matrix of a relation R on A. Print M_{R^*}, using the connectivity relation algorithm and Warshall's algorithm, and compare the number of bit operations required by them.
7. Read in the adjacency matrix of a relation on A. Determine if the relation is an equivalence relation.
8. Read in two positive integers r and n, where $r \leq n \leq 10$. Print the number of equivalence relations that can be defined on a set of size n, using Stirling numbers of the second kind and Bell numbers.
9. Read in the adjacency matrix of a partial order on a poset A.
   - Determine if it is a partial order.
   - Print the boolean matrix corresponding to its Hasse diagram.
   - Topologically sort the elements of the poset.
10. Determine the most likely day on which the 13th of a month will fall in the Gregorian calendar. Since the Gregorian calendar repeats every 400 years, you need only consider a period of 400 years.
11. Read in a positive integer $n \leq 1000$ and print all Wilson primes $\leq n$.

Exploratory Writing Projects

Using library and Internet resources, write a team report on each of the following in your own words. Provide a well-documented bibliography.

1. Explain and illustrate the various relational operations in the theory of databases.
2. Describe the algorithm employed by the United States Postal Service to encode the nine-digit zip code into barcodes, and decode the barcodes (62 bars) into zip codes.
3. Describe how modular arithmetic can be used to construct m-pointed stars.
4. Explain the coding scheme for creating *European Article Numbering* (EAN) barcodes to uniquely identify books. Extend it to include the five-digit add-on code to provide price information.
5. Describe the origins of the Julian and Gregorian calendars.
6. Develop a formula to determine the day d of the week for the rth day in a given month m of any given year y in the Gregorian calendar, where $y > 1600$.
7. Study the algorithms of assigning driver's license numbers in various states.

8. State and prove *the Chinese Remainder Theorem.* Illustrate it using ancient examples from China and India.

9. Write an essay on the various cryptosystems.

Enrichment Readings

1. A. V. Aho *et al.*, *Data Structures and Algorithms*, Addison-Wesley, Reading, MA, 1983.
2. W. T. Bailey, "Friday-the-Thirteenth,"*Mathematics Teacher,* Vol. 62 (May 1969), pp. 363–364.
3. J. A. Gallian, "Assigning Driver's License Numbers," *Mathematics Magazine*, Vol. 64 (Feb. 1991), pp. 13–22.
4. J. A. Gallian, "The Mathematics of Identification Numbers," *The College Mathematics Journal*, Vol. 22 (May 1991), pp. 194–202.
5. J. A. Gallian and S. Winters, "Modular Arithmetic in the Marketplace," *The American Mathematical Monthly*, Vol. 95 (June–July 1988), pp. 548–551.
6. D. W. Hardy and C. L. Walker, *Applied Algebra: Codes, Ciphers, and Discrete Algorithms*, Prentice-Hall, Upper Saddle River, NJ, 2003.
7. T. Koshy, *Elementary Number Theory with Applications*, Harcourt/ Academic Press, Boston, MA, 2002, pp. 210–436.
8. P. Lefton, "Number Theory and Public-Key Cryptology," *Mathematics Teacher*, Vol. 84 (Jan. 1991), pp. 54–62.
9. R. E. Lewand, *Cryptological Mathematics,* Math. Association of America, Washington, D.C., 2000.
10. J. E. Shockley, *Introduction to Number Theory*, Holt, Rinehart and Winston, New York, 1967, pp. 36–69.
11. J. R. Snow, "An Application of Number Theory to Cryptology," *Mathematics Teacher*, Vol. 82 (Jan. 1989), pp. 18–26.
12. P. M. Tuchinsky, "International Standard Book Number," *The UMAP Journal*, Vol. 5 (1985), pp. 41–54.