

Strong Induction and Well-Ordering

Section 5.2

Strong Induction

- **Strong Induction:** To prove that $P(n)$ is true for all positive integers n , where $P(n)$ is a propositional function, complete two steps:
 - *Basis Step:* Verify that the proposition $P(1)$ is true.
 - *Inductive Step:* Show the conditional statement $[P(1) \wedge P(2) \wedge \cdots \wedge P(k)] \rightarrow P(k + 1)$ holds for all positive integers k .

Strong Induction is sometimes called the *second principle of mathematical induction* or *complete induction*.

Strong Induction and the Infinite Ladder

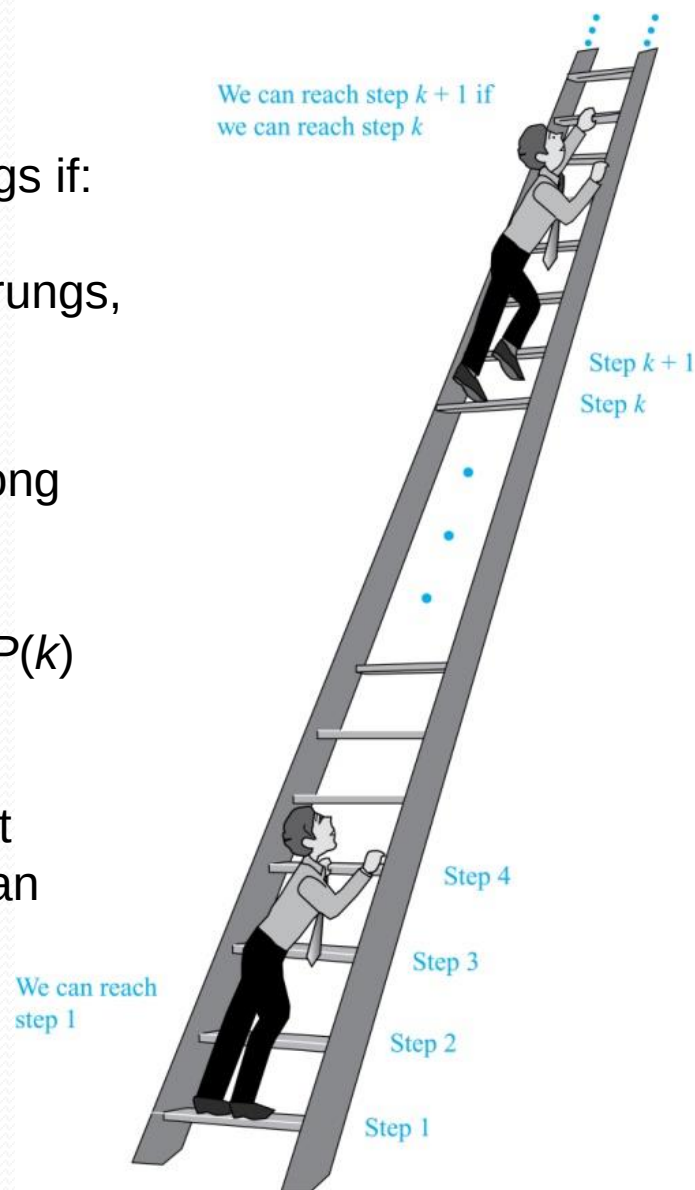
Strong induction tells us that we can reach all rungs if:

1. We can reach the first rung of the ladder.
2. For every integer k , if we can reach the first k rungs, then we can reach the $(k + 1)$ st rung.

To conclude that we can reach every rung by strong induction:

- BASIS STEP: $P(1)$ holds
- INDUCTIVE STEP: Assume $P(1) \wedge P(2) \wedge \dots \wedge P(k)$ holds for an arbitrary integer k , and show that $P(k + 1)$ must also hold.

We will have then shown by strong induction that for every positive integer n , $P(n)$ holds, i.e., we can reach the n th rung of the ladder.



Proof using Strong Induction

Example: Suppose we can reach the first and second rungs of an infinite ladder, and we know that if we can reach a rung, then we can reach two rungs higher. Prove that we can reach every rung.

(Try this with mathematical induction.)

Solution: Prove the result using strong induction.

- **BASIS STEP:** We can reach the first step.
- **INDUCTIVE STEP:** The inductive hypothesis is that we can reach the first k rungs, for any $k \geq 2$. We can reach the $(k + 1)$ st rung since we can reach the $(k - 1)$ st rung by the inductive hypothesis.
- Hence, we can reach all rungs of the ladder.



Which Form of Induction Should Be Used?

- We can always use strong induction instead of mathematical induction. But there is no reason to use it if it is simpler to use mathematical induction. (*See page 335 of text.*)
- In fact, the **principles** of mathematical induction, strong induction, and the well-ordering property are all **equivalent**. (*Exercises 41-43*)
- Sometimes it is clear how to proceed using one of the three methods, but not the other two.

Completion of the proof of the Fundamental Theorem of Arithmetic

Example: Show that if n is an integer greater than 1, then n can be written as the product of primes.

Solution: Let $P(n)$ be the proposition that n can be written as a product of primes.

- BASIS STEP: $P(2)$ is true since 2 itself is prime.
- INDUCTIVE STEP: The inductive hypothesis is $P(j)$ is true for all integers j with $2 \leq j \leq k$. To show that $P(k + 1)$ must be true under this assumption, two cases need to be considered:
 - If $k + 1$ is prime, then $P(k + 1)$ is true.
 - Otherwise, $k + 1$ is composite and can be written as the product of two positive integers a and b with $2 \leq a \leq b < k + 1$. By the inductive hypothesis a and b can be written as the product of primes and therefore $k + 1$ can also be written as the product of those primes.

Hence, it has been shown that every integer greater than 1 can be written as the product of primes.

(uniqueness proved in Section 4.3)



Proof using Strong Induction

Example: Prove that every amount of postage of 12 cents or more can be formed using just 4-cent and 5-cent stamps.

Solution: Let $P(n)$ be the proposition that postage of n cents can be formed using 4-cent and 5-cent stamps.

- BASIS STEP: $P(12)$, $P(13)$, $P(14)$, and $P(15)$ hold.
 - $P(12)$ uses three 4-cent stamps.
 - $P(13)$ uses two 4-cent stamps and one 5-cent stamp.
 - $P(14)$ uses one 4-cent stamp and two 5-cent stamps.
 - $P(15)$ uses three 5-cent stamps.
- INDUCTIVE STEP: The inductive hypothesis states that $P(j)$ holds for $12 \leq j \leq k$, where $k \geq 15$. Assuming the inductive hypothesis, it can be shown that $P(k + 1)$ holds.
- Using the inductive hypothesis, $P(k - 3)$ holds since $k - 3 \geq 12$. To form postage of $k + 1$ cents, add a 4-cent stamp to the postage for $k - 3$ cents.

Hence, $P(n)$ holds for all $n \geq 12$.



Proof of Same Example using Mathematical Induction

Example: Prove that every amount of postage of 12 cents or more can be formed using just 4-cent and 5-cent stamps.

Solution: Let $P(n)$ be the proposition that postage of n cents can be formed using 4-cent and 5-cent stamps.

- **BASIS STEP:** Postage of 12 cents can be formed using three 4-cent stamps.
- **INDUCTIVE STEP:** The inductive hypothesis $P(k)$ for any positive integer k is that postage of k cents can be formed using 4-cent and 5-cent stamps. To show $P(k + 1)$ where $k \geq 12$, we consider two cases:
 - If at least one 4-cent stamp has been used, then a 4-cent stamp can be replaced with a 5-cent stamp to yield a total of $k + 1$ cents.
 - Otherwise, no 4-cent stamp have been used and at least three 5-cent stamps were used. Three 5-cent stamps can be replaced by four 4-cent stamps to yield a total of $k + 1$ cents.

Hence, $P(n)$ holds for all $n \geq 12$.



Well-Ordering Property

- **Well-ordering property:** Every nonempty set of **nonnegative integers** has a least element.
- The well-ordering property is one of the axioms of the positive integers listed in Appendix 1.
- The well-ordering property can be used directly in proofs, as the next example illustrates.
- The well-ordering property can be generalized.
 - **Definition:** A set is *well ordered* if every subset has a **least element**.
 - $\mathbf{N}$ is well ordered under $\leq$.
 - The set of finite strings over an alphabet using lexicographic ordering is well ordered.
 - We will see a generalization of induction to sets other than the integers in the next section.

Well-Ordering Property

Example: Use the well-ordering property to prove the division algorithm, which states that if a is an integer and d is a positive integer, then there are unique integers q and r with $0 \leq r < d$, such that $a = dq + r$.

Solution: Let S be the set of nonnegative integers of the form $a - dq$, where q is an integer. The set is nonempty since $-dq$ can be made as large as needed.

- By the well-ordering property, S has a least element $r = a - dq_0$. The integer r is nonnegative. It also must be the case that $r < d$. If it were not, then there would be a smaller nonnegative element in S , namely,
 $a - d(q_0 + 1) = a - dq_0 - d = r - d > 0$.
- Therefore, there are integers q and r with $0 \leq r < d$.

(uniqueness of q and r is Exercise 37)



Recursive Definitions and Structural Induction

Section 5.3

Recursively Defined Functions

Definition: A *recursive* or *inductive definition* of a function consists of two steps.

- BASIS STEP: Specify the value of the function at zero.
- RECURSIVE STEP: Give a rule for finding its value at an integer from its values at smaller integers.
- A function $f(n)$ is the same as a sequence $a_0, a_1, \dots, a_j$, where $f(i) = a_i$. This was done using recurrence relations in Section 2.4.

Recursively Defined Functions

Example: Suppose f is defined by:

$$f(0) = 3,$$

$$f(n + 1) = 2f(n) + 3$$

Find $f(1)$, $f(2)$, $f(3)$, $f(4)$

Solution:

- $f(1) = 2f(0) + 3 = 2 \cdot 3 + 3 = 9$
- $f(2) = 2f(1) + 3 = 2 \cdot 9 + 3 = 21$
- $f(3) = 2f(2) + 3 = 2 \cdot 21 + 3 = 45$
- $f(4) = 2f(3) + 3 = 2 \cdot 45 + 3 = 93$

Example: Give a recursive definition of the factorial function $n!$.

Solution:

$$f(0) = 1$$

$$f(n + 1) = (n + 1) \cdot f(n)$$

Recursively Defined Functions

Example: Give a recursive definition of:

$$\sum_{k=0}^n a_k.$$

Solution: The first part of the definition is

$$\sum_{k=0}^0 a_k = a_0.$$

The second part is

$$\sum_{k=0}^{n+1} a_k = \left(\sum_{k=0}^n a_k \right) + a_{n+1}.$$

Fibonacci
(1170- 1250)



Fibonacci Numbers

Example : The Fibonacci numbers are defined as follows:

$$f_0 = 0$$

$$f_1 = 1$$

$$f_n = f_{n-1} + f_{n-2}$$

Find f_2, f_3, f_4, f_5 .

- $f_2 = f_1 + f_0 = 1 + 0 = 1$
- $f_3 = f_2 + f_1 = 1 + 1 = 2$
- $f_4 = f_3 + f_2 = 2 + 1 = 3$
- $f_5 = f_4 + f_3 = 3 + 2 = 5$

In Chapter 8, we will use the Fibonacci numbers to model population growth of rabbits. This was an application described by Fibonacci himself.

Next, we use strong induction to prove a result about the Fibonacci numbers.



الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

Chapter 6

Counting

Chapter Outlines

The Basics of Counting

The Pigeonhole Principle

Permutations and Combinations

Binomial Coefficients and Identities

Generalized Permutations and Combinations

The Basics of Counting

Section 6.1

Basic Counting Principles: The Product Rule

The Product Rule: A procedure can be broken down into a **sequence** of two tasks. There are n_1 ways to do the first task and n_2 ways to do the second task. Then there are $n_1 \cdot n_2$ ways to do the procedure.

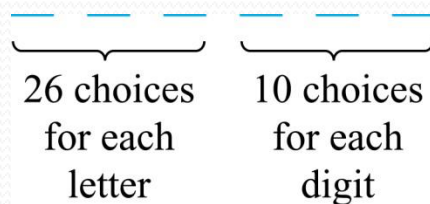
Example: How many bit strings of length seven are there?

Solution: Since each of the seven bits is either a 0 or a 1, the answer is $2^7 = 128$.

The Product Rule

Example: How many different license plates can be made if each plate contains a sequence of three uppercase English letters followed by three digits?

Solution: By the product rule,
there are $26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 17,576,000$ different possible license plates.



Counting Functions

Counting Functions: How many **functions** are there from a set with m elements to a set with n elements?

Solution: Since a function represents a choice of one of the n elements of the codomain for each of the m elements in the domain, the **product rule** tells us that there are $n \cdot n \cdot \cdots \cdot n = n^m$ such functions.

Counting One-to-One Functions: How many one-to-one functions are there from a set with m elements to one with n elements?

Solution: Suppose the elements in the domain are $a_1, a_2, \dots, a_m$. There are **n ways** to choose the value of a_1 and $n-1$ ways to choose a_2 , etc. The product rule tells us that there are $n(n-1)(n-2)\cdots(n-m+1)$ such functions.

Telephone Numbering Plan

Example: The *North American numbering plan (NANP)* specifies that a telephone number consists of 10 digits, consisting of a three-digit area code, a three-digit office code, and a four-digit station code. There are some restrictions on the digits.

- Let X denote a digit from 0 through 9.
- Let N denote a digit from 2 through 9.
- Let Y denote a digit that is 0 or 1.
- In the old plan (in use in the 1960s) the format was $NYX-NNX-XXXX$.
- In the new plan, the format is $NXX-NXX-XXXX$.

How many different telephone numbers are possible under the old plan and the new plan?

Solution: Use the Product Rule.

- There are $8 \cdot 2 \cdot 10 = 160$ area codes with the format NYX .
- There are $8 \cdot 10 \cdot 10 = 800$ area codes with the format NXX .
- There are $8 \cdot 8 \cdot 10 = 640$ office codes with the format NNX .
- There are $10 \cdot 10 \cdot 10 \cdot 10 = 10,000$ station codes with the format $XXXX$.

Number of old plan telephone numbers: $160 \cdot 640 \cdot 10,000 = 1,024,000,000$.

Number of new plan telephone numbers: $800 \cdot 800 \cdot 10,000 = 6,400,000,000$.

Counting Subsets of a Finite Set

Counting Subsets of a Finite Set: Use the product rule to show that the number of different subsets of a finite set S is $2^{|S|}$. (*In Section 5.1, **mathematical induction** was used to prove this same result.*)

Solution: When the elements of S are listed in an arbitrary order, there is a **one-to-one correspondence** between subsets of S and bit strings of length $|S|$. When the i th element is in the subset, the bit string has a 1 in the i th position and a 0 otherwise.

By the product rule, there are $2^{|S|}$ such bit strings, and therefore $2^{|S|}$ subsets.

Product Rule in Terms of Sets

- If $A_1, A_2, \dots, A_m$ are finite sets, then the number of elements in the Cartesian product of these sets is the product of the number of elements of each set.
- The task of choosing an element in the Cartesian product $A_1 \times A_2 \times \dots \times A_m$ is done by choosing an element in A_1 , an element in A_2 , ..., and an element in A_m .
- By the product rule, it follows that:

$$|A_1 \times A_2 \times \dots \times A_m| = |A_1| \cdot |A_2| \cdot \dots \cdot |A_m|.$$

DNA and Genomes

- A **gene** is a segment of a DNA molecule that encodes a particular protein and the entirety of genetic information of an organism is called its **genome**.
- DNA molecules consist of two strands of blocks known as nucleotides. Each nucleotide is composed of bases: adenine (A), cytosine (C), guanine (G), or thymine (T).
- The DNA of bacteria has between 10^5 and 10^7 links (one of the four bases). Mammals have between 10^8 and 10^{10} links. So, by the product rule there are at least 4^{10^5} different sequences of bases in the DNA of bacteria and 4^{10^8} different sequences of bases in the DNA of mammals.
- The human genome includes approximately 23,000 genes, each with 1,000 or more links.
- Biologists, mathematicians, and computer scientists all work on determining the DNA sequence (genome) of different organisms.

Basic Counting Principles: The Sum Rule

The Sum Rule: If a task can be done either in one of n_1 ways or in one of n_2 ways to do the second task, where none of the set of n_1 ways is the same as any of the n_2 ways, then there are $n_1 + n_2$ ways to do the task.

Example: The mathematics department must choose either a student or a faculty member as a representative for a university committee. How many choices are there for this representative if there are 37 members of the mathematics faculty and 83 mathematics majors and no one is both a faculty member and a student.

Solution: By the sum rule it follows that there are $37 + 83 = 120$ possible ways to pick a representative.

The Sum Rule in terms of sets.

- The sum rule can be phrased in terms of sets.

$|A \cup B| = |A| + |B|$ as long as A and B are **disjoint sets**.

- Or more generally,

$$|A_1 \cup A_2 \cup \cdots \cup A_m| = |A_1| + |A_2| + \cdots + |A_m|$$

when $A_i \cap A_j = \emptyset$ for all i, j .

- The case where the sets have elements in common will be discussed when we consider the subtraction rule and taken up fully in Chapter 8.

Combining the Sum and Product Rule

Example: Suppose statement labels in a programming language can be either a single letter or a letter followed by a digit. Find the number of possible labels.

Solution: Use the product rule.

$$26 + 26 \cdot 10 = 286$$

Counting Passwords

- Combining the sum and product rule allows us to solve more complex problems.

Example: Each user on a computer system has a password, which is six to eight characters long, where each character is an uppercase letter or a digit. Each password must contain at least one digit. How many possible passwords are there?

Solution: Let P be the total number of passwords, and let P_6 , P_7 , and P_8 be the passwords of length 6, 7, and 8.

- By the sum rule $P = P_6 + P_7 + P_8$.
- To find each of P_6 , P_7 , and P_8 , we find the number of passwords of the specified length composed of letters and digits and subtract the number composed only of letters. We find that:

$$P_6 = 36^6 - 26^6 = 2,176,782,336 - 308,915,776 = 1,867,866,560.$$

$$P_7 = 36^7 - 26^7 =$$

$$78,364,164,096 - 8,031,810,176 = 70,332,353,920.$$

$$P_8 = 36^8 - 26^8 =$$

$$2,821,109,907,456 - 208,827,064,576 = 2,612,282,842,880.$$

Consequently, $P = P_6 + P_7 + P_8 = 2,684,483,063,360$.

Internet Addresses

- Version 4 of the Internet Protocol (IPv4) uses 32 bits.

Bit Number	0	1	2	3	4	8	16	24	31	
Class A	0	netid					hostid			
Class B	1	0	netid					hostid		
Class C	1	1	0	netid					hostid	
Class D	1	1	1	0	Multicast Address					
Class E	1	1	1	1	0	Address				

- Class A Addresses:** used for the largest networks, a 0, followed by a 7-bit netid and a 24-bit hostid.
- Class B Addresses:** used for the medium-sized networks, a 10, followed by a 14-bit netid and a 16-bit hostid.
- Class C Addresses:** used for the smallest networks, a 110, followed by a 21-bit netid and a 8-bit hostid.
 - Neither Class D nor Class E addresses are assigned as the address of a computer on the internet. Only Classes A, B, and C are available.
 - 1111111 is not available as the netid of a Class A network.
 - Hostids consisting of all 0s and all 1s are not available in any network.

Counting Internet Addresses

Example: How many different IPv4 addresses are available for computers on the internet?

Solution: Use both the sum and the product rule. Let x be the number of available addresses, and let x_A , x_B , and x_C denote the number of addresses for the respective classes.

- To find, x_A : $2^7 - 1 = 127$ netids. $2^{24} - 2 = 16,777,214$ hostids.
 $x_A = 127 \cdot 16,777,214 = 2,130,706,178$.
- To find, x_B : $2^{14} = 16,384$ netids. $2^{16} - 2 = 16,534$ hostids.
 $x_B = 16,384 \cdot 16,534 = 1,073,709,056$.
- To find, x_C : $2^{21} = 2,097,152$ netids. $2^8 - 2 = 254$ hostids.
 $x_C = 2,097,152 \cdot 254 = 532,676,608$.
- Hence, the total number of available IPv4 addresses is

$$\begin{aligned}x &= x_A + x_B + x_C \\&= 2,130,706,178 + 1,073,709,056 + 532,676,608 \\&= 3,737,091,842.\end{aligned}$$

Not Enough Today !!

The newer IPv6 protocol solves the problem of too few addresses.

Basic Counting Principles: Subtraction Rule

Subtraction Rule: If a task can be done either in one of n_1 ways or in one of n_2 ways, then the total number of ways to do the task is $n_1 + n_2$ minus the number of ways to do the task that are common to the two different ways.

- Also known as, the *principle of inclusion-exclusion*:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

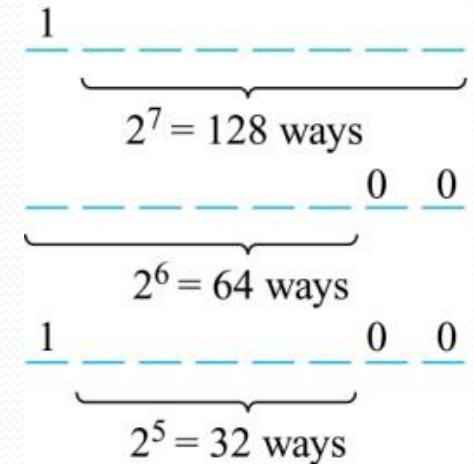
Counting Bit Strings

Example: How many bit strings of length eight either start with a 1 bit or end with the two bits 00?

Solution: Use the subtraction rule.

- Number of bit strings of length eight that start with a 1 bit: $2^7 = 128$
- Number of bit strings of length eight that end with bits 00: $2^6 = 64$
- Number of bit strings of length eight that start with a 1 bit and end with bits 00 : $2^5 = 32$

Hence, the number is $128 + 64 - 32 = 160$.



Basic Counting Principles: Division Rule

Division Rule: There are n/d ways to do a task if it can be done using a procedure that can be carried out in n ways, and for every way w , exactly d of the n ways correspond to way w .

- Restated in terms of sets: If the finite set A is the union of n **pairwise disjoint** subsets each with d elements, then $n = |A|/d$.
- In terms of functions: If f is a function from A to B , where both are finite sets, and for every value $y \in B$ there are exactly d values $x \in A$ such that $f(x) = y$, then $|B| = |A|/d$.

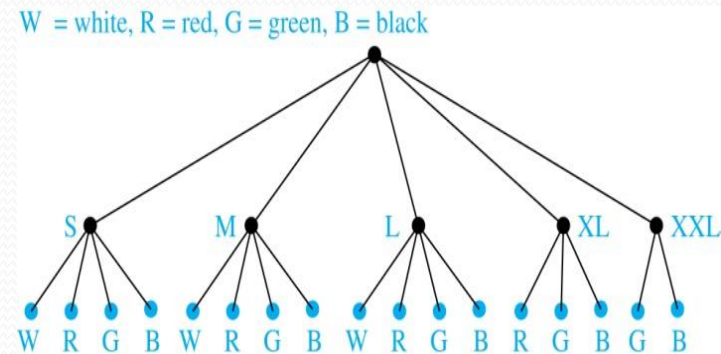
Example: How many ways are there to seat four people around a circular table, where two seatings are considered the same when each person has the same left and right neighbor?

Solution: Number the seats around the table from 1 to 4 proceeding clockwise. There are four ways to select the person for seat 1, 3 for seat 2, 2, for seat 3, and one way for seat 4. Thus there are $4! = 24$ ways to order the four people. But since two seatings are the same when each person has the same left and right neighbor, for every choice for seat 1, we get the same seating.

Therefore, by the division rule, there are $24/4 = 6$ different seating arrangements.

Tree Diagrams

- **Tree Diagrams:** We can solve many counting problems through the use of *tree diagrams*, where a *branch* represents a possible choice and the *leaves* represent possible outcomes.
- **Example:** Suppose that “I Love Discrete Math” T-shirts come in five different sizes: S,M,L,XL, and XXL. Each size comes in four colors (white, red, green, and black), except XL, which comes only in red, green, and black, and XXL, which comes only in green and black. What is the minimum number of stores that the campus book store needs to stock to have one of each size and color available?
- **Solution:** Draw the tree diagram.



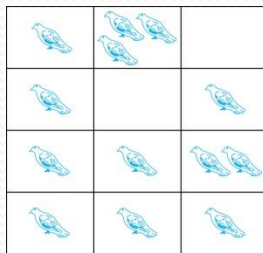
- The store must stock 17 T-shirts.

The Pigeonhole Principle

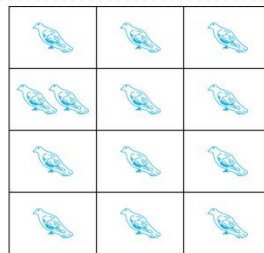
Section 6.2

The Pigeonhole Principle

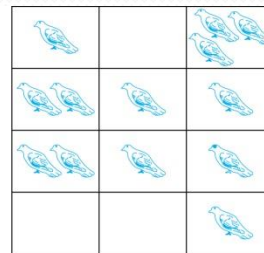
- If a flock of 20 pigeons roosts in a set of 19 pigeonholes, one of the pigeonholes must have more than 1 pigeon.



(a)



(b)



(c)

Pigeonhole Principle: If k is a positive integer and $k + 1$ objects are placed into k boxes, then **at least** one box contains two or more objects.

Proof: We use a proof by **contraposition**. Suppose none of the k boxes has more than one object. Then the total number of objects would be at most k . This **contradicts** the statement that we have $k + 1$ objects. ◀

The Pigeonhole Principle

Corollary 1: A function f from a set with $k + 1$ elements to a set with k elements is not one-to-one.

Proof: Use the pigeonhole principle.

- Create a box for each element y in the codomain of f .
- Put in the box for y all of the elements x from the domain such that $f(x) = y$.
- Because there are $k + 1$ elements and only k boxes, at least one box has two or more elements.

Hence, f can't be one-to-one.



Pigeonhole Principle

Example: Among any group of 367 people, there must be at least two with the same birthday, because there are only 366 possible birthdays.

Example (optional): Show that for every integer n there is a multiple of n that has only 0s and 1s in its decimal expansion.

Solution: Let n be a positive integer. Consider the $n + 1$ integers 1, 11, 111, ..., 11...1 (where the last has $n + 1$ 1s). There are n possible remainders when an integer is divided by n . By the pigeonhole principle, when each of the $n + 1$ integers is divided by n , at least two must have the same remainder. Subtract the smaller from the larger and the result is a multiple of n that has only 0s and 1s in its decimal expansion.

The Generalized Pigeonhole Principle

The Generalized Pigeonhole Principle: If N objects are placed into k boxes, then there is at least one box containing at least $\lceil N/k \rceil$ objects.

Proof: We use a proof by contraposition. Suppose that none of the boxes contains more than $\lceil N/k \rceil - 1$ objects. Then the total number of objects is at most

$$k \left(\left\lceil \frac{N}{k} \right\rceil - 1 \right) < k \left(\left(\frac{N}{k} + 1 \right) - 1 \right) = N,$$

where the inequality $\lceil N/k \rceil < \lceil N/k \rceil + 1$ has been used. This is a contradiction because there are a total of n objects.



Example: Among 100 people there are at least who were born in the same month.

$$\lceil 100/12 \rceil = 9$$

The Generalized Pigeonhole Principle

Example: a) How many cards must be selected from a standard deck of 52 cards to guarantee that at least three cards of the same suit are chosen?

b) How many must be selected to guarantee that at least three hearts are selected?

Solution: a) We assume four boxes; one for each suit. Using the generalized pigeonhole principle, at least one box contains at least $\lceil N/4 \rceil$ cards. At least three cards of one suit are selected if $\lceil N/4 \rceil \geq 3$. The smallest integer N such that $\lceil N/4 \rceil \geq 3$ is $N = 2 \cdot 4 + 1 = 9$.

b) A deck contains 13 hearts and 39 cards which are not hearts. So, if we select 41 cards, we may have 39 cards which are not hearts along with 2 hearts. However, when we select 42 cards, we must have at least three hearts. (Note that the generalized pigeonhole principle is not used here.)

Permutations and Combinations

Section 6.3

Permutations

Definition: A **permutation** of a set of distinct objects is an **ordered arrangement** of these objects. An ordered arrangement of r elements of a set is called an **r -permutation**.

Example: Let $S = \{1,2,3\}$.

- The ordered arrangement 3,1,2 is a **3-permutation** of S .
- The ordered arrangement 3,2 is a **2-permutation** of S .
- The number of r -permutations of a set with n elements is denoted by **$P(n,r)$** .
 - The 2-permutations of $S = \{1,2,3\}$ are 1,2; 1,3; 2,1; 2,3; 3,1; and 3,2. Hence, $P(3,2) = 6$.

A Formula for the Number of Permutations

Theorem 1: If n is a positive integer and r is an integer with $1 \leq r \leq n$, then there are

$$P(n, r) = n(n - 1)(n - 2) \cdots (n - r + 1)$$

r -permutations of a set with n distinct elements.

Proof: Use the product rule. The first element can be chosen in n ways. The second in $n - 1$ ways, and so on until there are $(n - (r - 1))$ ways to choose the last element.

- Note that $P(n, 0) = 1$, since there is only **one way** to order zero elements.

Corollary 1: If n and r are integers with $1 \leq r \leq n$, then

$$P(n, r) = \frac{n!}{(n-r)!}$$

Solving Counting Problems by Counting Permutations

Example: How many ways are there to select a first-prize winner, a second prize winner, and a third-prize winner from 100 different people who have entered a contest?

Solution:

$$P(100,3) = 100 \cdot 99 \cdot 98 = 970,200$$

Solving Counting Problems by Counting Permutations (*continued*)

Example: Suppose that a saleswoman has to visit eight different cities. She must begin her trip in a specified city, but she can visit the other seven cities in any order she wishes. How many possible orders can the saleswoman use when visiting these cities?

Solution: The first city is chosen, and the rest are ordered arbitrarily. Hence the orders are:

$$7! = 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 5040$$

If she wants to find the tour with the shortest path that visits all the cities, she must consider 5040 paths!

Solving Counting Problems by Counting Permutations (*continued*)

Example: How many permutations of the letters *ABCDEFGH* contain the string *ABC* ?

Solution: We solve this problem by counting the permutations of six objects, *ABC*, *D*, *E*, *F*, *G*, and *H*.

$$6! = 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 720$$

Combinations

Definition: An **r -combination** of elements of a set is an unordered selection of r elements from the set. Thus, an r -combination is simply a subset of the set with r elements.

- The number of r -combinations of a set with n distinct elements is denoted by $C(n, r)$. The notation $\binom{n}{r}$ is also used and is called a **binomial coefficient**. (*We will see the notation again in the binomial theorem in Section 6.4.*)

Example: Let S be the set $\{a, b, c, d\}$. Then $\{a, c, d\}$ is a 3-combination from S . It is the same as $\{d, c, a\}$ since the order listed does not matter.

- $C(4, 2) = 6$ because the 2-combinations of $\{a, b, c, d\}$ are the six subsets $\{a, b\}$, $\{a, c\}$, $\{a, d\}$, $\{b, c\}$, $\{b, d\}$, and $\{c, d\}$.

Combinations

Theorem 2: The number of *r-combinations* of a set with n elements, where $n \geq r \geq 0$, equals

$$C(n, r) = \frac{n!}{(n-r)!r!}.$$

Proof: By the product rule $P(n, r) = C(n, r) \cdot P(r, r)$.
Therefore,

$$C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!/(n-r)!}{r!/(r-r)!} = \frac{n!}{(n-r)!r!}.$$

Combinations

Example: How many poker hands of five cards can be dealt from a standard deck of 52 cards? Also, how many ways are there to select 47 cards from a deck of 52 cards?

Solution: Since the order in which the cards are dealt does not matter, the number of five card hands is:

$$\begin{aligned} C(52, 5) &= \frac{52!}{5!47!} \\ &= \frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 26 \cdot 17 \cdot 10 \cdot 49 \cdot 12 = 2,598,960 \end{aligned}$$

- The different ways to select 47 cards from 52 is

$$C(52, 47) = \frac{52!}{47!5!} = C(52, 5) = 2,598,960.$$

This is a special case of a general result. →

Combinations

Corollary 2: Let n and r be nonnegative integers with $r \leq n$.
Then $C(n, r) = C(n, n - r)$.

Proof: From Theorem 2, it follows that

$$C(n, r) = \frac{n!}{(n-r)!r!}$$

and

$$C(n, n - r) = \frac{n!}{(n-r)![n-(n-r)]!} = \frac{n!}{(n-r)!r!}.$$

Hence, $C(n, r) = C(n, n - r)$.



This result can be proved without using algebraic manipulation. →

Combinatorial Proofs

- **Definition 1:** A *combinatorial proof* of an identity is a proof that uses one of the following methods.
 - A *double counting proof* uses counting arguments to prove that both sides of an identity count the same objects, but in different ways.
 - A *bijective proof* shows that there is a bijection between the sets of objects counted by the two sides of the identity.

Combinatorial Proofs

- Here are two combinatorial proofs that

$$C(n, r) = C(n, n - r)$$

when r and n are nonnegative integers with $r < n$:

- *Bijjective Proof*: Suppose that S is a set with n elements. The function that maps a subset A of S to $\bar{A}$ is a bijection between the subsets of S with r elements and the subsets with $n - r$ elements. Since there is a bijection between the two sets, they must have the same number of elements. ◀
- *Double Counting Proof*: By definition the number of subsets of S with r elements is $C(n, r)$. Each subset A of S can also be described by specifying which elements are not in A , i.e., those which are in $\bar{A}$. Since the complement of a subset of S with r elements has $n - r$ elements, there are also $C(n, n - r)$ subsets of S with r elements. ◀

Combinations

Example: How many ways are there to select five players from a 10-member tennis team to make a trip to a match at another school.

Solution: By Theorem 2, the number of combinations is

$$C(10, 5) = \frac{10!}{5!5!} = 252.$$

Example: A group of 30 people have been trained as astronauts to go on the first mission to Mars. How many ways are there to select a crew of six people to go on this mission?

Solution: By Theorem 2, the number of possible crews is

$$C(30, 6) = \frac{30!}{6!24!} = \frac{30 \cdot 29 \cdot 28 \cdot 27 \cdot 26 \cdot 25}{6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 593,775 .$$

Binomial Coefficients and Identities

Section 6.4

Powers of Binomial Expressions

Definition: A **binomial expression** is the sum of two terms, such as $x + y$. (More generally, these terms can be products of constants and variables.)

- We can use **counting principles** to find the coefficients in the expansion of $(x + y)^n$ where n is a positive integer.
- To illustrate this idea, we first look at the process of expanding $(x + y)^3$.
- $(x + y)(x + y)(x + y)$ expands into a sum of terms that are the product of a term from each of the three sums.
- Terms of the form x^3 , x^2y , xy^2 , y^3 arise. The question is what are the coefficients?
 - To obtain x^3 , an x must be chosen from each of the sums. There is only one way to do this. So, the coefficient of x^3 is 1.
 - To obtain x^2y , an x must be chosen from two of the sums and a y from the other. There are $\binom{3}{2}$ ways to do this and so the coefficient of x^2y is 3.
 - To obtain xy^2 , an x must be chosen from **one** of the sums and a y from the other two. There are $\binom{3}{1}$ ways to do this and so the coefficient of xy^2 is 3.
 - To obtain y^3 , a y must be chosen from each of the sums. There is only one way to do this. So, the coefficient of y^3 is 1.
- We have used a counting argument to show that $(x + y)^3 = x^3 + 3x^2y + 3xy^2 + y^3$.
- Next we present the binomial theorem gives the coefficients of the terms in the expansion of $(x + y)^n$.

Binomial Theorem

Binomial Theorem: Let x and y be variables, and n a nonnegative integer. Then:

$$(x+y)^n = \sum_{j=0}^n \binom{n}{j} x^{n-j} y^j = \binom{n}{0} x^n + \binom{n}{1} x^{n-1} y + \cdots + \binom{n}{n-1} x y^{n-1} + \binom{n}{n} y^n.$$

Proof: We use combinatorial reasoning . The terms in the expansion of $(x + y)^n$ are of the form $x^{n-j}y^j$ for $j = 0, 1, 2, \dots, n$. To form the term $x^{n-j}y^j$, it is necessary to choose $n-j$ x s from the n sums. Therefore, the coefficient of $x^{n-j}y^j$ is $\binom{n}{n-j}$ which equals $\binom{n}{j}$. ◀

Using the Binomial Theorem

Example: What is the coefficient of $x^{12}y^{13}$ in the expansion of $(2x - 3y)^{25}$?

Solution: We view the expression as $(2x + (-3y))^{25}$.
By the binomial theorem

$$(2x + (-3y))^{25} = \sum_{j=0}^{25} \binom{25}{j} (2x)^{25-j} (-3y)^j.$$

Consequently, the coefficient of $x^{12}y^{13}$ in the expansion is obtained when $j = 13$.

$$\binom{25}{13} 2^{12} (-3)^{13} = -\frac{25!}{13!12!} 2^{12} 3^{13}.$$

Useful Identity:

Corollary 1: With $n \geq 0$,
$$\sum_{k=0}^n \binom{n}{k} = 2^n.$$

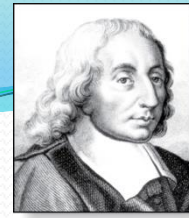
Proof (using binomial theorem): With $x = 1$ and $y = 1$, from the binomial theorem we see that:

$$2^n = (1 + 1)^n = \sum_{k=0}^n \binom{n}{k} 1^k 1^{(n-k)} = \sum_{k=0}^n \binom{n}{k}.$$

Proof (combinatorial): Consider the subsets of a set with n elements. There are $\binom{n}{0}$ subsets with zero elements, $\binom{n}{1}$ with one element, $\binom{n}{2}$ with two elements, ..., and $\binom{n}{n}$ with n elements. Therefore the total is
$$\sum_{k=0}^n \binom{n}{k}.$$

Since, we know that a set with n elements has 2^n subsets, we conclude:

$$\sum_{k=0}^n \binom{n}{k} = 2^n.$$



Pascal's Identity

Pascal's Identity: If n and k are integers with $n \geq k \geq 0$, then

$$\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}.$$

Proof (combinatorial): Let T be a set where $|T| = n + 1$, $a \in T$, and $S = T - \{a\}$. There are $\binom{n+1}{k}$ subsets of T containing k elements. Each of these subsets either:

- contains a with $k - 1$ other elements, or
- contains k elements of S and not a .

There are

- $\binom{n}{k-1}$ subsets of k elements that contain a , since there are $\binom{n}{k-1}$ subsets of $k - 1$ elements of S ,
- $\binom{n}{k}$ subsets of k elements of T that do not contain a , because there are $\binom{n}{k}$ subsets of k elements of S .

Hence, $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}.$ ◀

*See Exercise 19
for an algebraic
proof.*

Pascal's Triangle

The n th row in the triangle consists of the binomial coefficients $\binom{n}{k}$, $k = 0, 1, \dots, n$.

$$\begin{array}{c}
 \binom{0}{0} \\
 \binom{1}{0} \binom{1}{1} \\
 \binom{2}{0} \binom{2}{1} \binom{2}{2} \\
 \binom{3}{0} \binom{3}{1} \binom{3}{2} \binom{3}{3} \\
 \binom{4}{0} \binom{4}{1} \binom{4}{2} \binom{4}{3} \binom{4}{4} \\
 \binom{5}{0} \binom{5}{1} \binom{5}{2} \binom{5}{3} \binom{5}{4} \binom{5}{5} \\
 \binom{6}{0} \binom{6}{1} \binom{6}{2} \binom{6}{3} \binom{6}{4} \binom{6}{5} \binom{6}{6} \\
 \binom{7}{0} \binom{7}{1} \binom{7}{2} \binom{7}{3} \binom{7}{4} \binom{7}{5} \binom{7}{6} \binom{7}{7} \\
 \binom{8}{0} \binom{8}{1} \binom{8}{2} \binom{8}{3} \binom{8}{4} \binom{8}{5} \binom{8}{6} \binom{8}{7} \binom{8}{8} \\
 \dots \\
 \text{(a)}
 \end{array}$$

By Pascal's identity:

$$\binom{6}{4} + \binom{6}{5} = \binom{7}{5}$$

$$\begin{array}{c}
 1 \\
 1 \quad 1 \\
 1 \quad 2 \quad 1 \\
 1 \quad 3 \quad 3 \quad 1 \\
 1 \quad 4 \quad 6 \quad 4 \quad 1 \\
 1 \quad 5 \quad 10 \quad 10 \quad 5 \quad 1 \\
 1 \quad 6 \quad 15 \quad 20 \quad 15 \quad 6 \quad 1 \\
 1 \quad 7 \quad 21 \quad 35 \quad 35 \quad 21 \quad 7 \quad 1 \\
 1 \quad 8 \quad 28 \quad 56 \quad 70 \quad 56 \quad 28 \quad 8 \quad 1 \\
 \dots \\
 \text{(b)}
 \end{array}$$

By Pascal's identity, adding two adjacent binomial coefficients results in the binomial coefficient in the next row between these two coefficients.

Generalized Permutations and Combinations

Section 6.5

Permutations with Repetition

Theorem 1: The number of r -permutations of a set of n objects with repetition allowed is n^r .

Proof: There are n ways to select an element of the set for each of the r positions in the r -permutation when repetition is allowed. Hence, by the product rule there are n^r r -permutations with repetition. ◀

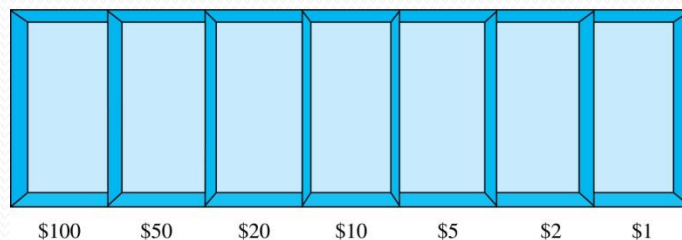
Example: How many strings of length r can be formed from the uppercase letters of the English alphabet?

Solution: The number of such strings is 26^r , which is the number of r -permutations of a set with 26 elements.

Combinations with Repetition

Example: How many ways are there to select five bills from a box containing at least five of each of the following denominations: \$1, \$2, \$5, \$10, \$20, \$50, and \$100?

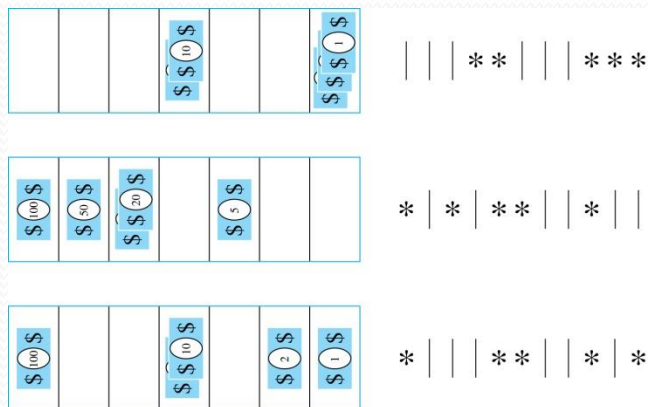
Solution: Place the selected bills in the appropriate position of a cash box illustrated below:



continued →

Combinations with Repetition

- Some possible ways of placing the five bills:



- The number of ways to select five bills corresponds to the number of ways to arrange six bars and five stars in a row.
- This is the number of unordered selections of 5 objects from a set of 11. Hence, there are

$$C(11, 5) = \frac{11!}{5!6!} = 462$$

ways to choose five bills with seven types of bills.

Combinations with Repetition

Theorem 2: The number of r -combinations from a set with n elements when **repetition** of elements is allowed is

$$C(n + r - 1, r) = C(n + r - 1, n - 1).$$

Proof: Each r -combination of a set with n elements with repetition allowed can be represented by a list of $n - 1$ bars and r stars. The bars mark the n cells containing a star for each time the i th element of the set occurs in the combination.

The number of such lists is $C(n + r - 1, r)$, because each list is a choice of the r positions to place the stars, from the total of $n + r - 1$ positions to place the stars and the bars. This is also equal to $C(n + r - 1, n - 1)$, which is the number of ways to place the $n - 1$ bars.



Combinations with Repetition

Example: How many solutions does the equation

$$x_1 + x_2 + x_3 = 11$$

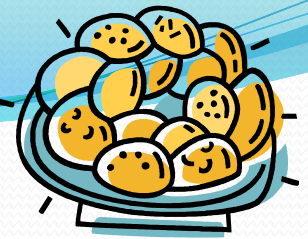
have, where x_1 , x_2 and x_3 are nonnegative integers?

Solution: Each solution corresponds to a way to select 11 items from a set with three elements; x_1 elements of type one, x_2 of type two, and x_3 of type three.

By Theorem 2 it follows that there are

$$C(3 + 11 - 1, 11) = C(13, 11) = C(13, 2) = \frac{13 \cdot 12}{1 \cdot 2} = 78$$

solutions.



Combinations with Repetition

Example: Suppose that a cookie shop has four different kinds of cookies. How many different ways can six cookies be chosen?

Solution: The number of ways to choose six cookies is the number of 6-combinations of a set with four elements. By Theorem 2

$$C(9, 6) = C(9, 3) = \frac{9 \cdot 8 \cdot 7}{1 \cdot 2 \cdot 3} = 84$$

is the number of ways to choose six cookies from the four kinds.

Summarizing the Formulas for Counting Permutations and Combinations with and without Repetition

TABLE 1 Combinations and Permutations With and Without Repetition.

<i>Type</i>	<i>Repetition Allowed?</i>	<i>Formula</i>
r -permutations	No	$\frac{n!}{(n-r)!}$
r -combinations	No	$\frac{n!}{r!(n-r)!}$
r -permutations	Yes	n^r
r -combinations	Yes	$\frac{(n+r-1)!}{r!(n-1)!}$



الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

Chapter 8

Advanced Counting Techniques

Chapter Outlines

Applications of Recurrence Relations

Solving Linear Recurrence Relations

Divide-and-Conquer Algorithms and Recurrence Relations

Applications of Recurrence Relations

Section 8.1

Recurrence Relations

(recalling definitions from Chapter 2)

Definition: A **recurrence relation** for the sequence $\{a_n\}$ is an equation that expresses a_n in terms of one or more of the previous terms of the sequence, namely, $a_0, a_1, \dots, a_{n-1}$, for all integers n with $n \geq n_0$, where n_0 is a nonnegative integer.

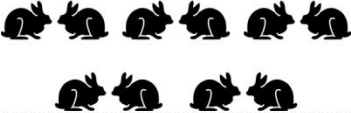
- A sequence is called a **solution** of a recurrence relation if its terms satisfy the recurrence relation.
- The **initial conditions** for a sequence specify the terms that precede the first term where the recurrence relation takes effect.

Rabbits and the Fibonacci Numbers

Example: A young pair of rabbits (one of each gender) is placed on an island. A pair of rabbits does not breed until they are 2 months old. After they are 2 months old, each pair of rabbits produces another pair each month. Find a recurrence relation for the number of pairs of rabbits on the island after n months, assuming that rabbits never die.

This is the original problem considered by Leonardo Pisano (Fibonacci) in the thirteenth century.

Rabbits and the Fibonacci Numbers (*cont.*)

Reproducing pairs (at least two months old)	Young pairs (less than two months old)	Month	Reproducing pairs	Young pairs	Total pairs
		1	0	1	1
		2	0	1	1
		3	1	1	2
		4	1	2	3
		5	2	3	5
		6	3	5	8

Modeling the Population Growth of Rabbits on an Island

Rabbits and the Fibonacci Numbers (*cont.*)

Solution: Let f_n be the the number of pairs of rabbits after n months.

- There are is $f_1 = 1$ pairs of rabbits on the island at the end of the first month.
- We also have $f_2 = 1$ because the pair does not breed during the first month.
- To find the number of pairs on the island after n months, add the number on the island after the previous month, f_{n-1} , and the number of newborn pairs, which equals f_{n-2} , because each newborn pair comes from a pair at least two months old.

Consequently the sequence $\{f_n\}$ satisfies the recurrence relation $f_n = f_{n-1} + f_{n-2}$ for $n \geq 3$ with the initial conditions $f_1 = 1$ and $f_2 = 1$. The number of pairs of rabbits on the island after n months is given by the n th Fibonacci number.

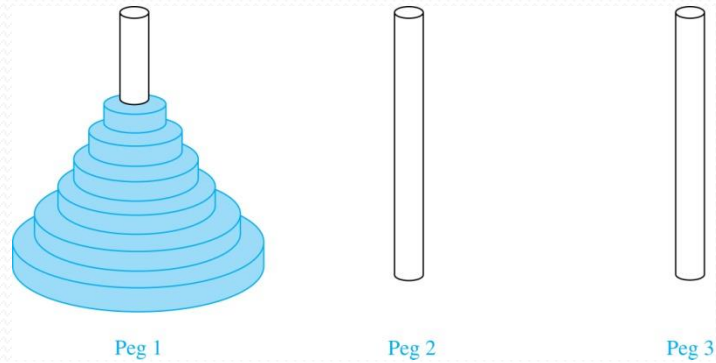
The Tower of Hanoi

In the late nineteenth century, the French mathematician Édouard Lucas invented a puzzle consisting of three pegs on a board with disks of different sizes. Initially all of the disks are on the first peg in order of size, with the largest on the bottom.

Rules: You are allowed to move the disks one at a time from one peg to another as long as a larger disk is never placed on a smaller.

Goal: Using allowable moves, end up with all the disks on the second peg in order of size with largest on the bottom.

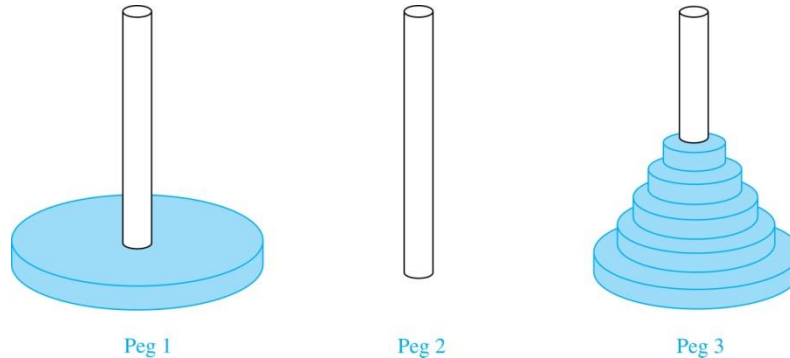
The Tower of Hanoi (*continued*)



The Initial Position in the Tower of Hanoi Puzzle

The Tower of Hanoi (*continued*)

Solution: Let $\{H_n\}$ denote the number of moves needed to solve the Tower of Hanoi Puzzle with n disks. Set up a recurrence relation for the sequence $\{H_n\}$. Begin with n disks on peg 1. We can transfer the top $n - 1$ disks, following the rules of the puzzle, to peg 3 using H_{n-1} moves.



First, we use 1 move to transfer the largest disk to the second peg. Then we transfer the $n - 1$ disks from peg 3 to peg 2 using H_{n-1} additional moves. This can not be done in fewer steps. Hence,

$$H_n = 2H_{n-1} + 1.$$

The initial condition is $H_1 = 1$ since a single disk can be transferred from peg 1 to peg 2 in one move.

The Tower of Hanoi (*continued*)

- We can use an iterative approach to solve this recurrence relation by repeatedly expressing H_n in terms of the previous terms of the sequence.

$$\begin{aligned}H_n &= 2H_{n-1} + 1 \\&= 2(2H_{n-2} + 1) + 1 = 2^2 H_{n-2} + 2 + 1 \\&= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3 H_{n-3} + 2^2 + 2 + 1 \\&\vdots \\&= 2^{n-1}H_1 + 2^{n-2} + 2^{n-3} + \dots + 2 + 1 \\&= 2^{n-1} + 2^{n-2} + 2^{n-3} + \dots + 2 + 1 \quad \text{because } H_1 = 1 \\&= 2^n - 1 \quad \text{using the formula for the sum of the terms of a geometric series}\end{aligned}$$

- There was a myth created with the puzzle. Monks in a tower in Hanoi are transferring 64 gold disks from one peg to another following the rules of the puzzle. They move one disk each day. When the puzzle is finished, the world will end.
- Using this formula for the 64 gold disks of the myth,
$$2^{64} - 1 = 18,446, 744,073, 709,551,615$$
days are needed to solve the puzzle, which is more than 500 billion years.
- Reve's puzzle (proposed in 1907 by Henry Dudeney) is similar but has 4 pegs. There is a well-known unsettled conjecture for the minimum number of moves needed to solve this puzzle. (see Exercises 38-45)

Counting Bit Strings

Example 3: Find a recurrence relation and give initial conditions for the number of bit strings of length n without two consecutive 0s. How many such bit strings are there of length five?

Solution: Let a_n denote the number of bit strings of length n without two consecutive 0s. To obtain a recurrence relation for $\{a_n\}$ note that the number of bit strings of length n that do not have two consecutive 0s is the number of bit strings ending with a 0 plus the number of such bit strings ending with a 1.

Now assume that $n \geq 3$.

- The bit strings of length n ending with 1 without two consecutive 0s are the bit strings of length $n-1$ with no two consecutive 0s with a 1 at the end. Hence, there are a_{n-1} such bit strings.
- The bit strings of length n ending with 0 without two consecutive 0s are the bit strings of length $n-2$ with no two consecutive 0s with 10 at the end. Hence, there are a_{n-2} such bit strings.

We conclude that $a_n = a_{n-1} + a_{n-2}$ for $n \geq 3$.

		Number of bit strings of length n with no two consecutive 0s:	
End with a 1:	Any bit string of length $n-1$ with no two consecutive 0s	1	a_{n-1}
End with a 0:	Any bit string of length $n-2$ with no two consecutive 0s	1 0	a_{n-2}
		Total:	$a_n = a_{n-1} + a_{n-2}$

Bit Strings (*continued*)

The initial conditions are:

- $a_1 = 2$, since both the bit strings 0 and 1 do not have consecutive 0s.
- $a_2 = 3$, since the bit strings 01, 10, and 11 do not have consecutive 0s, while 00 does.

To obtain a_5 , we use the recurrence relation three times to find that:

- $a_3 = a_2 + a_1 = 3 + 2 = 5$
- $a_4 = a_3 + a_2 = 5 + 3 = 8$
- $a_5 = a_4 + a_3 = 8 + 5 = 13$

Note that $\{a_n\}$ satisfies the same recurrence relation as the Fibonacci sequence. Since $a_1 = f_3$ and $a_2 = f_4$, we conclude that $a_n = f_{n+2}$.

Counting the Ways to Parenthesize a Product

Example: Find a recurrence relation for C_n , the number of ways to parenthesize the product of $n + 1$ numbers, $x_0 \cdot x_1 \cdot x_2 \cdot \cdots \cdot x_n$, to specify the order of multiplication.

For example, $C_3 = 5$, since all the possible ways to parenthesize 4 numbers are

$$((x_0 \cdot x_1) \cdot x_2) \cdot x_3, \quad (x_0 \cdot (x_1 \cdot x_2)) \cdot x_3, \quad (x_0 \cdot x_1) \cdot (x_2 \cdot x_3), \quad x_0 \cdot ((x_1 \cdot x_2) \cdot x_3), \quad x_0 \cdot (x_1 \cdot (x_2 \cdot x_3))$$

Solution: Note that however parentheses are inserted in $x_0 \cdot x_1 \cdot x_2 \cdot \cdots \cdot x_n$, one “ $\cdot$ ” operator remains outside all parentheses. This final operator appears between two of the $n + 1$ numbers, say x_k and x_{k+1} . Since there are C_k ways to insert parentheses in the product $x_0 \cdot x_1 \cdot x_2 \cdot \cdots \cdot x_k$ and C_{n-k-1} ways to insert parentheses in the product $x_{k+1} \cdot x_{k+2} \cdot \cdots \cdot x_n$, we have

$$\begin{aligned} C_n &= C_0 C_{n-1} + C_1 C_{n-2} + \cdots + C_{n-2} C_1 + C_{n-1} C_0 \\ &= \sum_{k=0}^{n-1} C_k C_{n-k-1} \end{aligned}$$

The initial conditions are $C_0 = 1$ and $C_1 = 1$.

The sequence $\{C_n\}$ is the sequence of **Catalan Numbers**. This recurrence relation can be solved using the method of **generating functions**; see Exercise 41 in Section 8.4.

Solving Linear Recurrence Relations

Section 8.2

Linear Homogeneous Recurrence Relations

Definition: A *linear homogeneous recurrence relation of degree k with constant coefficients* is a recurrence relation of the form $a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k}$, where $c_1, c_2, \dots, c_k$ are real numbers, and $c_k \neq 0$

- it is *linear* because the right-hand side is a sum of the previous terms of the sequence each multiplied by a function of n .
- it is *homogeneous* because no terms occur that are not multiples of the a_j s. Each coefficient is a constant.
- the *degree* is k because a_n is expressed in terms of the previous k terms of the sequence.

By strong induction, a sequence satisfying such a recurrence relation is uniquely determined by the recurrence relation and the k initial conditions $a_0 = C_0, a_1 = C_1, \dots, a_{k-1} = C_{k-1}$.

Examples of Linear Homogeneous Recurrence Relations

- $P_n = (1.11)P_{n-1}$ linear homogeneous recurrence relation of degree one
- $f_n = f_{n-1} + f_{n-2}$ linear homogeneous recurrence relation of degree two
- $a_n = a_{n-1} + a_{n-2}^2$ not linear
- $H_n = 2H_{n-1} + 1$ not homogeneous
- $B_n = nB_{n-1}$ coefficients are not constants

Solving Linear Homogeneous Recurrence Relations

- The basic approach is to look for solutions of the form $a_n = r^n$, where r is a constant.
- Note that $a_n = r^n$ is a solution to the recurrence relation $a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k}$ if and only if $r^n = c_1 r^{n-1} + c_2 r^{n-2} + \dots + c_k r^{n-k}$.
- Algebraic manipulation yields the *characteristic equation*:
$$r^k - c_1 r^{k-1} - c_2 r^{k-2} - \dots - c_{k-1} r - c_k = 0$$
- The sequence $\{a_n\}$ with $a_n = r^n$ is a solution if and only if r is a solution to the characteristic equation.
- The solutions to the characteristic equation are called the *characteristic roots* of the recurrence relation. The roots are used to give an explicit formula for all the solutions of the recurrence relation.

Solving Linear Homogeneous Recurrence Relations of Degree Two

Theorem 1: Let c_1 and c_2 be real numbers. Suppose that $r^2 - c_1r - c_2 = 0$ has two distinct roots r_1 and r_2 . Then the sequence $\{a_n\}$ is a solution to the recurrence relation $a_n = c_1a_{n-1} + c_2a_{n-2}$ if and only if

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n$$

for $n = 0, 1, 2, \dots$, where α_1 and α_2 are constants.

Using Theorem 1

Example: What is the solution to the recurrence relation

$$a_n = a_{n-1} + 2a_{n-2} \text{ with } a_0 = 2 \text{ and } a_1 = 7?$$

Solution: The characteristic equation is $r^2 - r - 2 = 0$.

Its roots are $r = 2$ and $r = -1$. Therefore, $\{a_n\}$ is a solution to the recurrence relation if and

only if $a_n = \alpha_1 2^n + \alpha_2 (-1)^n$, for some constants α_1 and α_2 .

To find the constants α_1 and α_2 , note that

$$a_0 = 2 = \alpha_1 + \alpha_2 \text{ and } a_1 = 7 = \alpha_1 2 + \alpha_2 (-1).$$

Solving these equations, we find that $\alpha_1 = 3$ and $\alpha_2 = -1$.

Hence, the solution is the sequence $\{a_n\}$ with $a_n = 3 \cdot 2^n - (-1)^n$.

An Explicit Formula for the Fibonacci Numbers

We can use Theorem 1 to find an explicit formula for the Fibonacci numbers. The sequence of Fibonacci numbers satisfies the recurrence relation $f_n = f_{n-1} + f_{n-2}$ with the initial conditions: $f_0 = 0$ and $f_1 = 1$.

Solution: The roots of the characteristic equation $r^2 - r - 1 = 0$ are

$$r_1 = \frac{1+\sqrt{5}}{2}$$

$$r_2 = \frac{1-\sqrt{5}}{2}$$

Fibonacci Numbers (*continued*)

Therefore by Theorem 1

$$f_n = \alpha_1 \left(\frac{1+\sqrt{5}}{2} \right)^n + \alpha_2 \left(\frac{1-\sqrt{5}}{2} \right)^n$$

for some constants α_1 and α_2 .

Using the initial conditions $f_0 = 0$ and $f_1 = 1$, we have

$$f_0 = \alpha_1 + \alpha_2 = 0$$

$$f_1 = \alpha_1 \left(\frac{1+\sqrt{5}}{2} \right) + \alpha_2 \left(\frac{1-\sqrt{5}}{2} \right) = 1.$$

Solving, we obtain $\alpha_1 = \frac{1}{\sqrt{5}}$, $\alpha_2 = -\frac{1}{\sqrt{5}}$.

Hence,

$$f_n = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n - \frac{1}{\sqrt{5}} \left(\frac{1-\sqrt{5}}{2} \right)^n$$

The Solution when there is a Repeated Root

Theorem 2: Let c_1 and c_2 be real numbers with $c_2 \neq 0$. Suppose that $r^2 - c_1r - c_2 = 0$ has one repeated root r_0 . Then the sequence $\{a_n\}$ is a solution to the recurrence relation $a_n = c_1a_{n-1} + c_2a_{n-2}$ if and only if

$$a_n = \alpha_1 r_0^n + \alpha_2 n r_0^n$$

for $n = 0, 1, 2, \dots$, where α_1 and α_2 are constants.

Using Theorem 2

Example: What is the solution to the recurrence relation $a_n = 6a_{n-1} - 9a_{n-2}$ with $a_0 = 1$ and $a_1 = 6$?

Solution: The characteristic equation is $r^2 - 6r + 9 = 0$.

The only root is $r = 3$. Therefore, $\{a_n\}$ is a solution to the recurrence relation if and only if

$$a_n = \alpha_1 3^n + \alpha_2 n(3)^n$$

where α_1 and α_2 are constants.

To find the constants α_1 and α_2 , note that

$$a_0 = 1 = \alpha_1 \quad \text{and} \quad a_1 = 6 = \alpha_1 \cdot 3 + \alpha_2 \cdot 3.$$

Solving, we find that $\alpha_1 = 1$ and $\alpha_2 = 1$.

Hence,

$$a_n = 3^n + n3^n.$$

Solving Linear Homogeneous Recurrence Relations of Arbitrary Degree

This theorem can be used to solve linear homogeneous recurrence relations with constant coefficients of any degree when the characteristic equation has distinct roots.

Theorem 3: Let $c_1, c_2, \dots, c_k$ be real numbers. Suppose that the characteristic equation

$$r^k - c_1 r^{k-1} - \dots - c_k = 0$$

has k distinct roots $r_1, r_2, \dots, r_k$. Then a sequence $\{a_n\}$ is a solution of the recurrence relation

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k}$$

if and only if

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n + \dots + \alpha_k r_k^n$$

for $n = 0, 1, 2, \dots$, where $\alpha_1, \alpha_2, \dots, \alpha_k$ are constants.

The General Case with Repeated Roots Allowed

Theorem 4: Let $c_1, c_2, \dots, c_k$ be real numbers. Suppose that the characteristic equation

$$r^k - c_1 r^{k-1} - \dots - c_k = 0$$

has t distinct roots $r_1, r_2, \dots, r_t$ with multiplicities $m_1, m_2, \dots, m_t$, respectively so that $m_i \geq 1$ for $i = 1, 2, \dots, t$ and $m_1 + m_2 + \dots + m_t = k$. Then a sequence $\{a_n\}$ is a solution of the recurrence relation

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k}$$

if and only if

$$\begin{aligned} a_n = & (\alpha_{1,0} + \alpha_{1,1}n + \dots + \alpha_{1,m_1-1}n^{m_1-1})r_1^n \\ & + (\alpha_{2,0} + \alpha_{2,1}n + \dots + \alpha_{2,m_2-1}n^{m_2-1})r_2^n \\ & + \dots + (\alpha_{t,0} + \alpha_{t,1}n + \dots + \alpha_{t,m_t-1}n^{m_t-1})r_t^n \end{aligned}$$

for $n = 0, 1, 2, \dots$, where $\alpha_{i,j}$ are constants for $1 \leq i \leq t$ and $0 \leq j \leq m_i - 1$.

Linear Nonhomogeneous Recurrence Relations with Constant Coefficients

Definition: A *linear nonhomogeneous recurrence relation with constant coefficients* is a recurrence relation of the form:

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k} + F(n),$$

where $c_1, c_2, \dots, c_k$ are real numbers, and $F(n)$ is a function not identically zero depending only on n .

The recurrence relation

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k},$$

is called the *associated* homogeneous recurrence relation.

Linear Nonhomogeneous Recurrence Relations with Constant Coefficients (*cont.*)

The following are **linear nonhomogeneous** recurrence relations with constant coefficients:

$$a_n = a_{n-1} + 2^n,$$

$$a_n = a_{n-1} + a_{n-2} + n^2 + n + 1,$$

$$a_n = 3a_{n-1} + n3^n,$$

$$a_n = a_{n-1} + a_{n-2} + a_{n-3} + n!$$

where the following are the associated linear homogeneous recurrence relations, respectively:

$$a_n = a_{n-1},$$

$$a_n = a_{n-1} + a_{n-2},$$

$$a_n = 3a_{n-1},$$

$$a_n = a_{n-1} + a_{n-2} + a_{n-3}$$

Solving Linear Nonhomogeneous Recurrence Relations with Constant Coefficients

Theorem 5: If $\{a_n^{(p)}\}$ is a particular solution of the nonhomogeneous linear recurrence relation with constant coefficients

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k} + F(n),$$

then every solution is of the form $\{a_n^{(p)} + a_n^{(h)}\}$, where $\{a_n^{(h)}\}$ is a solution of the associated homogeneous recurrence relation

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k}.$$

Solving Linear Nonhomogeneous Recurrence Relations with Constant Coefficients (*continued*)

Example: Find all solutions of the recurrence relation $a_n = 3a_{n-1} + 2n$.
What is the solution with $a_1 = 3$?

Solution: The associated linear homogeneous equation is $a_n = 3a_{n-1}$.
Its solutions are $a_n^{(h)} = \alpha 3^n$, where α is a constant.

Because $F(n) = 2n$ is a polynomial in n of degree one, to find a particular solution we might try a linear function in n , say $p_n = cn + d$, where c and d are constants. Suppose that $p_n = cn + d$ is such a solution.

Then $a_n = 3a_{n-1} + 2n$ becomes $cn + d = 3(c(n-1) + d) + 2n$.

Simplifying yields $(2 + 2c)n + (2d - 3c) = 0$. It follows that $cn + d$ is a solution if and only if $2 + 2c = 0$ and $2d - 3c = 0$. Therefore, $cn + d$ is a solution if and only if $c = -1$ and $d = -3/2$.
Consequently, $a_n^{(p)} = -n - 3/2$ is a particular solution.

By Theorem 5, all solutions are of the form $a_n = a_n^{(p)} + a_n^{(h)} = -n - 3/2 + \alpha 3^n$, where α is a constant.

To find the solution with $a_1 = 3$, let $n = 1$ in the above formula for the general solution.
Then $3 = -1 - 3/2 + 3\alpha$, and $\alpha = 11/6$. Hence, the solution is $a_n = -n - 3/2 + (11/6)3^n$.

Divide-and-Conquer Algorithms and Recurrence Relations

Section 8.3

Divide-and-Conquer Algorithmic Paradigm

Definition: A *divide-and-conquer algorithm* works by first *dividing* a problem into one or more instances of the same problem of smaller size and then *conquering* the problem using the solutions of the smaller problems to find a solution of the original problem.

Examples:

- Binary search, covered in Chapters 3 and 5: It works by comparing the element to be located to the middle element. The original list is then split into two lists and the search continues recursively in the appropriate sublist.
- Merge sort, covered in Chapter 5: A list is split into two approximately equal sized sublists, each recursively sorted by merge sort. Sorting is done by successively merging pairs of lists.

Divide-and-Conquer Recurrence Relations

- Suppose that a recursive algorithm divides a problem of size n into a subproblems.
- Assume each subproblem is of size n/b .
- Suppose $g(n)$ extra operations are needed in the conquer step.
- Then $f(n)$ represents the number of operations to solve a problem of size n satisfies the following recurrence relation:

$$f(n) = af(n/b) + g(n)$$

- This is called a *divide-and-conquer recurrence relation*.

Example: Binary Search

- Binary search reduces the search for an element in a sequence of size n to the search in a sequence of size $n/2$. Two comparisons are needed to implement this reduction;
 - one to decide whether to search the upper or lower half of the sequence and
 - the other to determine if the sequence has elements.
- Hence, if $f(n)$ is the number of comparisons required to search for an element in a sequence of size n , then

$$f(n) = f(n/2) + 2$$

when n is even.

Example

Let $f(n) = 3.f(n/2) + 9$, $f(4) = 8$, then find the value of $f(16)$.

Here $f(n) = 3f(n/2) + 9, f(4) = 8$ -----(i)

Put $n = 16$ in (i), we get

$$f(16) = 3f(8) + 9 \text{ -----(ii)}$$

Now put $n = 8$ in (i), we get

$$f(8) = 3f(8/2) + 9 = 3f(4) + 9 = 3.(8) + 9 = 33.$$

Using this value in (ii), we get

$$f(16) = 3f(8) + 9 = 3.(33) + 9 = 108.$$



الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

Chapter 9

Relations

Chapter Outlines

Relations and Their Properties

Representing Relations

Equivalence Relations

Partial Orderings

Relations and Their Properties

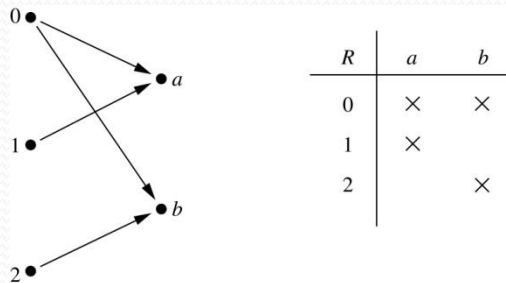
Section 9.1

Binary Relations

Definition: A **binary relation** R from a set A to a set B is a subset $R \subseteq A \times B$.

Example:

- Let $A = \{0,1,2\}$ and $B = \{a,b\}$
- $\{(0, a), (0, b), (1, a), (2, b)\}$ is a relation from A to B .
- We can represent relations from a set A to a set B **graphically** or using a **table**:



Relations are more general than functions. A function is a relation where exactly one element of B is related to each element of A .

Binary Relation on a Set

Definition: A **binary relation** R on a set A is a subset of $A \times A$ or a **relation** from A to A .

Example:

- Suppose that $A = \{a, b, c\}$. Then $R = \{(a, a), (a, b), (a, c)\}$ is a relation on A .
- Let $A = \{1, 2, 3, 4\}$. The **ordered pairs** in the relation $R = \{(a, b) \mid a \text{ divides } b\}$ are $(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 4), (3, 3)$, and $(4, 4)$.

Binary Relation on a Set (cont.)

Question: How many **relations** are there on a set A ?

Solution: Because a relation on A is the same thing as a **subset of $A \times A$** , we count the subsets of $A \times A$. Since $A \times A$ has n^2 elements when A has n elements, and a set with m elements has 2^m subsets, there are $2^{|A|^2}$ subsets of $A \times A$. Therefore, there are $2^{|A|^2}$ relations on a set A .

Binary Relations on a Set (cont.)

Example: Consider these relations on the set of integers:

$$R_1 = \{(a,b) \mid a \leq b\},$$

$$R_4 = \{(a,b) \mid a = b\},$$

$$R_2 = \{(a,b) \mid a > b\},$$

$$R_5 = \{(a,b) \mid a = b + 1\},$$

$$R_3 = \{(a,b) \mid a = b \text{ or } a = -b\},$$

$$R_6 = \{(a,b) \mid a + b \leq 3\}.$$

Note that these relations are on an infinite set and each of these relations is an infinite set.

Which of these relations contain each of the pairs

$(1,1)$, $(1, 2)$, $(2, 1)$, $(1, -1)$, and $(2, 2)$?

Solution: Checking the conditions that define each relation, we see that the pair $(1,1)$ is in R_1 , R_3 , R_4 , and R_6 ; $(1,2)$ is in R_1 and R_6 ; $(2,1)$ is in R_2 , R_5 , and R_6 ; $(1, -1)$ is in R_2 , R_3 , and R_6 ; $(2,2)$ is in R_1 , R_3 , and R_4 .

Reflexive Relations

Definition: R is **reflexive** iff $(a,a) \in R$ for every element $a \in A$.
Written symbolically, R is reflexive if and only if

$$\forall x[x \in U \rightarrow (x,x) \in R]$$

Example: The following relations on the integers **are reflexive**:

$$R_1 = \{(a,b) \mid a \leq b\},$$

$$R_3 = \{(a,b) \mid a = b \text{ or } a = -b\},$$

$$R_4 = \{(a,b) \mid a = b\}.$$

If $A = \emptyset$ then the empty relation is reflexive vacuously. That is the empty relation on an empty set is reflexive!

The following relations are **not reflexive**:

$$R_2 = \{(a,b) \mid a > b\} \text{ (note that } 3 \not> 3\text{),}$$

$$R_5 = \{(a,b) \mid a = b + 1\} \text{ (note that } 3 \neq 3 + 1\text{),}$$

$$R_6 = \{(a,b) \mid a + b \leq 3\} \text{ (note that } 4 + 4 \not\leq 3\text{).}$$

Symmetric Relations

Definition: R is **symmetric** iff $(b,a) \in R$ whenever $(a,b) \in R$ for all $a,b \in A$. Written symbolically, R is **symmetric** if and only if

$$\forall x \forall y [(x,y) \in R \rightarrow (y,x) \in R]$$

Example: The following relations on the integers **are symmetric**:

$$R_3 = \{(a,b) \mid a = b \text{ or } a = -b\},$$

$$R_4 = \{(a,b) \mid a = b\},$$

$$R_6 = \{(a,b) \mid a + b \leq 3\}.$$

The following are **not symmetric**:

$$R_1 = \{(a,b) \mid a \leq b\} \text{ (note that } 3 \leq 4, \text{ but } 4 \not\leq 3),$$

$$R_2 = \{(a,b) \mid a > b\} \text{ (note that } 4 > 3, \text{ but } 3 \not> 4),$$

$$R_5 = \{(a,b) \mid a = b + 1\} \text{ (note that } 4 = 3 + 1, \text{ but } 3 \neq 4 + 1).$$

Antisymmetric Relations

Definition: A relation R on a set A such that for all $a, b \in A$ if $(a, b) \in R$ and $(b, a) \in R$, then $a = b$ is called **antisymmetric**. Written symbolically, R is antisymmetric if and only if

$$\forall x \forall y [(x, y) \in R \wedge (y, x) \in R \rightarrow x = y]$$

- **Example:** The following relations on the integers are **antisymmetric**:

$$R_1 = \{(a, b) \mid a \leq b\},$$

$$R_2 = \{(a, b) \mid a > b\},$$

$$R_4 = \{(a, b) \mid a = b\},$$

$$R_5 = \{(a, b) \mid a = b + 1\}.$$

← For any integer, if $a \leq b$ and $a \geq b$, then $a = b$.

The following relations are **not antisymmetric**:

$$R_3 = \{(a, b) \mid a = b \text{ or } a = -b\}$$

(note that both $(1, -1)$ and $(-1, 1)$ belong to R_3),

$$R_6 = \{(a, b) \mid a + b \leq 3\} \text{ (note that both } (1, 2) \text{ and } (2, 1) \text{ belong to } R_6).$$

Transitive Relations

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

$$\forall x \forall y \forall z [(x,y) \in R \wedge (y,z) \in R \rightarrow (x,z) \in R]$$

- **Example:** The following relations on the integers **are transitive**:

$$R_1 = \{(a,b) \mid a \leq b\}, \quad \leftarrow$$

$$R_2 = \{(a,b) \mid a > b\},$$

$$R_3 = \{(a,b) \mid a = b \text{ or } a = -b\},$$

$$R_4 = \{(a,b) \mid a = b\}.$$

For every integer, $a \leq b$
and $b \leq c$, then $b \leq c$.

The following are **not transitive**:

$$R_5 = \{(a,b) \mid a = b + 1\} \text{ (note that both } (3,2) \text{ and } (4,3) \text{ belong to } R_5, \text{ but not } (3,3)),$$

$$R_6 = \{(a,b) \mid a + b \leq 3\} \text{ (note that both } (2,1) \text{ and } (1,2) \text{ belong to } R_6, \text{ but not } (2,2)).$$

Combining Relations

- Given two relations R_1 and R_2 , we can **combine** them using **basic set operations** to form new relations such as $R_1 \cup R_2$, $R_1 \cap R_2$, $R_1 - R_2$, and $R_2 - R_1$.
- Example:** Let $A = \{1,2,3\}$ and $B = \{1,2,3,4\}$. The relations $R_1 = \{(1,1),(2,2),(3,3)\}$ and $R_2 = \{(1,1),(1,2),(1,3),(1,4)\}$ can be combined using basic set operations to form new relations:

$$R_1 \cup R_2 = \{(1,1),(1,2),(1,3),(1,4),(2,2),(3,3)\}$$

$$R_1 \cap R_2 = \{(1,1)\} \qquad R_1 - R_2 = \{(2,2),(3,3)\}$$

$$R_2 - R_1 = \{(1,2),(1,3),(1,4)\}$$

Composition

Definition: Suppose

- R_1 is a relation from a set A to a set B .
- R_2 is a relation from B to a set C .

Then the **composition** (or *composite*) of R_2 with R_1 , is a **relation** from A to C where

- if (x,y) is a member of R_1 and (y,z) is a member of R_2 , then (x,z) is a member of $R_2 \circ R_1$.

Composition

What is the **composite** of the relations R and S , where R is the relation from $\{1, 2, 3\}$ to $\{1, 2, 3, 4\}$

with $R = \{(1, 1), (1, 4), (2, 3), (3, 1), (3, 4)\}$ and S is the relation from $\{1, 2, 3, 4\}$ to $\{0, 1, 2\}$

with $S = \{(1, 0), (2, 0), (3, 1), (3, 2), (4, 1)\}$?

Solution: $S \circ R$ is constructed using all ordered pairs in R and ordered pairs in S , where the second element of the ordered pair in R agrees with the first element of the ordered pair in S . For example, the ordered pairs $(2, 3)$ in R and $(3, 1)$ in S produce the ordered pair $(2, 1)$ in $S \circ R$. Computing all the ordered pairs in the composite, we find $S \circ R = \{(1, 0), (1, 1), (2, 1), (2, 2), (3, 0), (3, 1)\}$.

Powers of a Relation

Definition: Let R be a binary relation on A . Then the **powers** R^n of the relation R can be defined inductively by:

- **Basis Step:** $R^1 = R$
- **Inductive Step:** $R^{n+1} = R^n \circ R$

(see the slides for Section 9.3 for further insights)

The powers of a transitive relation are subsets of the relation. This is established by the following theorem:

Theorem 1: The relation R on a set A is **transitive** iff $R^n \subseteq R$ for $n = 1, 2, 3, \dots$

(see the text for a proof via mathematical induction)

Representing Relations

Section 9.3

Representing Relations Using Matrices

- A relation between finite sets can be represented using a **zero-one matrix**.
- Suppose R is a relation from $A = \{a_1, a_2, \dots, a_m\}$ to $B = \{b_1, b_2, \dots, b_n\}$.
 - The elements of the two sets can be listed in any particular arbitrary order. When $A = B$, we use the same ordering.
- The relation R is represented by the matrix $M_R = [m_{ij}]$, where

$$m_{ij} = \begin{cases} 1 & \text{if } (a_i, b_j) \in R, \\ 0 & \text{if } (a_i, b_j) \notin R. \end{cases}$$

- The matrix representing R has a 1 as its (i,j) entry when a_i is related to b_j and a 0 if a_i is not related to b_j .

Examples of Representing Relations Using Matrices

Example 1: Suppose that $A = \{1,2,3\}$ and $B = \{1,2\}$. Let R be the relation from A to B containing (a,b) if $a \in A$, $b \in B$, and $a > b$. What is the matrix representing R (assuming the ordering of elements is the same as the increasing numerical order)?

Solution: Because $R = \{(2,1), (3,1), (3,2)\}$, the matrix is

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

Examples of Representing Relations Using Matrices (*cont.*)

Example 2: Let $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3, b_4, b_5\}$. Which ordered pairs are in the relation R represented by the matrix

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

Solution: Because R consists of those ordered pairs (a_i, b_j) with $m_{ij} = 1$, it follows that:

$$R = \{(a_1, b_2), (a_2, b_1), (a_2, b_3), (a_2, b_4), (a_3, b_1), (a_3, b_3), (a_3, b_5)\}.$$

Matrices of Relations on Sets

- If R is a **reflexive relation**, all the elements on the **main diagonal** of M_R are **equal** to 1.

$$\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

- R is a **symmetric relation**, if and only if $m_{ij} = 1$ whenever $m_{ji} = 1$. R is an **antisymmetric relation**, if and only if $m_{ij} = 0$ or $m_{ji} = 0$ when $i \neq j$.

$$\begin{bmatrix} & & 1 \\ & \diagdown & \\ 1 & & \\ & \diagup & \\ & 0 & \end{bmatrix}$$

(a) Symmetric

$$\begin{bmatrix} & & 1 & 0 \\ & \diagdown & & \\ 0 & & & \\ & \diagup & & \\ & 0 & 1 & \\ & & & 0 \end{bmatrix}$$

(b) Antisymmetric

Example of a Relation on a Set

Example 3: Suppose that the relation R on a set is represented by the matrix

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

Is R reflexive, symmetric, and/or antisymmetric?

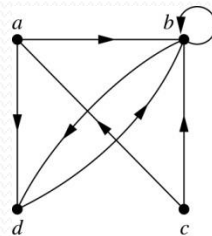
Solution: Because all the **diagonal elements** are equal to 1, R is **reflexive**. Because M_R is **symmetric**, R is **symmetric** and **not antisymmetric** because both $m_{1,2}$ and $m_{2,1}$ are 1.

Representing Relations Using Digraphs

Definition: A **directed graph**, or **digraph**, consists of a set V of **vertices (or nodes)** together with a set E of ordered pairs of elements of V called **edges (or arcs)**. The vertex a is called the **initial vertex** of the edge (a,b) , and the vertex b is called the **terminal vertex** of this edge.

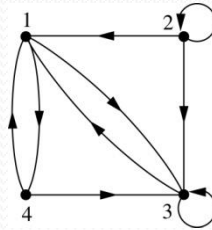
- An edge of the form (a,a) is called a **loop**.

Example 7: A drawing of the directed graph with vertices a, b, c , and d , and edges (a, b) , (a, d) , (b, b) , (b, d) , (c, a) , (c, b) , and (d, b) is shown here.



Examples of Digraphs Representing Relations

Example 8: What are the ordered pairs in the relation represented by this directed graph?

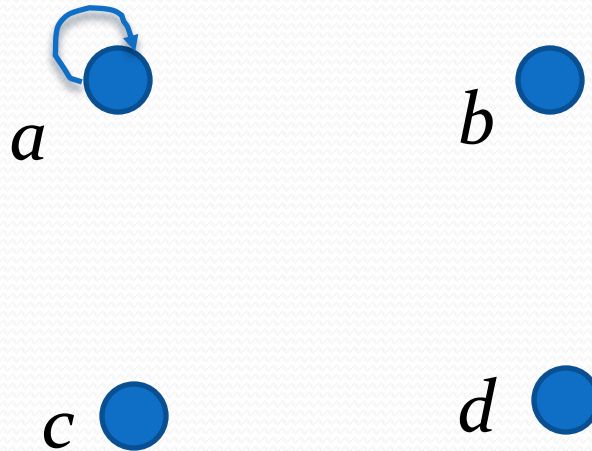


Solution: The ordered pairs in the relation are $(1, 3)$, $(1, 4)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(3, 1)$, $(3, 3)$, $(4, 1)$ and $(4, 3)$

Determining which Properties a Relation has from its Digraph

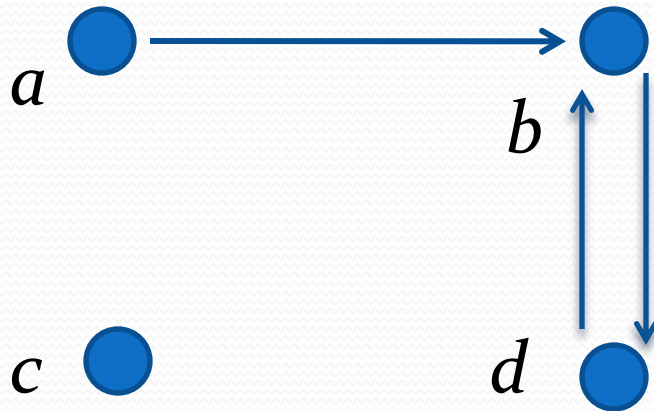
- **Reflexivity**: A loop must be present at all vertices in the graph.
- **Symmetry**: If (x,y) is an edge, then so is (y,x) .
- **Antisymmetry**: If (x,y) with $x \neq y$ is an edge, then (y,x) is not an edge.
- **Transitivity**: If (x,y) and (y,z) are edges, then so is (x,z) .

Determining which Properties a Relation has from its Digraph – Example 1



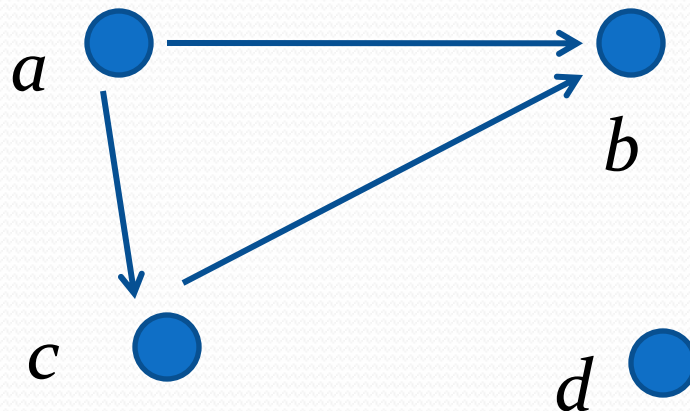
- *Reflexive?* No, not every vertex has a loop
- *Symmetric?* Yes (trivially), there is no edge from one vertex to another
- *Antisymmetric?* Yes (trivially), there is no edge from one vertex to another
- *Transitive?* Yes, (trivially) since there is no edge from one vertex to another

Determining which Properties a Relation has from its Digraph – Example 2



- *Reflexive*? No, there are no loops
- *Symmetric*? No, there is an edge from a to b , but not from b to a
- *Antisymmetric*? No, there is an edge from d to b and b to d
- *Transitive*? No, there are edges from a to b and from b to d , but there is no edge from a to d

Determining which Properties a Relation has from its Digraph – Example 3



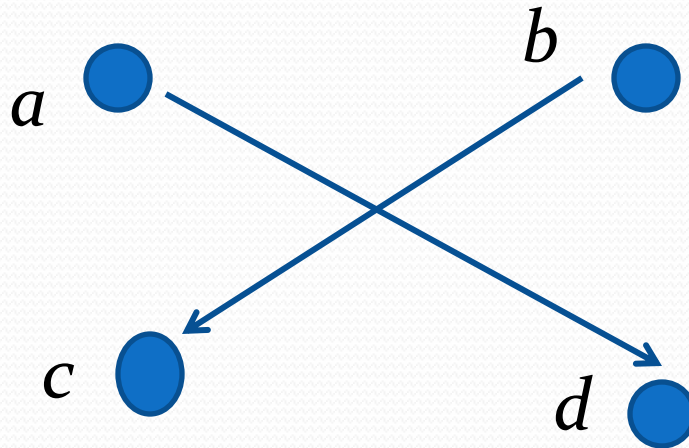
Reflexive? No, there are no loops

Symmetric? No, for example, there is no edge from c to a

Antisymmetric? Yes, whenever there is an edge from one vertex to another, there is not one going back

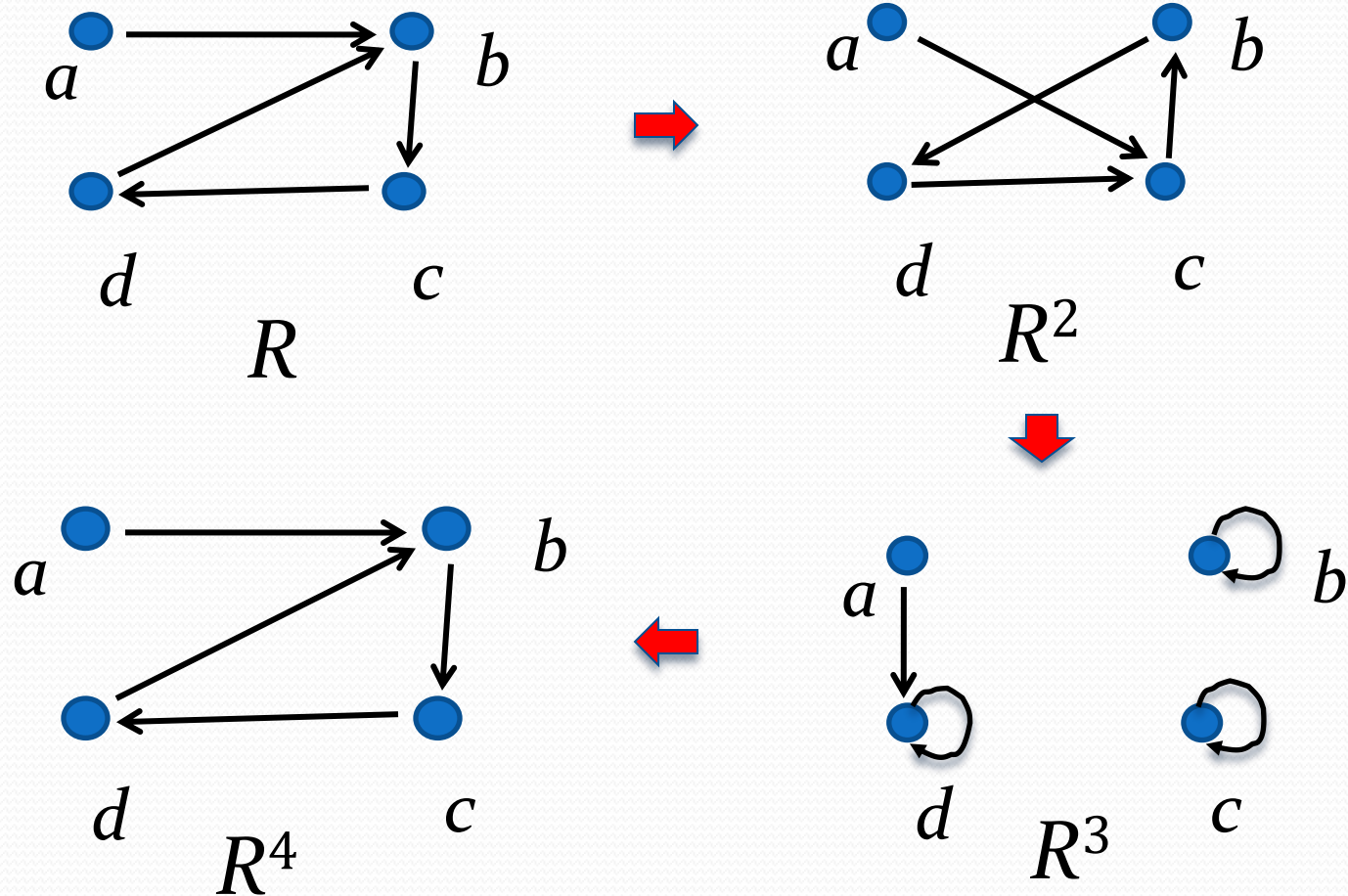
Transitive? yes, there is edge from a to c , and from c to b , then we have from a to b .

Determining which Properties a Relation has from its Digraph – Example 4



- *Reflexive*? No, there are no loops
- *Symmetric*? No, for example, there is no edge from d to a
- *Antisymmetric*? Yes, whenever there is an edge from one vertex to another, there is not one going back
- *Transitive*? Yes (trivially), there are no two edges where the first edge ends at the vertex where the second edge begins

Example of the Powers of a Relation



The pair (x,y) is in R^n if there is a path of length n from x to y in R (following the direction of the arrows).

Equivalence Relations

Section 9.5

Equivalence Relations

Definition 1: A relation on a set A is called an **equivalence relation** if it is reflexive, symmetric, and transitive.

Definition 2: Two elements a , and b that are related by an equivalence relation are called **equivalent**. The notation $a \sim b$ is often used to denote that a and b are equivalent elements with respect to a particular equivalence relation.

Strings

Example: Suppose that R is the relation on the set of strings of English letters such that aRb if and only if $l(a) = l(b)$, where $l(x)$ is the length of the string x . Is R an equivalence relation?

Solution: Show that all of the properties of an equivalence relation hold.

- *Reflexivity:* Because $l(a) = l(a)$, it follows that aRa for all strings a .
- *Symmetry:* Suppose that aRb . Since $l(a) = l(b)$, $l(b) = l(a)$ also holds and bRa .
- *Transitivity:* Suppose that aRb and bRc . Since $l(a) = l(b)$, and $l(b) = l(c)$, $l(a) = l(c)$ also holds and aRc .

Congruence Modulo m

Example: Let m be an integer with $m > 1$. Show that the relation

$$R = \{(a,b) \mid a \equiv b \pmod{m}\}$$

is an **equivalence relation** on the set of integers.

Solution: Recall that $a \equiv b \pmod{m}$ if and only if m divides $a - b$.

- *Reflexivity:* $a \equiv a \pmod{m}$ since $a - a = 0$ is divisible by m since $0 = 0 \cdot m$.
- *Symmetry:* Suppose that $a \equiv b \pmod{m}$. Then $a - b$ is divisible by m , and so $a - b = km$, where k is an integer. It follows that $b - a = (-k)m$, so $b \equiv a \pmod{m}$.
- *Transitivity:* Suppose that $a \equiv b \pmod{m}$ and $b \equiv c \pmod{m}$. Then m divides both $a - b$ and $b - c$. Hence, there are integers k and l with $a - b = km$ and $b - c = lm$. We obtain by adding the equations:

$$a - c = (a - b) + (b - c) = km + lm = (k + l)m.$$

Therefore, $a \equiv c \pmod{m}$.

Divides

Example: Show that the “**divides**” relation on the set of positive integers is not an equivalence relation.

Solution: The properties of reflexivity, and transitivity do hold, but there relation is not transitive. Hence, “**divides**” is not an equivalence relation.

- **Reflexivity:** $a \mid a$ for all a .
- **Not Symmetric:** For example, $2 \mid 4$, but $4 \nmid 2$. Hence, the relation is not symmetric.
- **Transitivity:** Suppose that a divides b and b divides c . Then there are positive integers k and l such that $b = ak$ and $c = bl$. Hence, $c = a(kl)$, so a divides c . Therefore, the relation is transitive.

Equivalence Classes

Definition 3: Let R be an equivalence relation on a set A . The set of all elements that are related to an element a of A is called the *equivalence class* of a . The equivalence class of a with respect to R is denoted by $[a]_R$.

When only one relation is under consideration, we can write $[a]$, without the subscript R , for this equivalence class.

Note that $[a]_R = \{s | (a,s) \in R\}$.

- If $b \in [a]_R$, then b is called a representative of this equivalence class. Any element of a class can be used as a representative of the class.
- The equivalence classes of the relation congruence modulo m are called the *congruence classes modulo m* . The congruence class of an integer a modulo m is denoted by $[a]_m$, so $[a]_m = \{\dots, a-2m, a-m, a, a+m, a+2m, \dots\}$. For example,

$$[0]_4 = \{\dots, -8, -4, 0, 4, 8, \dots\}$$

$$[1]_4 = \{\dots, -7, -3, 1, 5, 9, \dots\}$$

$$[2]_4 = \{\dots, -6, -2, 2, 6, 10, \dots\}$$

$$[3]_4 = \{\dots, -5, -1, 3, 7, 11, \dots\}$$

Equivalence Classes and Partitions

Theorem 1: let R be an equivalence relation on a set A . These statements for elements a and b of A are equivalent:

(i) aRb

(ii) $[a] = [b]$

(iii) $[a] \cap [b] = \emptyset$

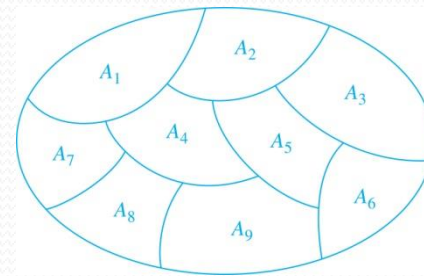
Proof: We show that (i) implies (ii). Assume that aRb . Now suppose that $c \in [a]$. Then aRc . Because aRb and R is symmetric, bRa . Because R is transitive and bRa and aRc , it follows that bRc . Hence, $c \in [b]$. Therefore, $[a] \subseteq [b]$. A similar argument (omitted here) shows that $[b] \subseteq [a]$. Since $[a] \subseteq [b]$ and $[b] \subseteq [a]$, we have shown that $[a] = [b]$.

(see text for proof that (ii) implies (iii) and (iii) implies (i))

Partition of a Set

Definition: A **partition** of a set S is a collection of **disjoint nonempty subsets** of S that have S as their union. In other words, the collection of subsets A_i , where $i \in I$ (where I is an index set), forms a partition of S if and only if

- $A_i \neq \emptyset$ for $i \in I$,
- $A_i \cap A_j = \emptyset$ when $i \neq j$,
- and $\bigcup_{i \in I} A_i = S$.



A Partition of a Set

An Equivalence Relation Partitions a Set

- Let R be an equivalence relation on a set A . The **union** of all the equivalence classes of R is all of A , since an element a of A is in its own equivalence class $[a]_R$. In other words,

$$\bigcup_{a \in A} [a]_R = A.$$

- From Theorem 1, it follows that these equivalence classes are either **equal** or **disjoint**, so $[a]_R \cap [b]_R = \emptyset$ when $[a]_R \neq [b]_R$.
- Therefore, the equivalence classes form a partition of A , because they split A into **disjoint subsets**.

An Equivalence Relation Partitions a Set (continued)

Theorem 2: Let R be an equivalence relation on a set S . Then the equivalence classes of R form a partition of S . Conversely, given a partition $\{A_i \mid i \in I\}$ of the set S , there is an equivalence relation R that has the sets A_i , $i \in I$, as its equivalence classes.

Proof: We have already shown the first part of the theorem.

For the second part, assume that $\{A_i \mid i \in I\}$ is a partition of S . Let R be the relation on S consisting of the pairs (x, y) where x and y belong to the same subset A_i in the partition. We must show that R satisfies the properties of an equivalence relation.

- *Reflexivity:* For every $a \in S$, $(a, a) \in R$, because a is in the same subset as itself.
- *Symmetry:* If $(a, b) \in R$, then b and a are in the same subset of the partition, so $(b, a) \in R$.
- *Transitivity:* If $(a, b) \in R$ and $(b, c) \in R$, then a and b are in the same subset of the partition, as are b and c . Since the subsets are disjoint and b belongs to both, the two subsets of the partition must be identical. Therefore, $(a, c) \in R$ since a and c belong to the same subset of the partition.

Partial Orderings

Section 9.6

Partial Orderings

Definition 1: A relation R on a set S is called a **partial ordering**, or **partial order**, if it is reflexive, antisymmetric, and transitive. A set together with a partial ordering R is called a **partially ordered set**, or **poset**, and is denoted by (S, R) . Members of S are called **elements** of the poset.

Partial Orderings (*continued*)

Example 1: Show that the “greater than or equal” relation ($\geq$) is a **partial ordering** on the set of integers.

- *Reflexivity:* $a \geq a$ for every integer a .
- *Antisymmetry:* If $a \geq b$ and $b \geq a$, then $a = b$.
- *Transitivity:* If $a \geq b$ and $b \geq c$, then $a \geq c$.

These properties all follow from the order axioms for the integers. (See *Appendix 1*).

Partial Orderings (*continued*)

Example 2: Show that the divisibility relation ($|$) is a **partial ordering** on the set of integers.

- *Reflexivity:* $a | a$ for all integers a . (see *Example 9 in Section 9.1*)
- *Antisymmetry:* If a and b are positive integers with $a | b$ and $b | a$, then $a = b$. (see *Example 12 in Section 9.1*)
- *Transitivity:* Suppose that a divides b and b divides c . Then there are positive integers k and l such that $b = ak$ and $c = bl$. Hence, $c = a(kl)$, so a divides c . Therefore, the relation is transitive.
- $(\mathbf{Z}^+, |)$ is a **poset**.

Partial Orderings (*continued*)

Example 3: Show that the inclusion relation ($\subseteq$) is a **partial ordering** on the power set of a set S .

- *Reflexivity:* $A \subseteq A$ whenever A is a subset of S .
- *Antisymmetry:* If A and B are positive integers with $A \subseteq B$ and $B \subseteq A$, then $A = B$.
- *Transitivity:* If $A \subseteq B$ and $B \subseteq C$, then $A \subseteq C$.

The properties all follow from the definition of set inclusion.



الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

Chapter 10

Graphs

Chapter Outlines

Graphs and Graph Models

Graph Terminology and Special Types of Graphs

Representing Graphs and Graph Isomorphism

Connectivity

Euler and Hamiltonian Graphs

Graphs and Graph Models

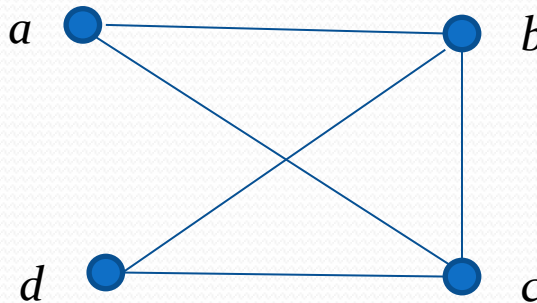
Section 10.1

Graphs

Definition: A graph $G = (V, E)$ consists of a nonempty set V of *vertices (or nodes)* and a set E of *edges*. Each edge has either one or two vertices associated with it, called its *endpoints*. An edge is said to *connect* its endpoints.

Example:

This is a graph with four vertices and five edges.



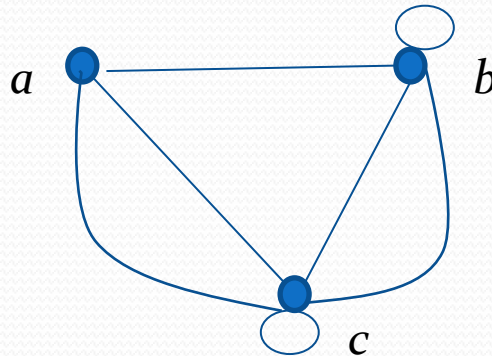
Remarks:

- The graphs we study here are unrelated to graphs of functions studied in Chapter 2.
- We have a lot of freedom when we draw a picture of a graph. All that matters is the connections made by the edges, not the particular geometry depicted. For example, the lengths of edges, whether edges cross, how vertices are depicted, and so on, do not matter.
- A graph with an infinite vertex set is called an *infinite graph*. A graph with a finite vertex set is called a *finite graph*. We (following the text) restrict our attention to finite graphs.

Some Terminology

- In a **simple graph** each edge connects two different vertices and no two edges **connect** the same pair of vertices.
- **Multigraphs** may have multiple edges connecting the same two vertices. When m different edges connect the vertices u and v , we say that $\{u,v\}$ is an edge of **multiplicity m** .
- An edge that connects a vertex to itself is called a **loop**.
- A **pseudograph** may include loops, as well as multiple edges connecting the same pair of vertices.

Example:
This pseudograph has both multiple edges and a loop.



Remark: There is no standard terminology for graph theory. So, it is crucial that you understand the terminology being used whenever you read material about graphs.

Directed Graphs

Definition: A **directed graph** (or **digraph**) $G = (V, E)$ consists of a nonempty set V of **vertices** (or **nodes**) and a set E of **directed edges** (or **arcs**). Each edge is associated with an ordered pair of vertices. The directed edge associated with the ordered pair (u, v) is said to *start at u* and *end at v* .

Remark:

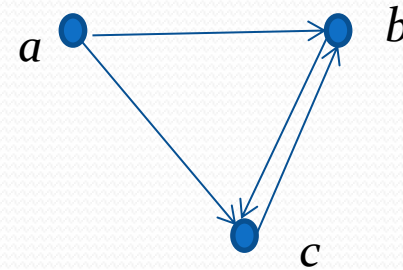
- Graphs where the end points of an edge are not ordered are said to be **undirected graphs**.

Some Terminology (*continued*)

- A **simple directed graph** has no loops and no multiple edges.

Example:

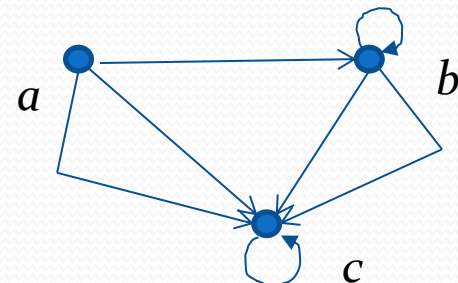
This is a directed graph with three vertices and four edges.



- A **directed multigraph** may have multiple directed edges. When there are m directed edges from the vertex u to the vertex v , we say that (u,v) is an edge of **multiplicity m** .

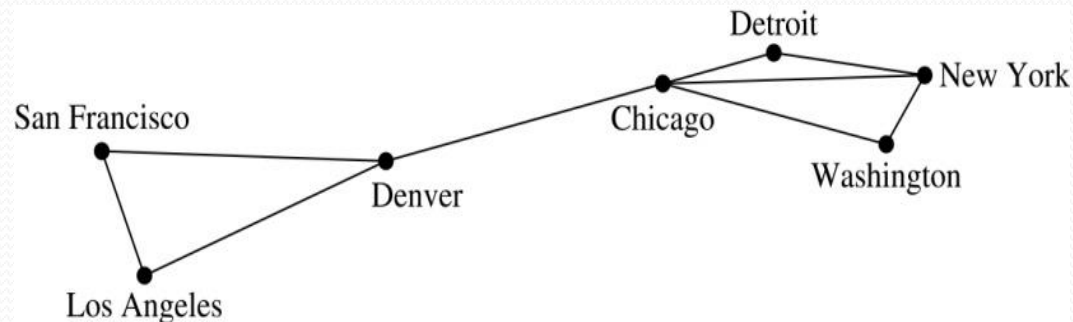
Example:

In this directed multigraph the multiplicity of (a,b) is 1 and the multiplicity of (b,c) is 2.



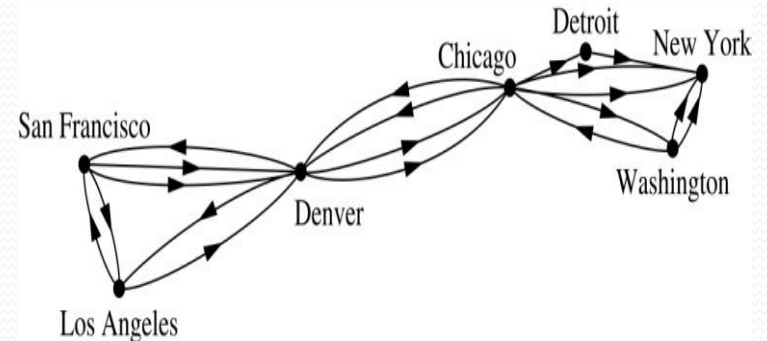
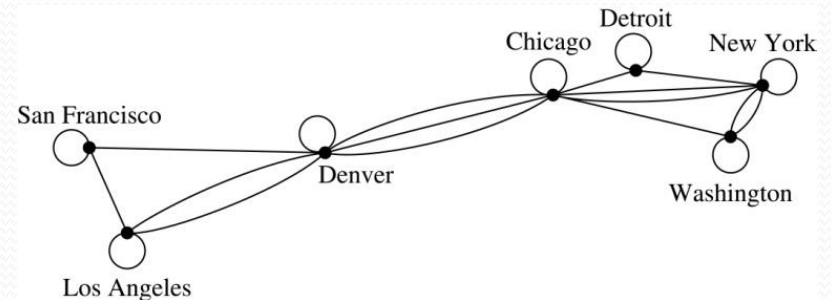
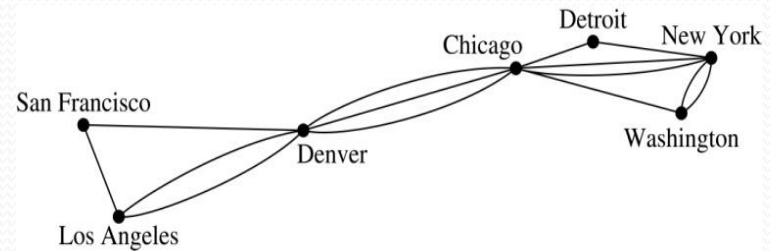
Graph Models: Computer Networks

- When we build a graph model, we use the appropriate type of graph to capture the important features of the application.
- We illustrate this process using graph models of different types of computer networks. In all these graph models, the vertices represent data centers and the edges represent communication links.
- To model a computer network where we are only concerned whether two data centers are connected by a communications link, we use a **simple graph**. This is the appropriate type of graph when we only care whether two data centers are directly linked (and not how many links there may be) and all communications links work in both directions.



Graph Models: Computer Networks (*continued*)

- To model a computer network where we care about the number of links between data centers, we use a multigraph.
- To model a computer network with diagnostic links at data centers, we use a pseudograph, as loops are needed.
- To model a network with multiple one-way links, we use a directed multigraph. Note that we could use a directed graph without multiple edges if we only care whether there is at least one link from a data center to another data center.



Graph Terminology: Summary

- To understand the structure of a graph and to build a graph model, we ask these questions:
 - Are the edges of the graph undirected or directed (or both)?
 - If the edges are undirected, are multiple edges present that connect the same pair of vertices? If the edges are directed, are multiple directed edges present?
 - Are loops present?

TABLE 1 Graph Terminology.			
<i>Type</i>	<i>Edges</i>	<i>Multiple Edges Allowed?</i>	<i>Loops Allowed?</i>
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed and undirected	Yes	Yes

Other Applications of Graphs

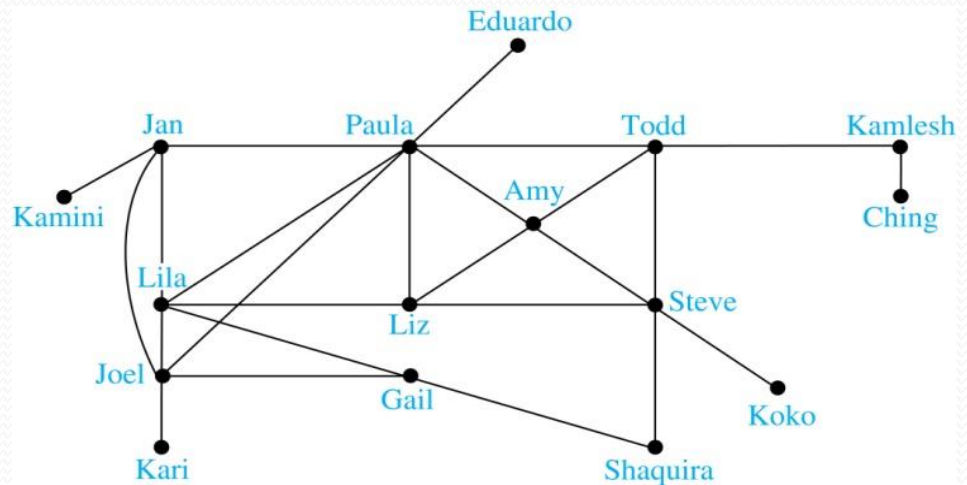
- We will illustrate how graph theory can be used in models of:
 - Social networks
 - Communications networks
 - Information networks
 - Software design
 - Transportation networks
 - Biological networks
- It's a challenge to find a subject to which graph theory has not yet been applied. Can you find an area without applications of graph theory?

Graph Models: Social Networks

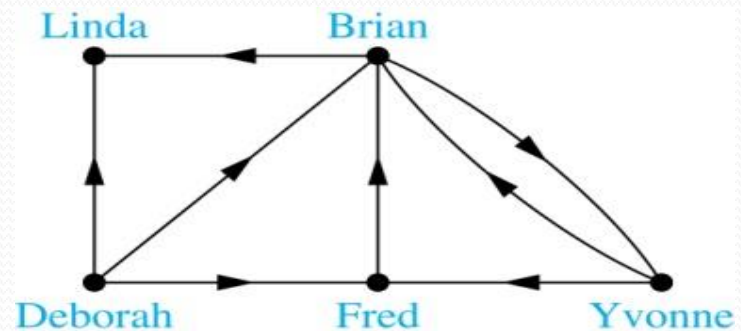
- Graphs can be used to model social structures based on different kinds of relationships between people or groups.
- In a **social network**, vertices represent individuals or organizations and edges represent relationships between them.
- Useful graph models of social networks include:
 - **friendship graphs** - undirected graphs where two people are connected if they are friends (in the real world, on Facebook, or in a particular virtual world, and so on.)
 - **collaboration graphs** - undirected graphs where two people are connected if they collaborate in a specific way
 - **influence graphs** - directed graphs where there is an edge from one person to another if the first person can influence the second person

Graph Models: Social Networks (*continued*)

Example: A friendship graph where two people are connected if they are Facebook friends.



Example: An influence graph



Next Slide: Collaboration Graphs

Applications to Information Networks

- Graphs can be used to model different types of networks that link different types of information.
- In a **web graph**, web pages are represented by vertices and links are represented by directed edges.
 - A web graph models the web at a particular time.
 - We will explain how the web graph is used by search engines in Section 11.4.
- In a **citation network**:
 - Research papers in a particular discipline are represented by vertices.
 - When a paper cites a second paper as a **reference**, there is an edge from the vertex representing this paper to the vertex representing the second paper.

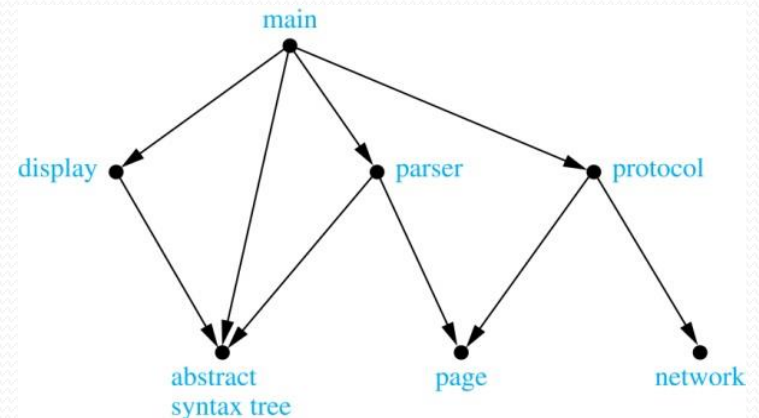
Transportation Graphs

- Graph models are extensively used in the study of transportation networks.
- Airline networks can be modeled using directed multigraphs where
 - airports are represented by vertices
 - each flight is represented by a directed edge from the vertex representing the departure airport to the vertex representing the destination airport
- Road networks can be modeled using graphs where
 - vertices represent intersections and edges represent roads.
 - undirected edges represent two-way roads and directed edges represent one-way roads.

Software Design Applications

- Graph models are extensively used in software design. We will introduce two such models here; one representing the dependency between the modules of a software application and the other representing restrictions in the execution of statements in computer programs.
- When a top-down approach is used to design software, the system is divided into modules, each performing a specific task.
- We use a **module dependency graph** to represent the dependency between these modules. These dependencies need to be understood before coding can be done.
 - In a module dependency graph vertices represent software modules and there is an edge from one module to another if the second module depends on the first.

Example: The dependencies between the seven modules in the design of a web browser are represented by this module dependency graph.

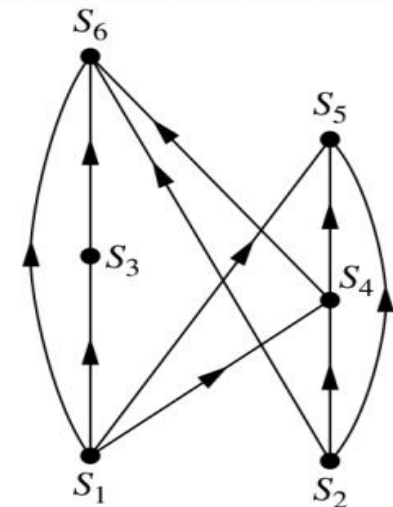


Software Design Applications (*continued*)

- We can use a directed graph called a **precedence graph** to represent which statements must have already been executed before we execute each statement.
 - Vertices represent **statements** in a computer program
 - There is a directed edge from a vertex to a second vertex if the second vertex cannot be executed before the first

Example: This precedence graph shows which statements must already have been executed before we can execute each of the six statements in the program.

S_1	$a := 0$
S_2	$b := 1$
S_3	$c := a + 1$
S_4	$d := b + a$
S_5	$e := d + 1$
S_6	$e := c + d$



Graph Terminology and Special Types of Graphs

Section 10.2

Basic Terminology

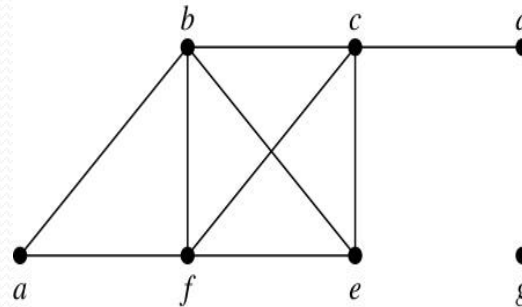
Definition 1. Two vertices u, v in an undirected graph G are called **adjacent (or neighbors)** in G if there is an edge e between u and v . Such an edge e is called **incident** with the vertices u and v and e is said to *connect* u and v .

Definition 2. The set of all neighbors of a vertex v of $G = (V, E)$, denoted by $N(v)$, is called the **neighborhood** of v . If A is a subset of V , we denote by $N(A)$ the set of all vertices in G that are adjacent to at least one vertex in A . So, $N(A) = \bigcup_{v \in A} N(v)$.

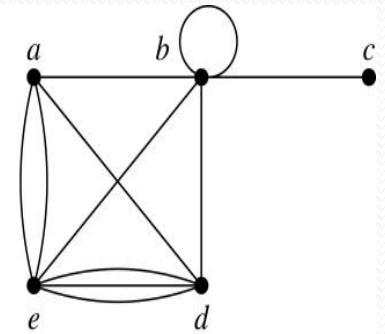
Definition 3. The **degree** of a vertex in an **undirected graph** is the number of edges incident with it, except that a **loop** at a vertex contributes two to the degree of that vertex. The degree of the vertex v is denoted by **$\deg(v)$** .

Degrees and Neighborhoods of Vertices

Example: What are the degrees and neighborhoods of the vertices in the graphs G and H ?



G



H

Solution:

G : $\deg(a) = 2, \deg(b) = \deg(c) = \deg(f) = 4, \deg(d) = 1,$
 $\deg(e) = 3, \deg(g) = 0.$

$N(a) = \{b, f\}, N(b) = \{a, c, e, f\}, N(c) = \{b, d, e, f\}, N(d) = \{c\},$
 $N(e) = \{b, c, f\}, N(f) = \{a, b, c, e\}, N(g) = \emptyset.$

H : $\deg(a) = 4, \deg(b) = \deg(e) = 6, \deg(c) = 1, \deg(d) = 5.$

$N(a) = \{b, d, e\}, N(b) = \{a, b, c, d, e\}, N(c) = \{b\},$
 $N(d) = \{a, b, e\}, N(e) = \{a, b, d\}.$

Degrees of Vertices

Theorem 1 (*Handshaking Theorem*): If $G = (V, E)$ is an undirected graph with m edges, then

$$2m = \sum_{v \in V} \deg(v)$$

Proof:

Each edge contributes twice to the degree count of all vertices. Hence, both the left-hand and right-hand sides of this equation equal twice the number of edges.



Think about the graph where vertices represent the people at a party and an edge connects two people who have shaken hands.

Handshaking Theorem

We now give two examples illustrating the usefulness of the handshaking theorem.

Example: How many edges are there in a graph with 10 vertices of degree six?

Solution: Because the sum of the degrees of the vertices is $6 \cdot 10 = 60$, the handshaking theorem tells us that $2m = 60$. So the number of edges $m = 30$.

Example: If a graph has 5 vertices, can each vertex have degree 3?

Solution: This is not possible by the handshaking theorem, because the sum of the degrees of the vertices $3 \cdot 5 = 15$ is odd.

Degree of Vertices (*continued*)

Theorem 2: An undirected graph has an even number of vertices of odd degree.

Proof: Let V_1 be the vertices of even degree and V_2 be the vertices of odd degree in an undirected graph $G = (V, E)$ with m edges. Then

$$\text{even} \longrightarrow 2m = \sum_{v \in V} \deg(v) = \sum_{v \in V_1} \deg(v) + \sum_{v \in V_2} \deg(v).$$

must be
even since
 $\deg(v)$ is
even for
each $v \in V_1$

This sum must be even because $2m$ is even and the sum of the degrees of the vertices of even degrees is also even. Because this is the sum of the degrees of all vertices of odd degree in the graph, there must be an even number of such vertices.

Directed Graphs

Recall the definition of a directed graph.

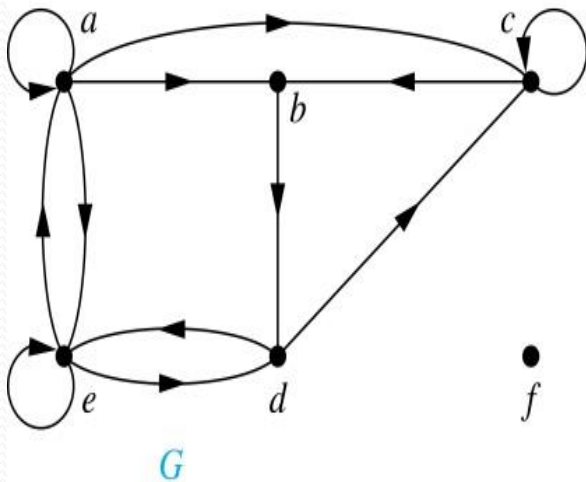
Definition: A **directed graph** $G = (V, E)$ consists of V , a nonempty set of **vertices (or nodes)**, and E , a set of **directed edges or arcs**. Each edge is an ordered pair of vertices. The directed edge (u, v) is said to start at u and end at v .

Definition: Let (u, v) be an edge in G . Then u is the **initial** vertex of this edge and is *adjacent to* v and v is the **terminal (or end)** vertex of this edge and is *adjacent from* u . The initial and terminal vertices of a **loop** are the same.

Directed Graphs (*continued*)

Definition: The *in-degree* of a vertex v , denoted $\deg^-(v)$, is the number of edges which *terminate* at v . The *out-degree* of v , denoted $\deg^+(v)$, is the number of edges with v as their *initial* vertex. Note that a *loop* at a vertex contributes 1 to both the *in-degree* and the *out-degree* of the vertex.

Example: In the graph G we have



$$\deg^-(a) = 2, \deg^-(b) = 2, \deg^-(c) = 3, \deg^-(d) = 2, \\ \deg^-(e) = 3, \deg^-(f) = 0.$$

$$\deg^+(a) = 4, \deg^+(b) = 1, \deg^+(c) = 2, \deg^+(d) = 2, \\ \deg^+(e) = 3, \deg^+(f) = 0.$$

Directed Graphs (*continued*)

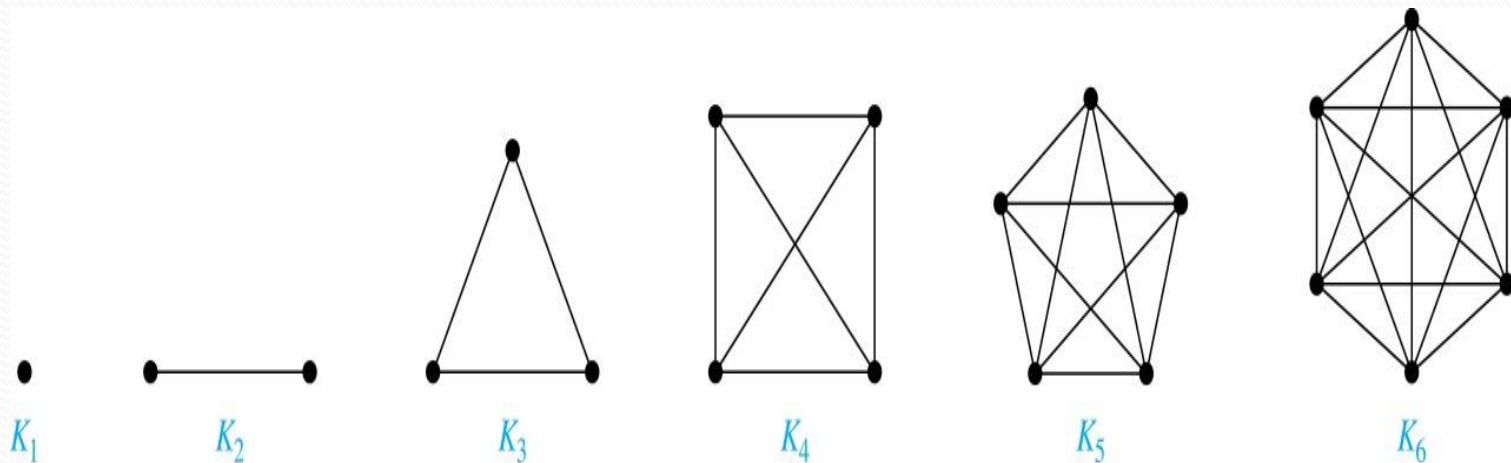
Theorem 3: Let $G = (V, E)$ be a graph with directed edges. Then:

$$|E| = \sum_{v \in V} \deg^{-}(v) = \sum_{v \in V} \deg^{+}(v).$$

Proof: The first sum counts the number of outgoing edges over all vertices and the second sum counts the number of incoming edges over all vertices. It follows that both sums equal the number of edges in the graph. ◀

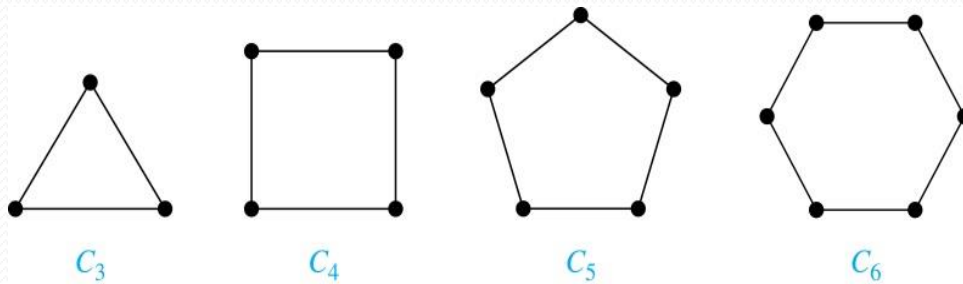
Special Types of Simple Graphs: Complete Graphs

A **complete graph** on n vertices, denoted by K_n , is the simple graph that contains exactly one edge between each pair of distinct vertices.

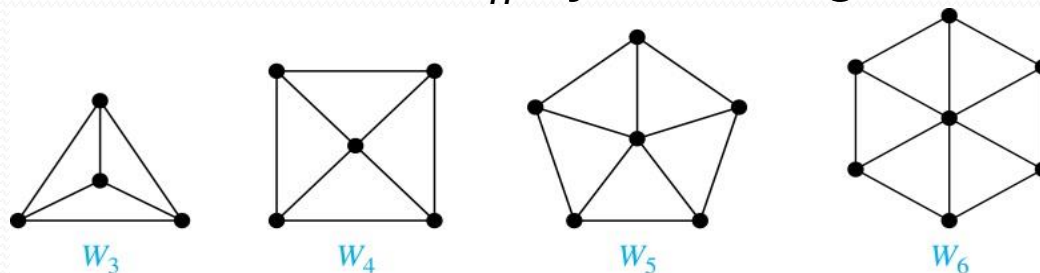


Special Types of Simple Graphs: Cycles and Wheels

A **cycle** C_n for $n \geq 3$ consists of n vertices $v_1, v_2, \dots, v_n$, and **edges** $\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}, \{v_n, v_1\}$.

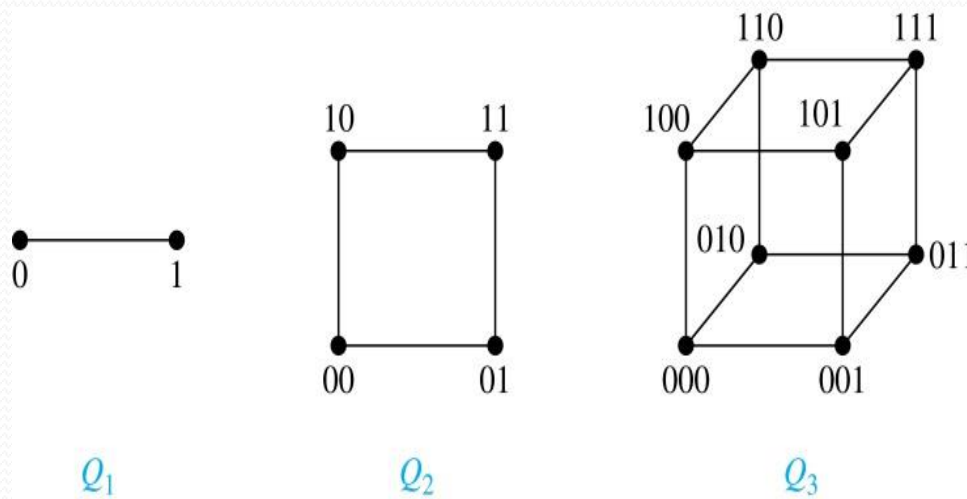


A **wheel** W_n is obtained by adding an **additional vertex** to a cycle C_n for $n \geq 3$ and connecting this new vertex to each of the n vertices in C_n by new edges.



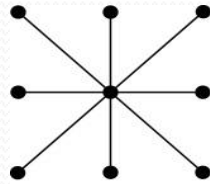
Special Types of Simple Graphs: n -Cubes

An **n -dimensional hypercube**, or **n -cube**, Q_n , is a graph with 2^n vertices representing all bit strings of length n , where there is an edge between two vertices that differ in exactly **one bit position**.

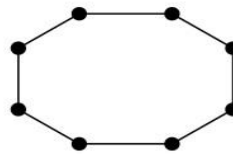


Special Types of Graphs and Computer Network Architecture

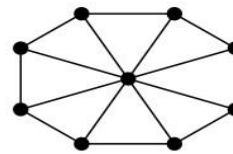
Various special graphs play an important role in the design of computer networks.



(a)

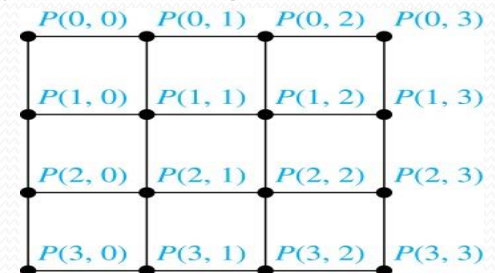


(b)



(c)

- Some local area networks use a *star topology*, which is a complete bipartite graph $K_{1,n}$, as shown in (a). All devices are connected to a central control device.
- Other local networks are based on a *ring topology*, where each device is connected to exactly two others using C_n , as illustrated in (b). Messages may be sent around the ring.
- Others, as illustrated in (c), use a W_n – based topology, combining the features of a star topology and a ring topology.
- Various special graphs also play a role in parallel processing where processors need to be interconnected as one processor may need the output generated by another.
 - The n -dimensional hypercube, or n -cube, Q_n , is a common way to connect processors in parallel, e.g., Intel Hypercube.
 - Another common method is the *mesh* network, illustrated here for 16 processors.

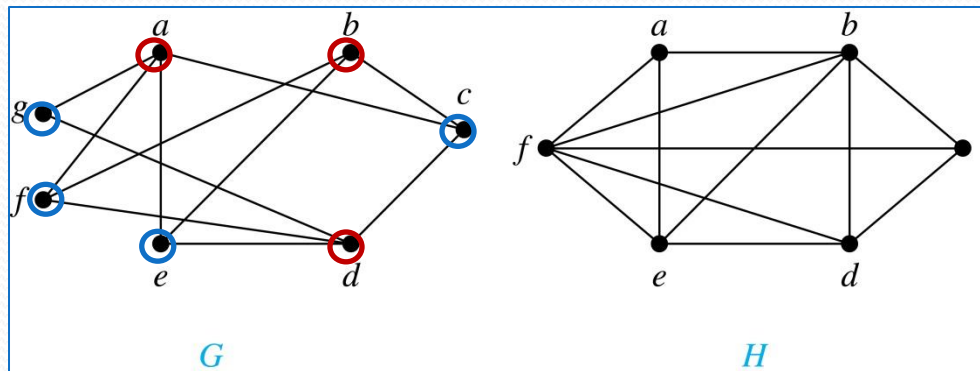


Bipartite Graphs

Definition: A simple graph G is **bipartite** if V can be partitioned into two **disjoint subsets** V_1 and V_2 such that every edge connects a vertex in V_1 and a vertex in V_2 . In other words, there are no edges which connect two vertices in V_1 or in V_2 .

It is not hard to show that an equivalent definition of a bipartite graph is a graph where it is possible to color the vertices red or blue so that no two adjacent vertices are the same color.

G is bipartite

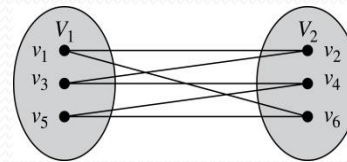
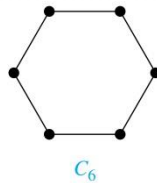
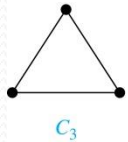


H is not bipartite since if we color a red, then the adjacent vertices f and b must both be blue.

Bipartite Graphs (*continued*)

Example: Show that C_6 is bipartite.

Solution: We can **partition** the vertex set into $V_1 = \{v_1, v_3, v_5\}$ and $V_2 = \{v_2, v_4, v_6\}$ so that every edge of C_6 **connects** a vertex in V_1 and V_2 .



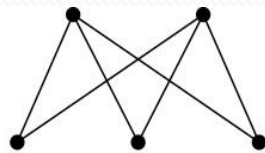
Example: Show that C_3 is not bipartite.

Solution: If we divide the vertex set of C_3 into two **nonempty sets**, one of the two must contain two vertices. But in C_3 every vertex is **connected** to every other vertex. Therefore, the two vertices in the same **partition** are connected. Hence, C_3 is **not bipartite**.

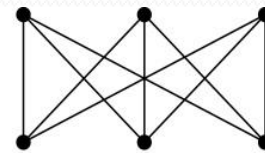
Complete Bipartite Graphs

Definition: A **complete bipartite graph** $K_{m,n}$ is a graph that has its vertex set **partitioned** into two subsets V_1 of size m and V_2 of size n such that there is an edge from every vertex in V_1 to every vertex in V_2 .

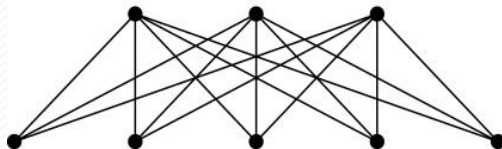
Example: We display four complete bipartite graphs here.



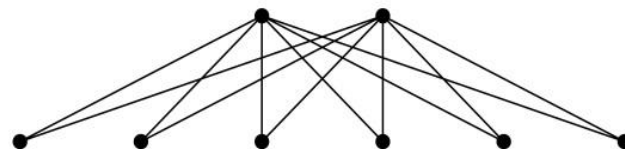
$K_{2,3}$



$K_{3,3}$



$K_{3,5}$

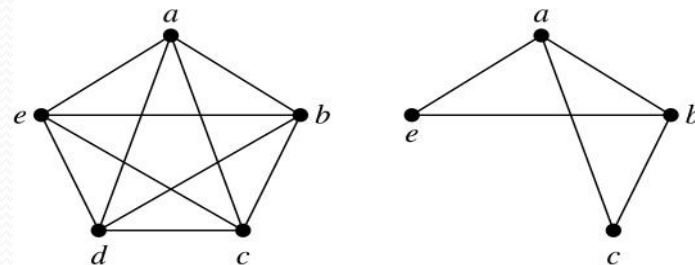


$K_{2,6}$

New Graphs from Old

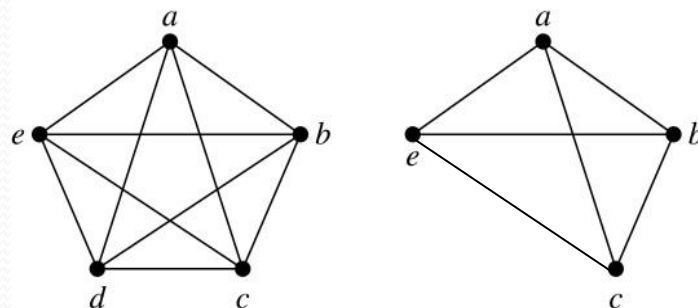
Definition: A **subgraph** of a graph $G = (V, E)$ is a graph (W, F) , where $W \subset V$ and $F \subset E$. A subgraph H of G is a proper subgraph of G if $H \neq G$.

Example: Here we show K_5 and one of its subgraphs.



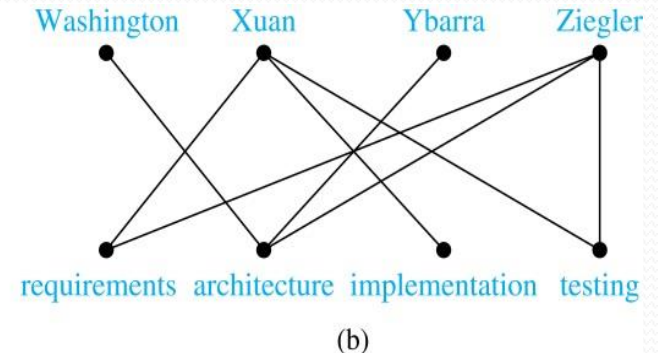
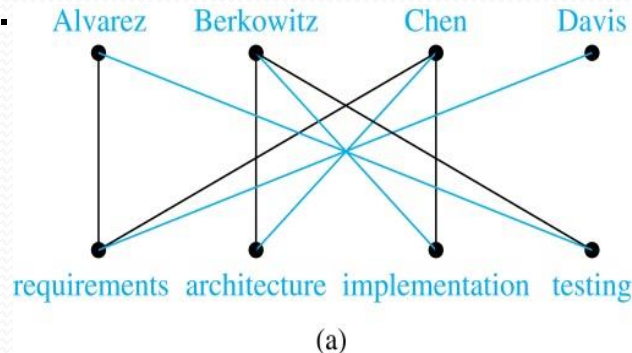
Definition: Let $G = (V, E)$ be a simple graph. The **subgraph induced** by a subset W of the vertex set V is the graph (W, F) , where the edge set F contains an edge in E if and only if both endpoints are in W .

Example: Here we show K_5 and the subgraph induced by $W = \{a, b, c, e\}$.



Bipartite Graphs and Matchings

- Bipartite graphs are used to model applications that involve matching the elements of one set to elements in another, for example:
- *Job assignments* - vertices represent the jobs and the employees, edges link employees with those jobs they have been trained to do. A common goal is to **match** jobs to employees so that the most jobs are done.



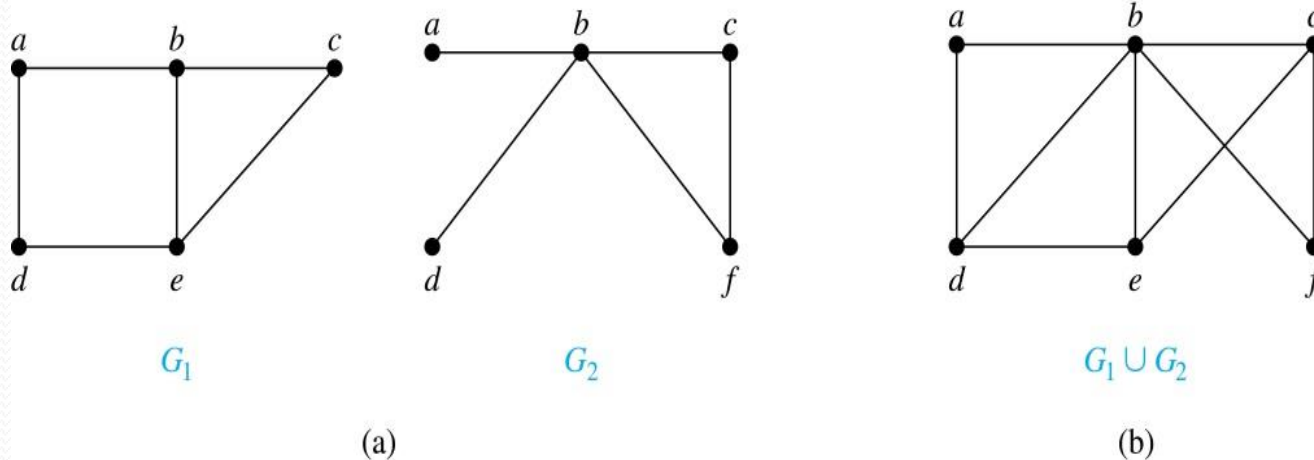
- *Marriage* - vertices represent the men and the women and edges link a man and a woman if they are an acceptable spouse. We may wish to find the largest number of possible marriages.

See the text for more about matchings in bipartite graphs.

New Graphs from Old (*continued*)

Definition: The *union* of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$. The union of G_1 and G_2 is denoted by $G_1 \cup G_2$.

Example:



Representing Graphs and Graph Isomorphism

Section 10.3

Representing Graphs:Adjacency Lists

Definition: An *adjacency list* can be used to represent a graph with no *multiple edges* by specifying the vertices that are adjacent to each vertex of the graph.

Example:

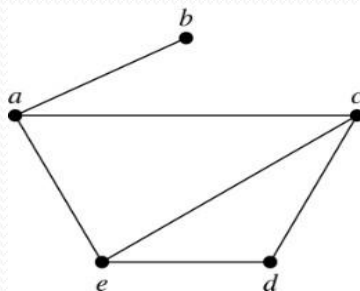


TABLE 1 An Adjacency List for a Simple Graph.

Vertex	Adjacent Vertices
<i>a</i>	<i>b, c, e</i>
<i>b</i>	<i>a</i>
<i>c</i>	<i>a, d, e</i>
<i>d</i>	<i>c, e</i>
<i>e</i>	<i>a, c, d</i>

Example:

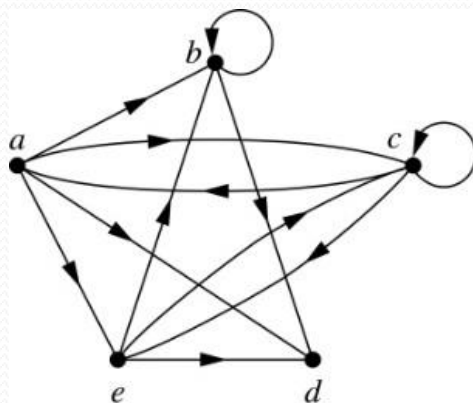


TABLE 2 An Adjacency List for a Directed Graph.

Initial Vertex	Terminal Vertices
<i>a</i>	<i>b, c, d, e</i>
<i>b</i>	<i>b, d</i>
<i>c</i>	<i>a, c, e</i>
<i>d</i>	
<i>e</i>	<i>b, c, d</i>

Representation of Graphs: Adjacency Matrices

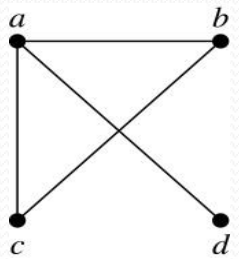
Definition: Suppose that $G = (V, E)$ is a simple graph where $|V| = n$. Arbitrarily list the vertices of G as $v_1, v_2, \dots, v_n$. The **adjacency matrix** A_G of G , with respect to the listing of vertices, is the $n \times n$ **zero-one matrix** with 1 as its (i, j) th entry when v_i and v_j are **adjacent**, and 0 as its (i, j) th entry when they are **not adjacent**.

- In other words, if the graph's adjacency matrix is $A_G = [a_{ij}]$, then

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

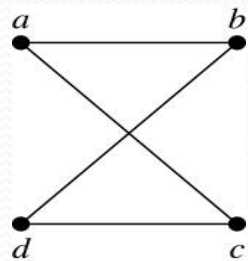
Adjacency Matrices (*continued*)

Example:



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

The *ordering* of vertices is a, b, c, d .



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

The *ordering* of vertices is a, b, c, d .

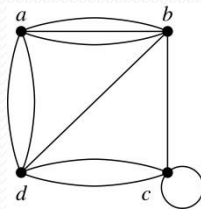
When a graph is sparse, that is, it has few edges relatively to the total number of possible edges, it is much more efficient to represent the graph using an adjacency list than an adjacency matrix. But for a dense graph, which includes a high percentage of possible edges, an adjacency matrix is preferable.

Note: The adjacency matrix of a simple graph is symmetric, i.e., $a_{ij} = a_{ji}$. Also, since there are no loops, each diagonal entry a_{ii} for $i = 1, 2, 3, \dots, n$, is 0.

Adjacency Matrices (*continued*)

- Adjacency matrices can also be used to represent graphs with **loops** and **multiple edges**.
- A loop at the vertex v_i is represented by a 1 at the (i, i) th position of the matrix.
- When **multiple edges** connect the same pair of vertices v_i and v_j , (or if **multiple loops** are present at the same vertex), the (i, j) th entry **equals** the number of edges connecting the pair of vertices.

Example: We give the adjacency matrix of the **pseudograph** shown here using the ordering of vertices a, b, c, d .



$$\begin{bmatrix} 0 & 2 & 0 & 2 \\ 2 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

Adjacency Matrices (*continued*)

- Adjacency matrices can also be used to represent **directed graphs**. The **matrix** for a directed graph $G = (V, E)$ has a 1 in its (i, j) th position if there is an **edge** from v_i to v_j , where $v_1, v_2, \dots, v_n$ is a list of the vertices.

- In other words, if the graph's adjacency matrix is $\mathbf{A}_G = [a_{ij}]$, then

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

- The adjacency matrix for a directed graph does not have to be **symmetric**, because there may not be an edge from v_i to v_j , when there is an edge from v_j to v_i .
- To represent **directed multigraphs**, the value of a_{ij} is the number of edges connecting v_i to v_j .

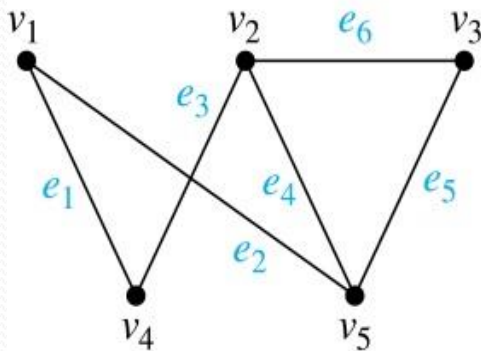
Representation of Graphs: Incidence Matrices

Definition: Let $G = (V, E)$ be an undirected graph with vertices where $v_1, v_2, \dots, v_n$ and edges $e_1, e_2, \dots, e_m$. The **incidence matrix** with respect to the ordering of V and E is the $n \times m$ matrix $\mathbf{M} = [m_{ij}]$, where

$$m_{ij} = \begin{cases} 1 & \text{when edge } e_j \text{ is incident with } v_i, \\ 0 & \text{otherwise.} \end{cases}$$

Incidence Matrices (*continued*)

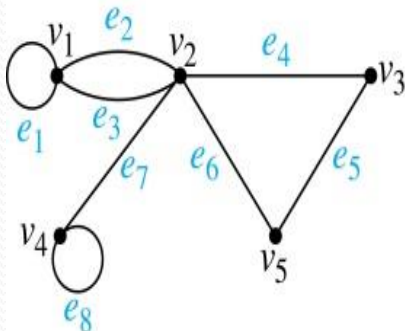
Example: Simple Graph and Incidence Matrix



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

The rows going from top to bottom represent v_1 through v_5 and the columns going from left to right represent e_1 through e_6 .

Example: Pseudograph and Incidence Matrix



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

The rows going from top to bottom represent v_1 through v_5 and the columns going from left to right represent e_1 through e_8 .

Isomorphism of Graphs

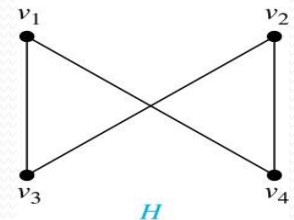
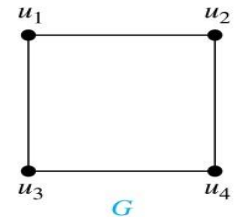
Definition: The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are **isomorphic** if there is a one-to-one and onto function f from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if $f(a)$ and $f(b)$ are adjacent in G_2 , for all a and b in V_1 . Such a function f is called an **isomorphism**. Two simple graphs that are not isomorphic are called **nonisomorphic**.

Isomorphism of Graphs (cont.)

Example: Show that the graphs $G = (V, E)$ and $H = (W, F)$ are isomorphic.

Solution: The function f with $f(u_1) = v_1$, $f(u_2) = v_4$, $f(u_3) = v_3$, and $f(u_4) = v_2$ is a one-to-one correspondence between V and W .

Note that adjacent vertices in G are u_1 and u_2 , u_1 and u_3 , u_2 and u_4 , and u_3 and u_4 . Each of the pairs $f(u_1) = v_1$ and $f(u_2) = v_4$, $f(u_1) = v_1$ and $f(u_3) = v_3$, $f(u_2) = v_4$ and $f(u_4) = v_2$, and $f(u_3) = v_3$ and $f(u_4) = v_2$ consists of two adjacent vertices in H .

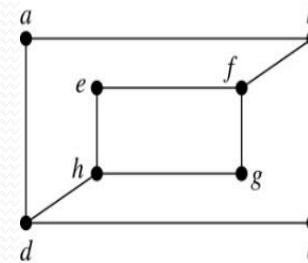


Isomorphism of Graphs (cont.)

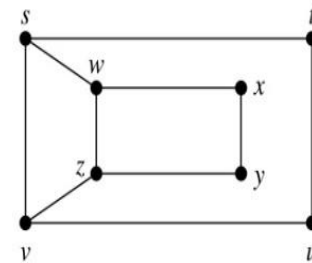
- It is difficult to determine whether two simple graphs are isomorphic using brute force because there are $n!$ possible one-to-one correspondences between the vertex sets of two simple graphs with n vertices.
- The best algorithms for determining whether two graphs are isomorphic have exponential worst case complexity in terms of the number of vertices of the graphs.
- Sometimes it is not hard to show that two graphs are not isomorphic. We can do so by finding a property, preserved by isomorphism, that only one of the two graphs has. Such a property is called **graph invariant**.
- There are many different useful graph invariants that can be used to distinguish nonisomorphic graphs, such as the number of vertices, number of edges, and degree sequence (list of the degrees of the vertices in nonincreasing order). We will encounter others in later sections of this chapter.

Isomorphism of Graphs (cont.)

Example: Determine whether these two graphs are isomorphic.



G

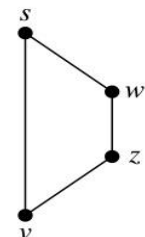
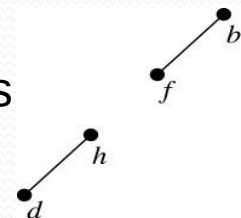


H

Solution: Both graphs have eight vertices and ten edges. They also both have four vertices of degree two and four of degree three.

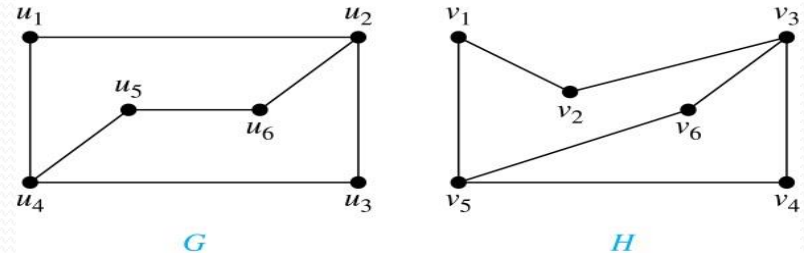
However, *G* and *H* are **not isomorphic**. Note that since $\deg(a) = 2$ in *G*, *a* must correspond to *t*, *u*, *x*, or *y* in *H*, because these are the vertices of degree 2. But each of these vertices is adjacent to another vertex of degree two in *H*, which is not true for *a* in *G*.

Alternatively, note that the **subgraphs** of *G* and *H* made up of vertices of degree three and the edges connecting them must be isomorphic. But the subgraphs, as shown at the right, are not isomorphic.



Isomorphism of Graphs (cont.)

Example: Determine whether these two graphs are isomorphic.



Solution: Both graphs have six vertices and seven edges. They also both have four vertices of degree two and two of degree three. The subgraphs of G and H consisting of all the vertices of degree two and the edges connecting them are isomorphic. So, it is reasonable to try to find an isomorphism f .

We define an injection f from the vertices of G to the vertices of H that preserves the degree of vertices. We will determine whether it is an isomorphism.

The function f with $f(u_1) = v_6$, $f(u_2) = v_3$, $f(u_3) = v_4$, and $f(u_4) = v_5$, $f(u_5) = v_1$, and $f(u_6) = v_2$ is a one-to-one correspondence between G and H . Showing that this correspondence preserves edges is straightforward, so we will omit the details here. Because f is an isomorphism, it follows that G and H are isomorphic graphs.

See the text for an illustration of how adjacency matrices can be used for this verification.

Applications of Graph Isomorphism

- The question whether graphs are **isomorphic** plays an important role in applications of graph theory. For example,
 - chemists use molecular graphs to model chemical compounds. Vertices represent atoms and edges represent chemical bonds. When a new compound is synthesized, a database of molecular graphs is checked to determine whether the graph representing the new compound is isomorphic to the graph of a compound that this already known.
 - Electronic circuits are modeled as **graphs** in which the vertices represent components and the edges represent connections between them. Graph isomorphism is the basis for
 - the verification that a particular layout of a circuit corresponds to the design's original schematics.
 - determining whether a chip from one vendor includes the intellectual property of another vendor.

Connectivity

Section 10.4

Paths

Informal Definition: A *path* is a sequence of edges that begins at a vertex of a graph and travels from vertex to vertex along edges of the graph. As the path travels along its edges, it visits the vertices along this path, that is, the endpoints of these.

Applications: Numerous problems can be modeled with paths formed by traveling along edges of graphs such as:

- determining whether a message can be sent between two computers.
- efficiently planning routes for mail delivery.

Paths

Definition: Let n be a nonnegative integer and G an undirected graph. A **path of length** n from u to v in G is a **sequence** of n edges $e_1, \dots, e_n$ of G for which there exists a sequence $x_0 = u, x_1, \dots, x_{n-1}, x_n = v$ of vertices such that e_i has, for $i = 1, \dots, n$, the endpoints x_{i-1} and x_i .

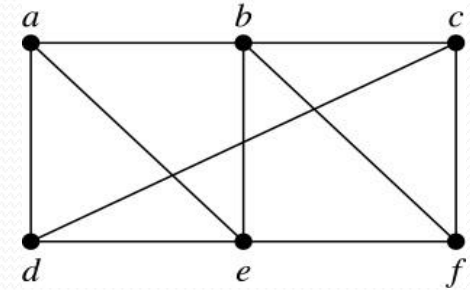
- When the graph is simple, we denote this path by its vertex sequence $x_0, x_1, \dots, x_n$ (since listing the vertices uniquely determines the path).
- The path is a **circuit** if it begins and ends at the same vertex ($u = v$) and has length greater than zero.
- The path or circuit is said to **pass through** the vertices $x_1, x_2, \dots, x_{n-1}$ and **traverse** the edges $e_1, \dots, e_n$.
- A path or circuit is **simple** if it does not contain the same edge more than once.

This terminology is readily extended to directed graphs. (see *text*)

Paths (*continued*)

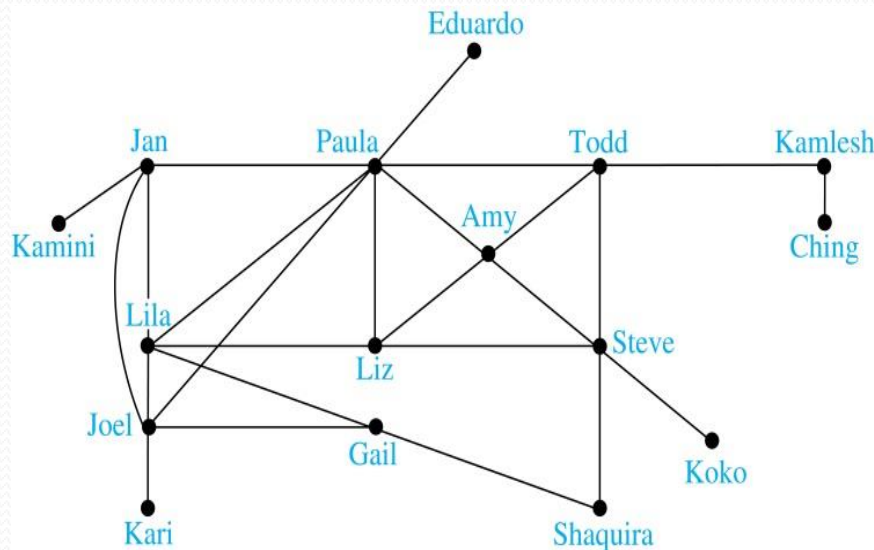
Example: In the simple graph here:

- a, d, c, f, e is a **simple path** of length 4.
- d, e, c, a is not a path because e is not **connected** to c .
- b, c, f, e, b is a **circuit** of length 4.
- a, b, e, d, a, b is a path of length 5, but it is not a **simple path**.



Degrees of Separation

Example: *Paths in Acquaintanceship Graphs.* In an acquaintanceship graph there is a path between two people if there is a chain of people linking these people, where two people adjacent in the chain know one another. In this graph there is a chain of six people linking Kamini and Ching.

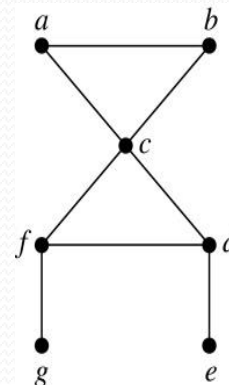


Some have speculated that almost every pair of people in the world are linked by a small chain of no more than six, or maybe even, five people. The play *Six Degrees of Separation* by John Guare is based on this notion.

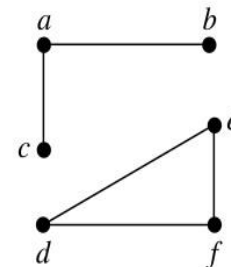
Connectedness in Undirected Graphs

Definition: An undirected graph is called **connected** if there is a path between every pair of vertices. An undirected graph that is not *connected* is called **disconnected**. We say that we *disconnect* a graph when we remove vertices or edges, or both, to produce a disconnected subgraph.

Example: G_1 is connected because there is a path between any pair of its vertices, as can be easily seen. However G_2 is not connected because there is no path between vertices a and f , for example.



G_1

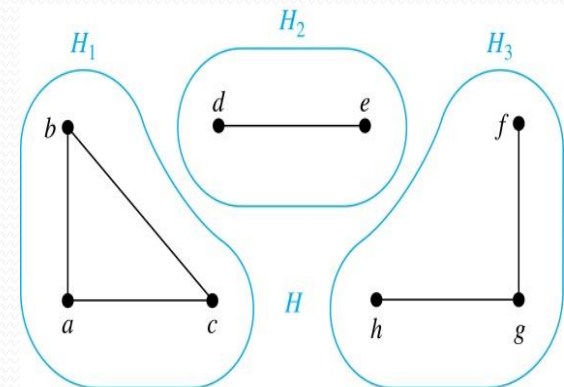


G_2

Connected Components

Definition: A **connected component** of a graph G is a connected subgraph of G that is not a **proper subgraph** of another connected subgraph of G . A graph G that is not connected has two or more connected components that are disjoint and have G as their union.

Example: The graph H is the union of three **disjoint subgraphs** H_1 , H_2 , and H_3 , none of which are proper subgraphs of a larger connected subgraph of G . These three subgraphs are the connected components of H .



Connectedness in Directed Graphs

Definition: A directed graph is *strongly connected* if there is a path from a to b and a path from b to a whenever a and b are vertices in the graph.

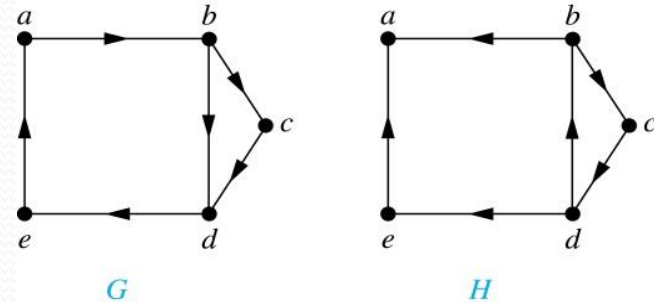
Definition: A directed graph is *weakly connected* if there is a path between every two vertices in the underlying undirected graph, which is the undirected graph obtained by ignoring the directions of the edges of the directed graph.

Connectedness in Directed Graphs

(continued)

Example: G is **strongly connected** because there is a path between any two vertices in the directed graph. Hence, G is also weakly connected.

The graph H is not strongly connected, since there is no directed path from a to b , but it is weakly connected.



Definition: The subgraphs of a directed graph G that are strongly connected but not contained in larger strongly connected subgraphs, that is, the maximal strongly connected subgraphs, are called the **strongly connected components** or **strong components** of G .

Example (continued): The graph H has three strongly connected components, consisting of the vertex a ; the vertex e ; and the subgraph consisting of the vertices b, c, d and edges (b, c) , (c, d) , and (d, b) .

The Connected Components of the Web Graph

- Recall that at any particular instant the web graph provides a snapshot of the web, where vertices represent web pages and edges represent links. According to a 1999 study, the Web graph at that time had over 200 million vertices and over 1.5 billion edges. (The numbers today are several orders of magnitude larger.)
- The underlying undirected graph of this Web graph has a connected component that includes approximately 90% of the vertices.
- There is a **giant strongly connected component** (GSCC) consisting of more than 53 million vertices. A Web page in this component can be reached by following links starting in any other page of the component. There are three other categories of pages with each having about 44 million vertices:
 - pages that can be reached from a page in the GSCC, but do not link back.
 - pages that link back to the GSCC, but can not be reached by following links from pages in the GSCC.
 - pages that cannot reach pages in the GSCC and can not be reached from pages in the GSCC.

Counting Paths between Vertices

- We can use the adjacency matrix of a graph to find the number of paths between two vertices in the graph.

Theorem: Let G be a graph with adjacency matrix $\mathbf{A}$ with respect to the ordering $v_1, \dots, v_n$ of vertices (with directed or undirected edges, multiple edges and loops allowed). The number of different paths of length r from v_i to v_j , where $r > 0$ is a positive integer, equals the (i,j) th entry of $\mathbf{A}^r$.

Proof by mathematical induction:

Basis Step: By definition of the adjacency matrix, the number of paths from v_i to v_j of length 1 is the (i,j) th entry of $\mathbf{A}$.

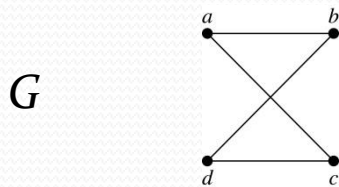
Inductive Step: For the inductive hypothesis, we assume that the (i,j) th entry of $\mathbf{A}^r$ is the number of different paths of length r from v_i to v_j .

- Because $\mathbf{A}^{r+1} = \mathbf{A}^r \mathbf{A}$, the (i,j) th entry of $\mathbf{A}^{r+1}$ equals $b_{i1}a_{1j} + b_{i2}a_{2j} + \dots + b_{in}a_{nj}$, where b_{ik} is the (i,k) th entry of $\mathbf{A}^r$. By the inductive hypothesis, b_{ik} is the number of paths of length r from v_i to v_k .
- A path of length $r + 1$ from v_i to v_j is made up of a path of length r from v_i to some v_k , and an edge from v_k to v_j . By the product rule for counting, the number of such paths is the product of the number of paths of length r from v_i to v_k (i.e., b_{ik}) and the number of edges from v_k to v_j (i.e., a_{kj}). The sum over all possible intermediate vertices v_k is $b_{i1}a_{1j} + b_{i2}a_{2j} + \dots + b_{in}a_{nj}$.



Counting Paths between Vertices (*continued*)

Example: How many paths of length four are there from a to d in the graph G .



$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad \text{adjacency matrix of } G$$

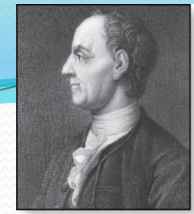
Solution: The **adjacency matrix** of G (ordering the vertices as a, b, c, d) is given above. Hence the number of paths of length four from a to d is the (1, 4)th entry of A^4 . The eight paths are as:

$$A^4 = \begin{bmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{bmatrix}$$

a, b, a, b, d	a, b, a, c, d
a, b, d, b, d	a, b, d, c, d
a, c, a, b, d	a, c, a, c, d
a, c, d, b, d	a, c, d, c, d

Euler and Hamiltonian Graphs

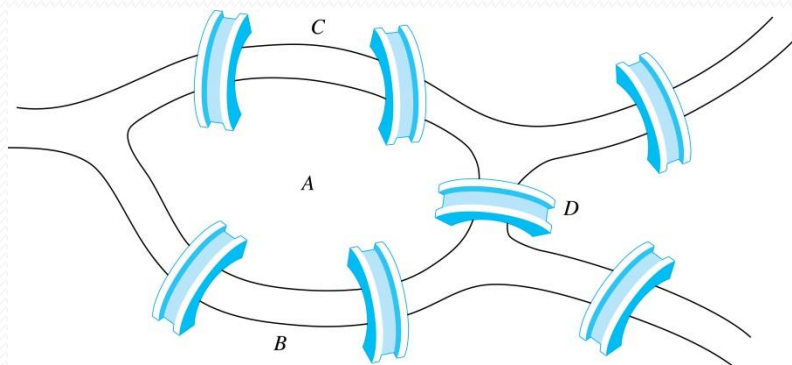
Section 10.5



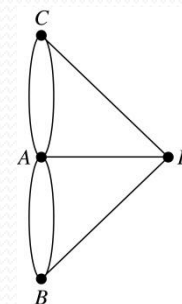
Leonard Euler
(1707-1783)

Euler Paths and Circuits

- The town of Königsberg, Prussia (now Kalingrad, Russia) was divided into four sections by the branches of the Pregel river. In the 18th century seven bridges connected these regions.
- People wondered whether it was possible to follow a path that crosses each bridge exactly once and returns to the starting point.
- The Swiss mathematician Leonard Euler proved that no such path exists. This result is often considered to be the first theorem ever proved in graph theory.



The 7 Bridges of Königsberg

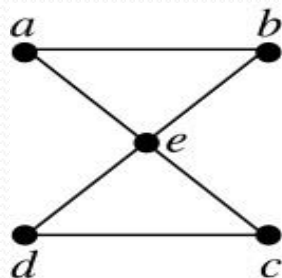


**Multigraph
Model of the
Bridges of
Königsberg**

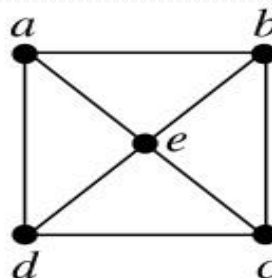
Euler Paths and Circuits (*continued*)

Definition: An **Euler circuit** in a graph G is a simple circuit containing every edge of G . An **Euler path** in G is a simple path containing every edge of G .

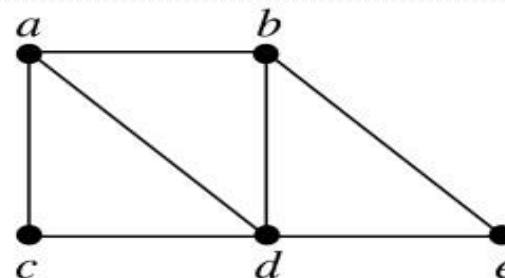
Example: Which of the undirected graphs G_1 , G_2 , and G_3 has a Euler circuit? Of those that do not, which has an Euler path?



G_1



G_2



G_3

Solution: The graph G_1 has an Euler circuit (e.g., a, e, c, d, e, b, a). But, as can easily be verified by inspection, neither G_2 nor G_3 has an Euler circuit. Note that G_3 has an Euler path (e.g., a, c, d, e, b, d, a, b), but there is no Euler path in G_2 , which can be verified by inspection.

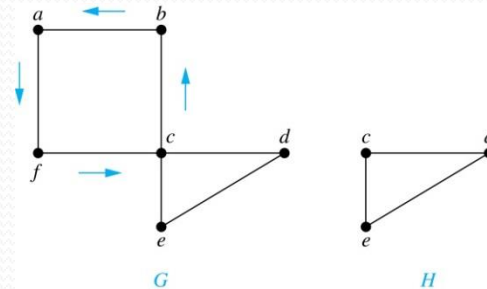
Necessary Conditions for Euler Circuits and Paths

- An Euler circuit begins with a vertex a and continues with an edge incident with a , say $\{a, b\}$. The edge $\{a, b\}$ contributes one to $\deg(a)$.
- Each time the circuit passes through a vertex it contributes two to the vertex's degree.
- Finally, the circuit terminates where it started, contributing one to $\deg(a)$. Therefore $\deg(a)$ must be even.
- We conclude that the degree of every other vertex must also be even.
- By the same reasoning, we see that the initial vertex and the final vertex of an Euler path have odd degree, while every other vertex has even degree. So, a graph with an Euler path has exactly two vertices of odd degree.
- In the next slide we will show that these necessary conditions are also sufficient conditions.

Sufficient Conditions for Euler Circuits and Paths

Suppose that G is a connected multigraph with ≥ 2 vertices, all of even degree. Let $x_0 = a$ be a vertex of even degree. Choose an edge $\{x_0, x_1\}$ incident with a and proceed to build a simple path $\{x_0, x_1\}, \{x_1, x_2\}, \dots, \{x_{n-1}, x_n\}$ by adding edges one by one until another edge can not be added.

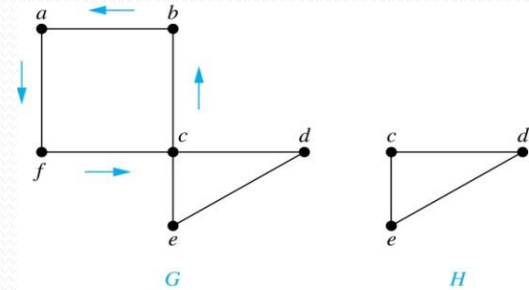
We illustrate this idea in the graph G here. We begin at a and choose the edges $\{a, f\}$, $\{f, c\}$, $\{c, b\}$, and $\{b, a\}$ in succession.



- The path begins at a with an edge of the form $\{a, x\}$; we show that it must terminate at a with an edge of the form $\{y, a\}$. Since each vertex has an even degree, there must be an even number of edges incident with this vertex. Hence, every time we enter a vertex other than a , we can leave it. Therefore, the path can only end at a .
- If all of the edges have been used, an Euler circuit has been constructed. Otherwise, consider the subgraph H obtained from G by deleting the edges already used.

In the example H consists of the vertices c, d, e .

Sufficient Conditions for Euler Circuits and Paths (*continued*)



- Because G is connected, H must have at least one vertex in common with the circuit that has been deleted.

In the example, the vertex is c.

- Every vertex in H must have even degree because all the vertices in G have even degree and for each vertex, pairs of edges incident with this vertex have been deleted. Beginning with the shared vertex construct a path ending in the same vertex (as was done before). Then splice this new circuit into the original circuit.

In the example, we end up with the circuit a, f, c, d, e, c, b, a .

- Continue this process until all edges have been used. This produces an Euler circuit. Since every edge is included and no edge is included more than once.
- Similar reasoning can be used to show that a graph with exactly two vertices of odd degree must have an Euler path connecting these two vertices of odd degree

Algorithm for Constructing an Euler Circuits

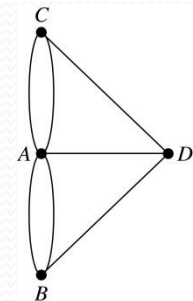
In our proof we developed this algorithms for constructing a Euler circuit in a graph with no vertices of odd degree.

```
procedure Euler(G: connected multigraph with all vertices of even degree)
  circuit := a circuit in G beginning at an arbitrarily chosen vertex with edges
             successively added to form a path that returns to this vertex.
  H := G with the edges of this circuit removed
  while H has edges
    subcircuit := a circuit in H beginning at a vertex in H that also is
                  an endpoint of an edge in circuit.
    H := H with edges of subcircuit and all isolated vertices removed
    circuit := circuit with subcircuit inserted at the appropriate vertex.
return circuit{circuit is an Euler circuit}
```

Necessary and Sufficient Conditions for Euler Circuits and Paths (*continued*)

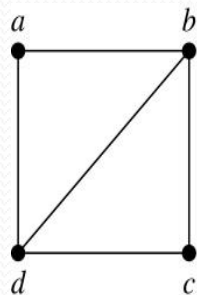
Theorem: A connected **multigraph** with at least two vertices has an **Euler circuit** if and only if each of its vertices has an even degree and it has an Euler path if and only if it has exactly two vertices of odd degree.

Example: Two of the vertices in the multigraph model of the Königsberg bridge problem have odd degree. Hence, there is no Euler circuit in this multigraph and it is impossible to start at a given point, cross each bridge exactly once, and return to the starting point.

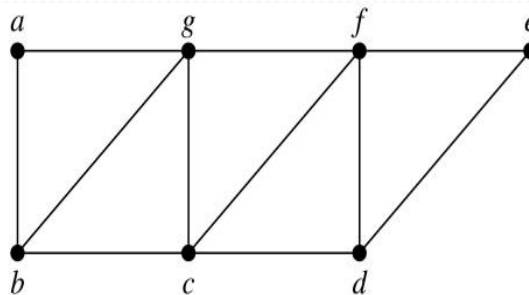


Euler Circuits and Paths

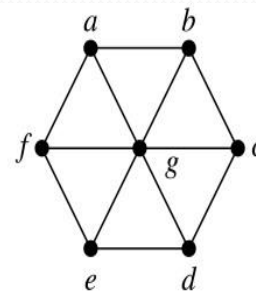
Example:



G_1



G_2



G_3

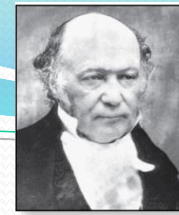
G_1 contains exactly two vertices of odd degree (b and d). Hence it has an Euler path, e.g., d, a, b, c, d, b .

G_2 has exactly two vertices of odd degree (b and d). Hence it has an Euler path, e.g., $b, a, g, f, e, d, c, g, b, c, f, d$.

G_3 has six vertices of odd degree. Hence, it does not have an Euler path.

Applications of Euler Paths and Circuits

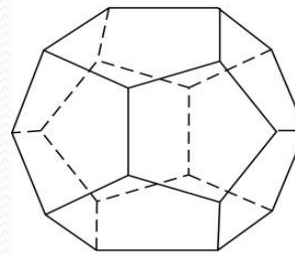
- Euler paths and circuits can be used to solve many practical problems such as finding a path or circuit that traverses each
 - street in a neighborhood,
 - road in a transportation network,
 - connection in a utility grid,
 - link in a communications network.
- Other applications are found in the
 - layout of circuits,
 - network multicasting,
 - molecular biology, where Euler paths are used in the sequencing of DNA.



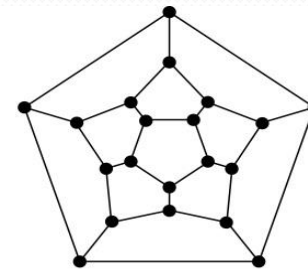
William Rowan
Hamilton
(1805- 1865)

Hamilton Paths and Circuits

- Euler paths and circuits contained every edge only once. Now we look at paths and circuits that contain every vertex exactly once.
- William Hamilton invented the *Icosian puzzle* in 1857. It consisted of a wooden dodecahedron (with 12 regular pentagons as faces), illustrated in (a), with a peg at each vertex, labeled with the names of different cities. String was used to plot a circuit visiting 20 cities exactly once
- The graph form of the puzzle is given in (b).

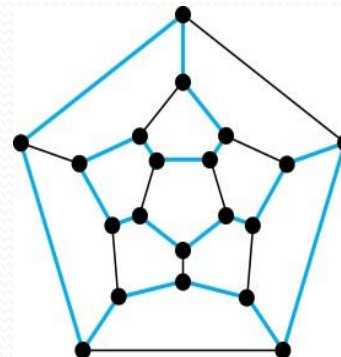


(a)



(b)

- The solution (a Hamilton circuit) is given here.



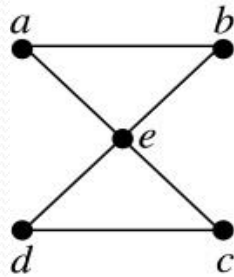
Hamilton Paths and Circuits

Definition: A simple path in a graph G that passes through every vertex exactly once is called a **Hamilton path**, and a simple circuit in a graph G that passes through every vertex exactly once is called a **Hamilton circuit**.

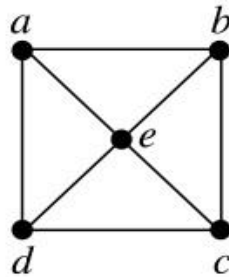
That is, a simple path $x_0, x_1, \dots, x_{n-1}, x_n$ in the graph $G = (V, E)$ is called a Hamilton path if $V = \{x_0, x_1, \dots, x_{n-1}, x_n\}$ and $x_i \neq x_j$ for $0 \leq i < j \leq n$, and the simple circuit $x_0, x_1, \dots, x_{n-1}, x_n, x_0$ (with $n > 0$) is a Hamilton circuit if $x_0, x_1, \dots, x_{n-1}, x_n$ is a Hamilton path.

Hamilton Paths and Circuits (*continued*)

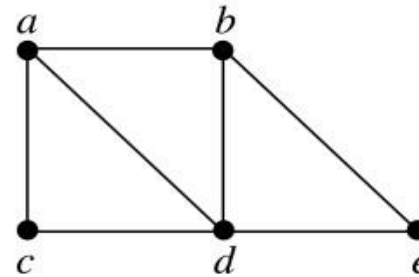
Example: Which of these simple graphs has a Hamilton circuit or, if not, a Hamilton path?



G_1



G_2



G_3

Solution: G_1 has a Hamilton circuit: a, b, c, d, e, a .

G_2 does not have a Hamilton circuit (Why?), but does have a Hamilton path : a, b, c, d .

G_3 does not have a Hamilton circuit, or a Hamilton path. Why?



Gabriel Andrew Dirac
(1925-1984)

Necessary Conditions for Hamilton Circuits

- Unlike for an Euler circuit, no simple necessary and sufficient conditions are known for the existence of a Hamilton circuit.
- However, there are some useful necessary conditions. We describe two of these now.

Dirac's Theorem: If G is a simple graph with $n \geq 3$ vertices such that the degree of every vertex in G is $\geq n/2$, then G has a Hamilton circuit.

Ore's Theorem: If G is a simple graph with $n \geq 3$ vertices such that $\deg(u) + \deg(v) \geq n$ for every pair of nonadjacent vertices, then G has a Hamilton circuit.



Øysten Ore
(1899-1968)

Applications of Hamilton Paths and Circuits

- Applications that ask for a path or a circuit that visits each intersection of a city, each place pipelines intersect in a utility grid, or each node in a communications network exactly once, can be solved by finding a Hamilton path in the appropriate graph.
- The famous *traveling salesperson problem* (TSP) asks for the shortest route a traveling salesperson should take to visit a set of cities. This problem reduces to finding a Hamilton circuit such that the total sum of the weights of its edges is as small as possible.
- A family of binary codes, known as *Gray codes*, which minimize the effect of transmission errors, correspond to Hamilton circuits in the n -cube Q_n . (See the text for details.)



الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

Chapter 11

Trees

Chapter Outlines

Introduction to Trees

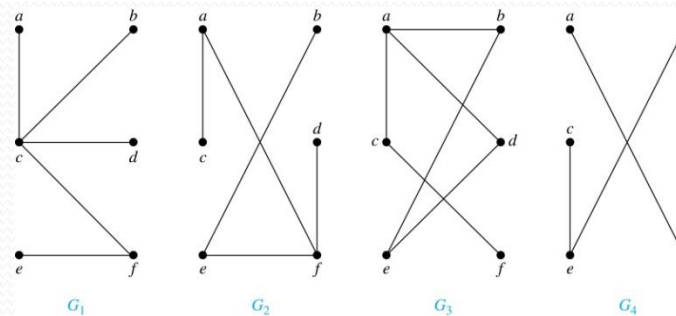
Introduction to Trees

Section 11.1

Trees

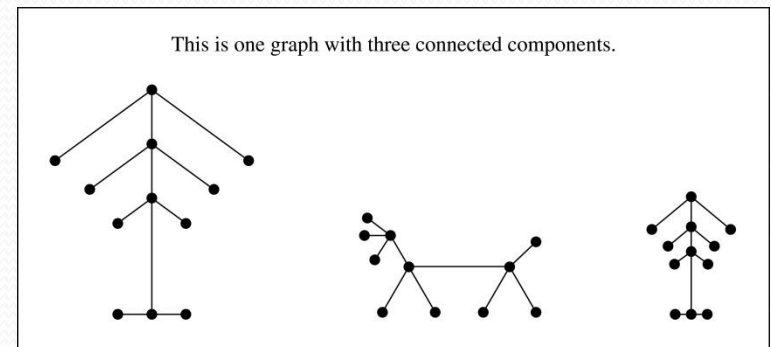
Definition: A **tree** is a **connected undirected graph** with no **simple circuits**.

Example: Which of these graphs are trees?



Solution: G_1 and G_2 are trees - both are connected and have no **simple circuits**. Because e, b, a, d, e is a simple circuit, G_3 is not a tree. G_4 is not a tree because it is **not connected**.

Definition: A **forest** is a graph that has no **simple circuit**, but is **not connected**. Each of the connected components in a forest is a tree.



Trees (*continued*)

Theorem: An undirected graph is a tree if and only if there is a **unique simple path** between any two of its vertices.

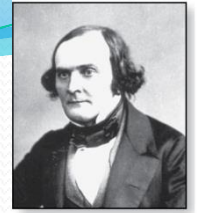
Proof: Assume that T is a tree. Then T is connected with no simple circuits. Hence, if x and y are distinct vertices of T , there is a simple path between them (by Theorem 1 of Section 10.4). This path must be unique - for if there were a second path, there would be a simple circuit in T (by Exercise 59 of Section 10.4). Hence, there is a unique simple path between any two vertices of a tree.

Now assume that there is a unique simple path between any two vertices of a graph T . Then T is connected because there is a path between any two of its vertices. Furthermore, T can have no simple circuits since if there were a simple circuit, there would be two paths between some two vertices.

Hence, a graph with a **unique simple path** between any two vertices is a tree.

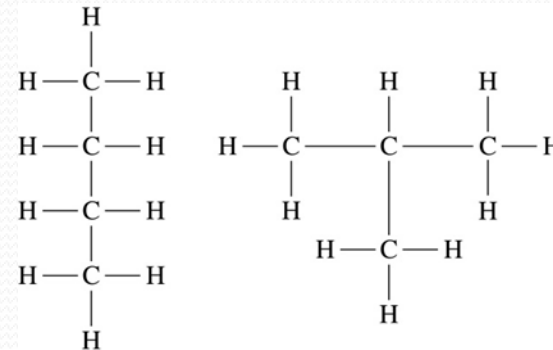


Arthur Cayley
(1821-1895)



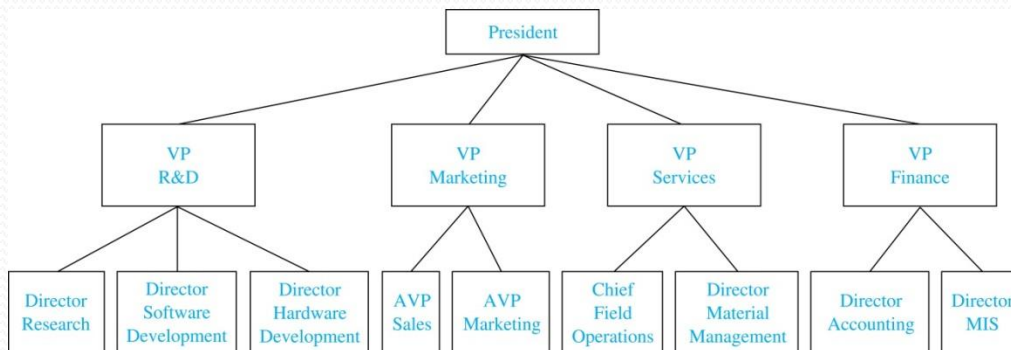
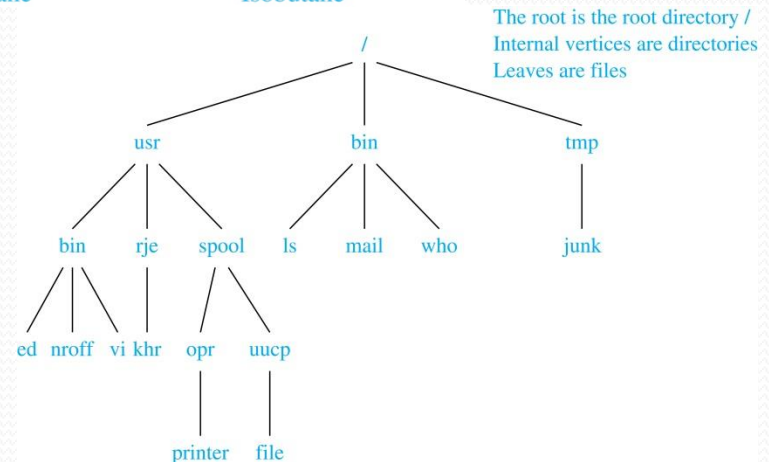
Trees as Models

- Trees are used as **models** in computer science, chemistry, geology, botany, psychology, and many other areas.
- Trees were introduced by the mathematician Cayley in 1857 in his work counting the number of isomers of saturated hydrocarbons. The two isomers of butane are shown at the right.
- The organization of a computer file system into directories, subdirectories, and files is naturally represented as a tree.
- Trees are used to represent the structure of organizations.



Butane

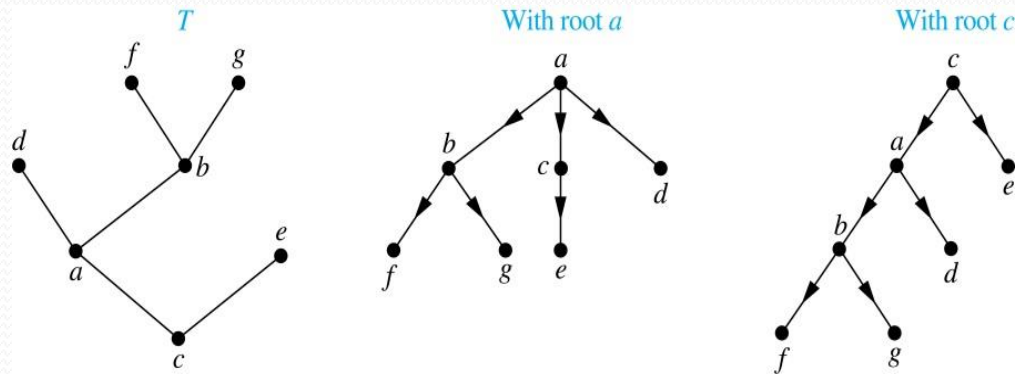
Isobutane



Rooted Trees

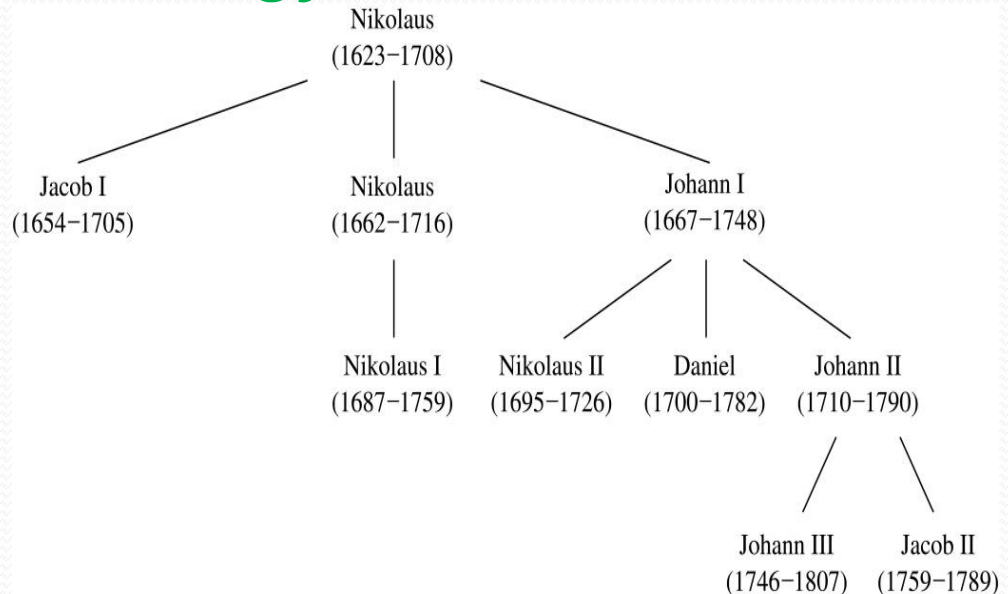
Definition: A **rooted tree** is a tree in which one vertex has been designated as the *root* and every edge is directed away from the root.

An **unrooted tree** is converted into different rooted trees when different vertices are chosen as the root.



Rooted Tree Terminology

- Terminology for rooted trees is a mix from botany and genealogy (such as this family tree of the Bernoulli family of mathematicians).

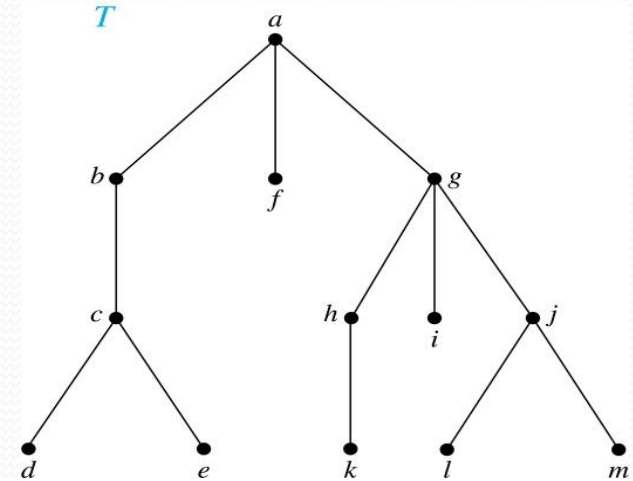


- If v is a vertex of a rooted tree other than the root, the **parent** of v is the unique vertex u such that there is a directed edge from u to v . When u is a parent of v , v is called a **child** of u . Vertices with the same parent are called **siblings**.
- The **ancestors** of a vertex are the vertices in the path from the root to this vertex, excluding the vertex itself and including the root. The **descendants** of a vertex v are those vertices that have v as an ancestor.
- A vertex of a rooted tree with no children is called a **leaf**. Vertices that have children are called **internal vertices**.
- If a is a vertex in a tree, the **subtree** with a as its root is the subgraph of the tree consisting of a and its descendants and all edges incident to these descendants.

Terminology for Rooted Trees

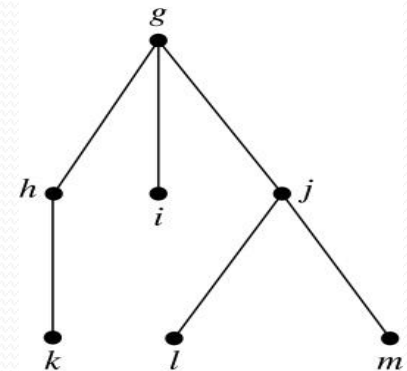
Example: In the rooted tree T (with root a):

- (i) Find the parent of c , the children of g , the siblings of h , the ancestors of e , and the descendants of b .
- (ii) Find all internal vertices and all leaves.
- (iii) What is the subtree rooted at G ?



Solution:

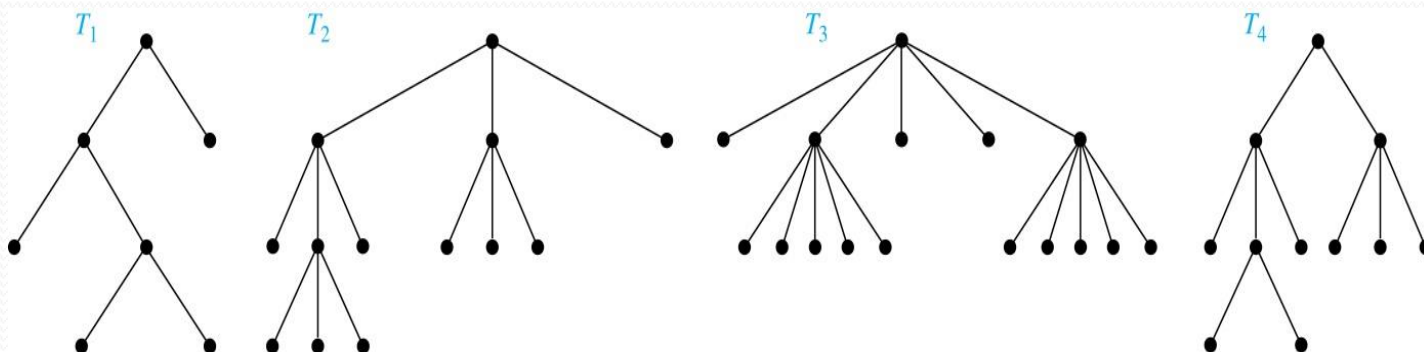
- (i) The parent of c is b . The children of g are h , i , and j . The siblings of h are i and j . The ancestors of e are c , b , and a . The descendants of b are c , d , and e .
- (ii) The internal vertices are a , b , c , g , h , and j . The leaves are d , e , f , i , k , l , and m .
- (iii) We display the subtree rooted at g .



m -ary Rooted Trees

Definition: A rooted tree is called an **m -ary tree** if every internal vertex has no more than m children. The tree is called a **full m -ary tree** if every internal vertex has exactly m children. An m -ary tree with $m = 2$ is called a **binary tree**.

Example: Are the following rooted trees full m -ary trees for some positive integer m ?



Solution: T_1 is a full binary tree because each of its internal vertices has two children. T_2 is a full 3-ary tree because each of its internal vertices has three children. In T_3 each internal vertex has five children, so T_3 is a full 5-ary tree. T_4 is not a full m -ary tree for any m because some of its internal vertices have two children and others have three children.

Ordered Rooted Trees

Definition: An **ordered rooted tree** is a rooted tree where the children of each internal vertex are ordered.

- We draw ordered rooted trees so that the children of each internal vertex are shown in order from left to right.

Definition: A **binary tree** is an ordered rooted tree where each internal vertex has **at most** two children. If an internal vertex of a binary tree has two children, the first is called the **left child** and the second the **right child**. The tree rooted at the left child of a vertex is called the **left subtree** of this vertex, and the tree rooted at the right child of a vertex is called the **right subtree** of this vertex.

Example: Consider the binary tree T .

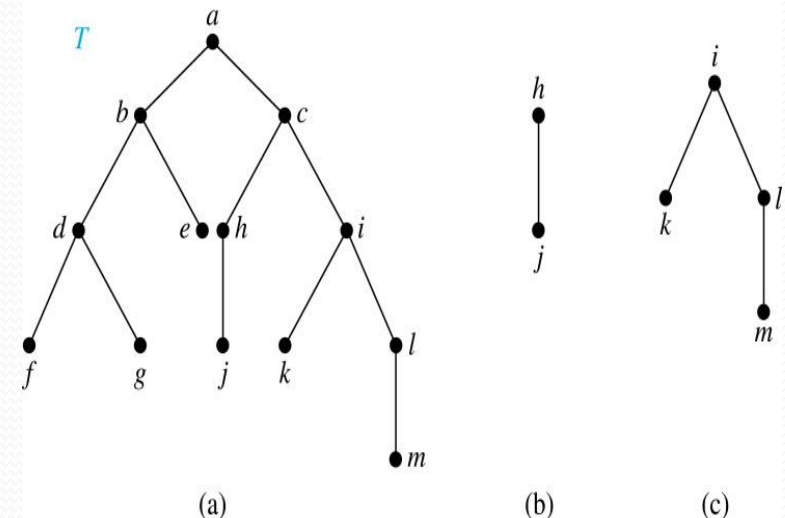
(i) What are the left and right children of d ?

(ii) What are the left and right subtrees of c ?

Solution:

(i) The left child of d is f and the right child is g .

(ii) The left and right subtrees of c are displayed in (b) and (c).



Properties of Trees

Theorem 2: A tree with n vertices has $n - 1$ edges.

Proof (by mathematical induction):

BASIS STEP: When $n = 1$, a tree with one vertex has no edges. Hence, the theorem holds when $n = 1$.

INDUCTIVE STEP: Assume that every tree with k vertices has $k - 1$ edges.

Suppose that a tree T has $k + 1$ vertices and that v is a leaf of T . Let w be the parent of v . Removing the vertex v and the edge connecting w to v produces a tree T' with k vertices. By the inductive hypothesis, T' has $k - 1$ edges. Because T has one more edge than T' , we see that T has k edges. This completes the inductive step.



Counting Vertices in Full m -Ary Trees

Theorem 3: A full m -ary tree with i internal vertices has $n = mi + 1$ vertices.

Proof : Every vertex, except the root, is the child of an internal vertex. Because each of the i internal vertices has m children, there are mi vertices in the tree other than the root. Hence, the tree contains $n = mi + 1$ vertices.



Counting Vertices in Full m -Ary Trees (continued)

Theorem 4: A full m -ary tree with

(i) n vertices has $i = (n - 1)/m$ internal vertices and $l = [(m - 1)n + 1]/m$ leaves,

(ii) i internal vertices has $n = mi + 1$ vertices and $l = (m - 1)i + 1$ leaves,

(iii) l leaves has $n = (ml - 1)/(m - 1)$ vertices and $i = (l - 1)/(m - 1)$ internal vertices.

*proofs of
parts (ii) and
(iii) are left as
exercises*

Proof (of part i): Solving for i in $n = mi + 1$ (from Theorem 3) gives $i = (n - 1)/m$. Since each vertex is either a leaf or an internal vertex, $n = l + i$. By solving for l and using the formula for i , we see that

$$l = n - i = n - (n - 1)/m = [(m - 1)n + 1]/m .$$

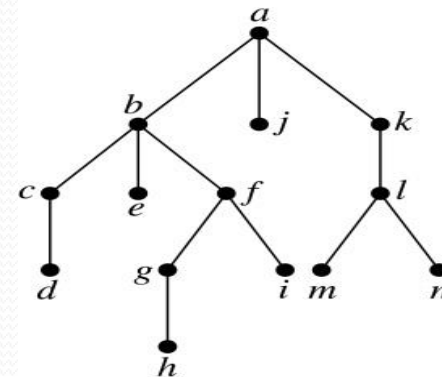


Level of vertices and height of trees

- When working with trees, we often want to have rooted trees where the subtrees at each vertex contain paths of approximately the same length.
- To make this idea precise we need some definitions:
 - The **level** of a vertex v in a rooted tree is the length of the unique path from the root to this vertex.
 - The **height** of a rooted tree is the maximum of the levels of the vertices.

Example:

- Find the level of each vertex in the tree to the right.
- What is the height of the tree?



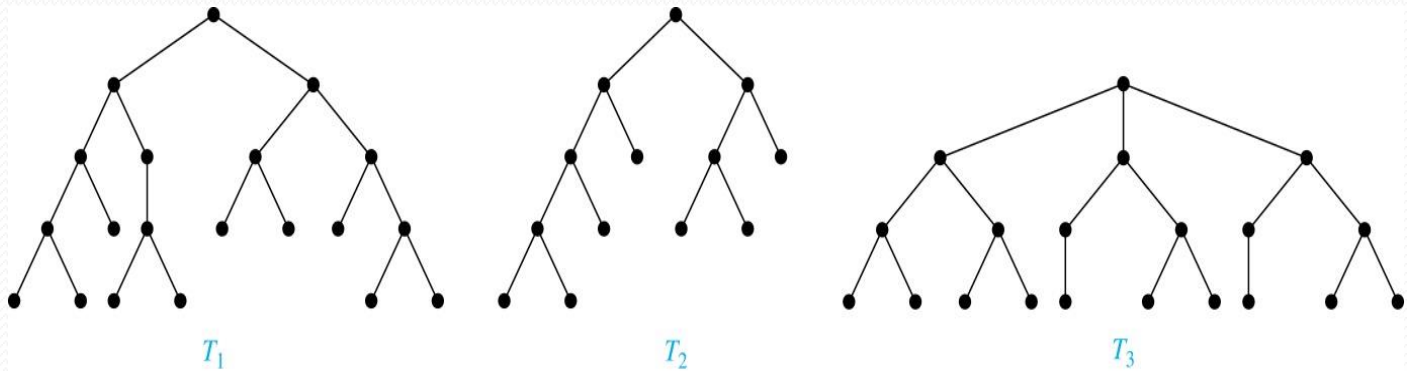
Solution:

- The root a is at level 0. Vertices b , j , and k are at level 1. Vertices c , e , f , and l are at level 2. Vertices d , g , i , m , and n are at level 3. Vertex h is at level 4.
- The height is 4, since 4 is the largest level of any vertex.

Balanced m -Ary Trees

Definition: A rooted m -ary tree of height h is **balanced** if all leaves are at levels h or $h - 1$.

Example: Which of the rooted trees shown below is balanced?



Solution: T_1 and T_3 are balanced, but T_2 is not because it has leaves at levels 2, 3, and 4.