Discrete Mathematics (151)

Department of Mathematics
College of Sciences
King Saud University

Chapter 4: Graphs

4.1 Graphs and Graph Models (10.1 in book).

DEFINITION 1

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

Remark

The set of vertices V of a graph G may be infinite. A graph with an infinite vertex set or an infinite number of edges is called an **infinite graph**, and in comparison, a graph with a finite vertex set and a finite edge set is called a **finite graph**. In this book we will usually consider only finite graphs.

Graphs and Graph Models.

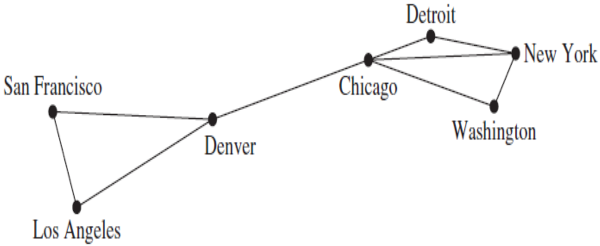

Figure 1: Example of Graph.

DEFINITION 2

A **directed graph** (or **digraph**) (V ,E) consists of a nonempty set of vertices V and a set of directed edges (or arcs) E. Each directed 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.

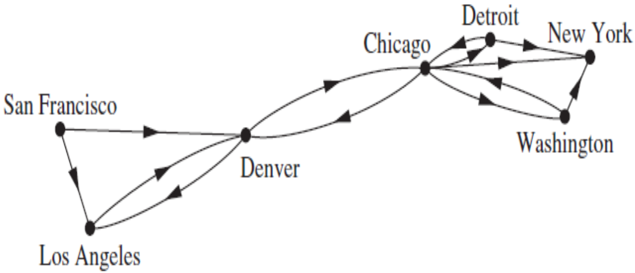

Figure 2: Example of Digraph.

4.2 Graph Terminology and Special Types of Graphs (10.2 in book).

Graph Terminology and Special Types of Graphs.

- We introduce some of the basic vocabulary of graph theory in this section.
- We will use this vocabulary later in this chapter when we solve many different types of problems.
- One such problem involves determining whether a graph can be drawn in the plane so that no two of its edges cross.
- Another example is deciding whether there is a one-to-one correspondence between the vertices of two graphs that produces a one-to-one correspondence between the edges of the graphs.
- We will also introduce several important families of graphs often used as examples and in models.
- Several important applications will be described where these special types of graphs arise.

Graph Terminology and Special Types of Graphs.

Basic Terminology

First, we give some terminology that describes the vertices and edges of undirected graphs.

DEFINITION 1

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

We will also find useful terminology describing the set of vertices adjacent to a particular vertex of a graph.

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, $N(A) = \bigcup_{v \in A} N(v)$.

Basic Terminology

To keep track of how many edges are incident to a vertex, we make the following definition.

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 twice to the degree of that vertex. The degree of the vertex v is denoted by $deg(v)$.

Example 1

What are the degrees and what are the neighborhoods of the vertices in the graphs G displayed in Figure 3?

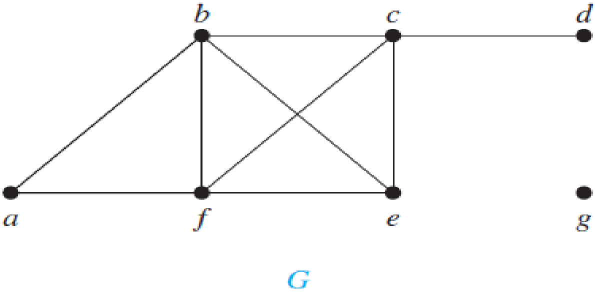

Figure 3: The Undirected Graph G

Solution: In G, $deg(a) = 2$, $deg(b) = deg(c) = deg(f) = 4$, $deg(d) = 1$, $deg(e) = 3$, and $deg(g) = 0$. The neighborhoods of these vertices are $N(a) = \{b, f\}$, $N(b) = \{a, c, e, f\}$,
$N(c) = \{b, d, e, f\}$, $N(d) = \{c\}$, $N(e) = \{b, c, f\}$, $N(f) = \{a, b, c, e\}$, and $N(g) = \emptyset$.

Graph Terminology and Special Types of Graphs.

- A vertex of degree zero is called **isolated**. It follows that an isolated vertex is not adjacent to any vertex. Vertex g in graph G in Example 1 is isolated.
- A vertex is **pendant** if and only if it has degree one. Consequently, a pendant vertex is adjacent to exactly one other vertex. Vertex d in graph G in Example 1 is pendant.

- What do we get when we add the degrees of all the vertices of a graph $G = (V, E)$?
- Each edge contributes two to the sum of the degrees of the vertices because an edge is incident with exactly two (possibly equal) vertices.

THEOREM 1: THE HANDSHAKING THEOREM

Let $G = (V, E)$ be an undirected graph with m edges. Then

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

Example 2 (3 in book)

How many edges are there in a graph with 10 vertices each of degree six?
Solution: Because the sum of the degrees of the vertices is $6.10 = 60$, it follows that $2m = 60$ where m is the number of edges. Therefore, $m = 30$.

Theorem 1 shows that the sum of the degrees of the vertices of an undirected graph is even. This simple fact has many consequences, one of which is given as Theorem 2.

THEOREM 2

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

Terminology for graphs with directed edges reflects the fact that edges in directed graphs have directions.

DEFINITION 4

When (u, v) is an edge of the graph G with directed edges, u is said to be **adjacent to** v and v is said to be **adjacent from** u. The vertex u is called the **initial vertex** of (u, v), and v is called the **terminal** or **end vertex** of (u, v). The initial vertex and terminal vertex of a loop are the same.

Because the edges in graphs with directed edges are ordered pairs, the definition of the degree of a vertex can be refined to reflect the number of edges with this vertex as the initial vertex and as the terminal vertex.

DEFINITION 5

In a graph with directed edges the **in-degree of a vertex** v, denoted by $deg^{-}(v)$, is the number of edges with v as their terminal vertex. The **out-degree of a vertex** v, denoted by $deg^{+}(v)$, 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 this vertex.)

Example 3 (4 in book)

Find the in-degree and out-degree of each vertex in the graph G with directed edges shown in Figure 4.

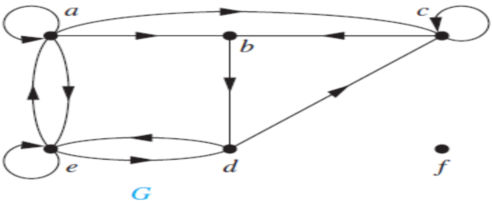

Figure 4: The directed Graph G

Solution: The in-degrees in G are
$deg^{-}(a) = 2, deg^{-}(b) = 2, deg^{-}(c) = 3, deg^{-}(d) = 2, deg^{-}(e) = 3$, and $deg^{-}(f) = 0$.
The out-degrees are
$deg^{+}(a) = 4, deg^{+}(b) = 1, deg^{+}(c) = 2, deg^{+}(d) = 2, deg^{+}(e) = 3$, and $deg^{+}(f) = 0$.

Because each edge has an initial vertex and a terminal vertex, the sum of the in-degrees and the sum of the out-degrees of all vertices in a graph with directed edges are the same. Both of these sums are the number of edges in the graph.

THEOREM 3

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

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

Some Special Simple Graphs

We will nowintroduce several classes of simple graphs. These graphs are often used as examples and arise in many applications.

Example 4: Complete Graphs

A **complete graph** on n vertices, denoted by K_n, is a simple graph that contains exactly one edge between each pair of distinct vertices. The graphs K_n, for $n = 1, 2, 3, 4, 5, 6$, are displayed in Figure 5. A simple graph for which there is at least one pair of distinct vertex not connected by an edge is called **noncomplete**.

A complete graph on n vertices has exactly $\frac{n(n-1)}{2}$ edges.

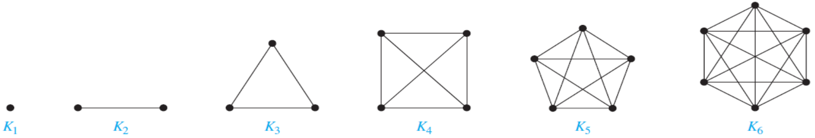

Figure 5: The Graphs K_n for $1 \leq n \leq 6$.

Example 5: Cycles

A cycle C_n, $n \geq 3$, consists of n vertices $v_1, v_2, \ldots, v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}$, and $\{v_n, v_1\}$. The cycles C_3, C_4, C_5, and C_6 are displayed in Figure 6.

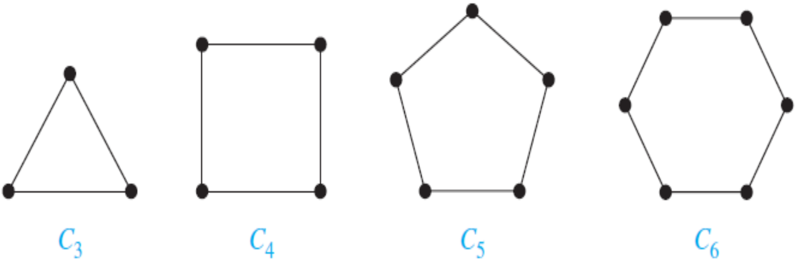

Figure 6: The Cycles C_3, C_4, C_5, and C_6

Example 7: Wheels

We obtain a **wheel** W_n when we add an additional vertex to a cycle C_n, for $n \geq 3$, and connect this new vertex to each of the n vertices in C_n, by new edges. The wheels W_3, W_4, W_5, and W_6 are displayed in Figure 7.

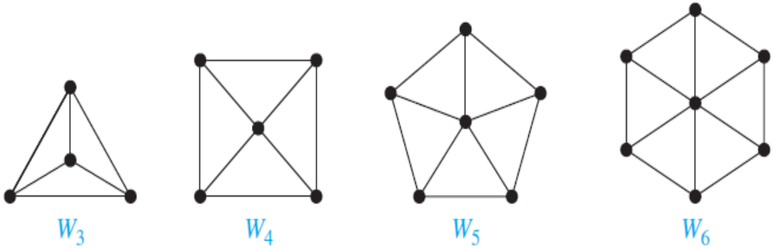

Figure 7: The Wheels W_3, W_4, W_5, and W_6

Bipartite Graphs

DEFINITION 6

A simple graph G is called **bipartite** if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2). When this condition holds, we call the pair (V_1, V_2) a **bipartition** of the vertex set V of G.

In Example 8 we will show that C_6 is bipartite, and in Example 9 we will show that K_3 is not bipartite.

Example 8 (9 in book)

C_6 is bipartite, as shown in Figure 6, because its vertex set can be partitioned into the two sets $V_1 = \{v_1, v_3, v_5\}$ and $V2 = \{v_2, v_4, v_6\}$, and every edge of C_6 connects a vertex in V_1 and a vertex in V_2.

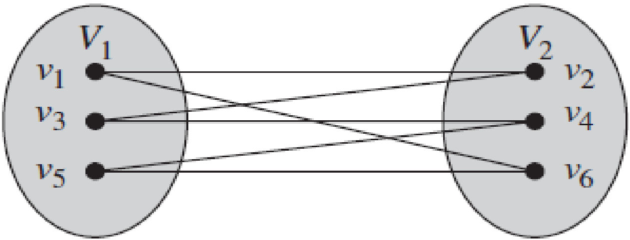

Figure 8: Showing That C_6 Is Bipartite.

Example 9 (10 in book)

K_3 is not bipartite. To verify this, note that if we divide the vertex set of K_3 into two disjoint sets, one of the two sets must contain two vertices. If the graph were bipartite, these two vertices could not be connected by an edge, but in K_3 each vertex is connected to every other vertex by an edge.

THEOREM

A simple graph is bipartite if and only if it does not contain an odd-length cycle.

Example 10 (11 in book)

Are the graphs G and H displayed in Figure 9 bipartite?

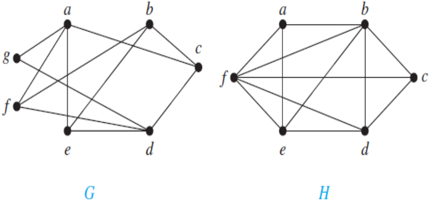

Figure 9: The Undirected Graphs G and H.

Solution: Graph G is bipartite because its vertex set is the union of two disjoint sets, $\{a, b, d\}$ and $\{c, e, f, g\}$, and each edge connects a vertex in one of these subsets to a vertex in the other subset. (Note that for G to be bipartite it is not necessary that every vertex in $\{a, b, d\}$ be adjacent to every vertex in $\{c, e, f, g\}$. For instance, b and g are not adjacent.)
Graph H is not bipartite because its vertex set cannot be partitioned into two subsets so that edges do not connect two vertices from the same subset. (We verify this by considering the vertices a, b, and f.)

Graph Terminology and Special Types of Graphs.

Theorem 4 provides a useful criterion for determining whether a graph is bipartite.

THEOREM 4

A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.

Example 11: Complete Bipartite Graphs

A **complete bipartite** graph $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets of m and n vertices, respectively with an edge between two vertices if and only if one vertex is in the first subset and the other vertex is in the second subset. The complete bipartite graphs $K_{2,3}$, $K_{3,3}$, $K_{3,5}$, and $K_{2,6}$ are displayed in Figure 10.
A complete bipartite graph $K_{m,n}$ have exactly $n.m$ edges.

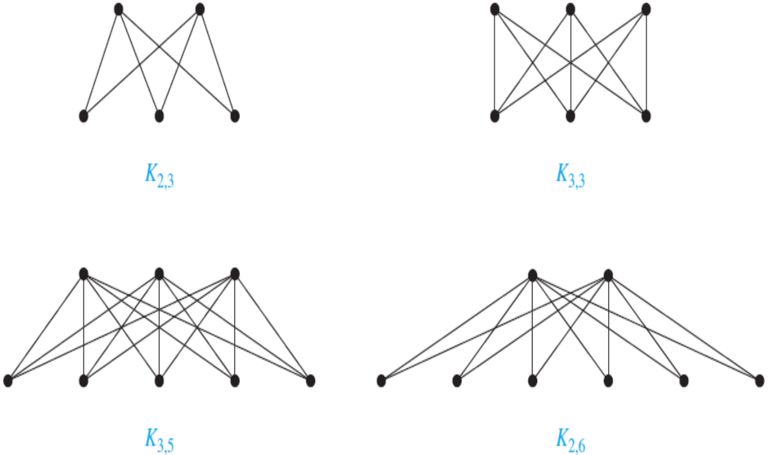

Figure 10: Some Complete Bipartite Graphs.

New Graphs from Old

DEFINITION 7

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

Given a set of vertices of a graph, we can form a subgraph of this graph with these vertices and the edges of the graph that connect them.

DEFINITION 8

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 of this edge are in W.

Example 12 (18 in book)

The graph G shown in Figure 11 is a subgraph of K_5. If we add the edge connecting c and e to G, we obtain the subgraph induced by $W = \{a, b, c, e\}$.

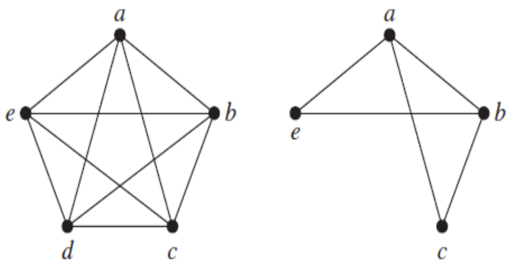

Figure 11: A Subgraph of K_5.

Complement

The **complement** of G is $\overline{G} = (V, E(\overline{G}))$.

$$x \neq y \in V \,/\, (x, y) \in E(\overline{G}) \Longleftrightarrow (x, y) \notin E(G)$$

If G is a Graph with n vertices, then the sum of the number of edge of G and those of $\overline{G}$ is $\frac{n(n-1)}{2}$

$$|E(G)| + |E(\overline{G})| = \frac{n(n-1)}{2}$$

4.3 Representing Graphs and Graph Isomorphism (10.3 in book).

Introduction

- There are many useful ways to represent graphs. As we will see throughout this chapter, in working with a graph it is helpful to be able to choose its most convenient representation.
- In this section we will show how to represent graphs in several different ways.
- Sometimes, two graphs have exactly the same form, in the sense that there is a one-to-one correspondence between their vertex sets that preserves edges. In such a case, we say that the two graphs are **isomorphic**.
- Determining whether two graphs are isomorphic is an important problem of graph theory that we will study in this section.

Representing Graphs and Graph Isomorphism.

Adjacency Matrices Suppose that $G = (V, E)$ is a simple graph where $|V| = n$. Suppose that the vertices of G are listed arbitrarily as $v_1, v_2, \ldots, v_n$. The **adjacency matrix** A **(or** A_G**)** of G, with respect to this listing of the 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 its adjacency matrix is $A = [a_{ij}]$, then

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

Example 1 (3 in book)

Use an adjacency matrix to represent the graph shown in Figure 12.

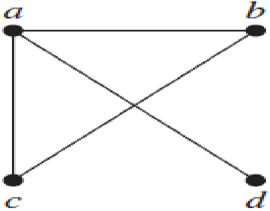

Figure 12: Simple Graph.

Solution: We order the vertices as a, b, c, d. The matrix representing this graph is

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

Example 2 (4 in book)

Draw a graph with the adjacency matrix

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

with respect to the ordering of vertices a, b, c, d.

Solution: A graph with this adjacency matrix is shown in Figure 13.

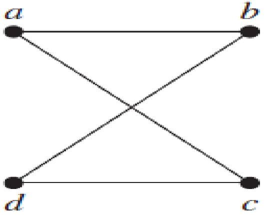

Figure 13: A Graph with the Given Adjacency Matrix.

Incidence Matrices Another common way to represent graphs is to use incidence matrices. Let $G = (V, E)$ be an undirected graph. Suppose that $v_1, v_2, \ldots, v_n$ are the vertices and $e_1, e_2, \ldots, e_m$ are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix $M = [m_{ij}]$, where

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

Example 3 (6 in book)

Represent the graph shown in Figure 14 with an incidence matrix..

Representing Graphs and Graph Isomorphism.

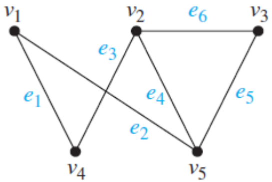

Figure 14: Simple Graph.

Solution: The incidence matrix is

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

Isomorphism of Graphs

DEFINITION 1

The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are **isomorphic** if there exists 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**.

Example 4 (8 in book)

Show that the graphs $G = (V, E)$ and $H = (W, F)$, displayed in Figure 15, are isomorphic.

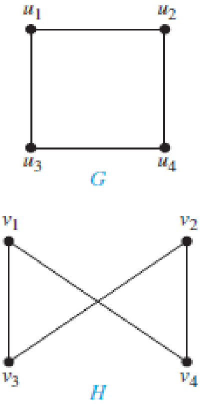

Figure 15: The Graphs G and H.

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.
To see that this correspondence preserves adjacency, 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, and 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.

Representing Graphs and Graph Isomorphism.

Sometimes it is not hard to show that two graphs are not isomorphic.

- Isomorphic simple graphs must have the same number of vertices, because there is a one-to-one correspondence between the sets of vertices of the graphs.
- Isomorphic simple graphs also must have the same number of edges, because the one-to-one correspondence between vertices establishes a one-to-one correspondence between edges.
- In addition, the degrees of the vertices in isomorphic simple graphs must be the same. That is, a vertex v of degree d in G must correspond to a vertex $f(v)$ of degree d in H, because a vertex w in G is adjacent to v if and only if $f(v)$ and $f(w)$ are adjacent in H.

Example 5 (9 in book)

Show that the graphs displayed in Figure 16 are not isomorphic.

Solution: Both G and H have five vertices and six edges. However, H has a vertex of degree one, namely, e, whereas G has no vertices of degree one. It follows that G and H are not isomorphic.

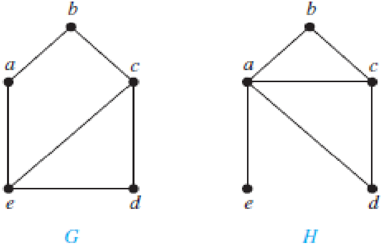

Figure 16: The Graphs G and H.

Representing Graphs and Graph Isomorphism.

Example 6 (10 in book)

Determine whether the graphs shown in Figure 17 are isomorphic.

Solution: The graphs G and H both have eight vertices and 10 edges. They also both have four vertices of degree two and four of degree three. Because these invariants all agree, it is still conceivable that these graphs are isomorphic.

However, G and H are not isomorphic. To see this, note that because $deg(a) = 2$ in G, a must correspond to either t, u, x, or y in H, because these are the vertices of degree two in H. However, each of these four vertices in H is adjacent to another vertex of degree two in H, which is not true for a in G.

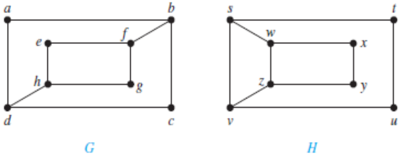

Figure 17: The Graphs G and H.

4.4 Connectivity (10.4 in book).

Introduction

- Many problems can be modeled with paths formed by traveling along the edges of graphs.
- For instance, the problem of determining whether a message can be sent between two computers using intermediate links can be studied with a graph model.
- Problems of efficiently planning routes for mail delivery, garbage pickup, diagnostics in computer networks, and so on can be solved using models that involve paths in graphs.

Connectivity.

Path Informally, 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 edges.

DEFINITION 1

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 $e_1, \ldots, e_n$ of G for which there exists a sequence $x_0 = u, x_1, \ldots, x_{n-1}, x_n = v$ of vertices such that e_i has, for $i = 1, \ldots, n$, the endpoints x_{i-1} and x_i . When the graph is simple, we denote this path by its vertex sequence $x_0, x_1, \cdots, x_n$ (because listing these vertices uniquely determines the path). The path is a **circuit** if it begins and ends at the same vertex, that is, if $u = v$, and has length greater than zero. The path or circuit is said to **pass through** the vertices $x_1, x_2, \cdots, x_{n-1}$ or **traverse** the edges $e_1, e_2, \ldots, e_n$. A path or circuit is simple if it does not contain the same edge more than once.

Connectivity.

A path or circuit is **simple** if it does not contain the same edge more than once. There is considerable variation of terminology concerning the concepts defined in Definition 1.

Remarks

- In some books, the term walk is used instead of path, where a **walk** is defined to be an alternating sequence of vertices and edges of a graph, $v_0, e_1, v_1, e_2, \ldots, v_{n1}, e_n, v_n$, where v_{i-1} and v_i are the endpoints of e_i for $i = 1, 2, \ldots, n$.
- When this terminology is used, **closed walk** is used instead of circuit to indicate a walk that begins and ends at the same vertex, and **trail** is used to denote a walk that has no repeated edge (replacing the term simple path). When this terminology is used, the terminology path is often used for a trail with no repeated vertices, conflicting with the terminology in Definition 1.

Connectivity.

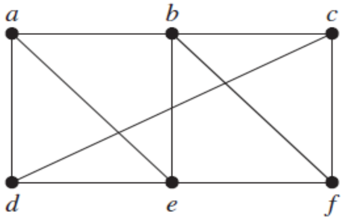

Figure 18: A Simple Graph.

Example 1

In the simple graph shown in Figure 18, a, d, c, f, e is a simple path of length 4, because $\{a, d\}$, $\{d, c\}$, $\{c, f\}$, and $\{f, e\}$ are all edges. However, d, e, c, a is not a path, because $\{e, c\}$ is not an edge. Note that b, c, f, e, b is a circuit of length 4 because $\{b, c\}, \{c, f\}, \{f, e\}$, and $\{e, b\}$ are edges, and this path begins and ends at b. The path a, b, e, d, a, b, which is of length 5, is not simple because it contains the edge $\{a, b\}$ twice.

Connectivity.

Connectedness in Undirected Graphs: When does a computer network have the property that every pair of computers can share information, if messages can be sent through one or more intermediate computers? When a graph is used to represent this computer network, where vertices represent the computers and edges represent the communication links, this question becomes: When is there always a path between two vertices in the graph?

DEFINITION 2

An undirected graph is called **connected** if there is a path between every pair of distinct vertices of the graph. 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.

Thus, any two computers in the network can communicate if and only if the graph of this network is connected.

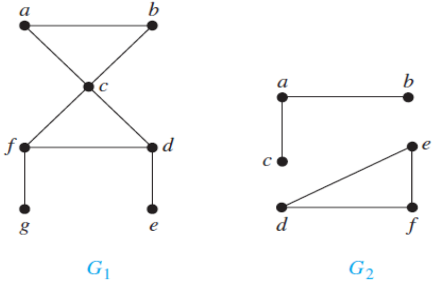

Figure 19: The Graphs G_1 and G_2.

Example 2 (4 in book)

The graph G_1 in Figure 19 is connected, because for every pair of distinct vertices there is a path between them (we can verify this). However, the graph G_2 in Figure 19 is not connected. For instance, there is no path in G_2 between vertices a and d.

Connectivity.

THEOREM 1:

There is a simple path between every pair of distinct vertices of a connected undirected graph.

CONNECTED COMPONENTS

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. That is, a connected component of a graph G is a maximal 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.

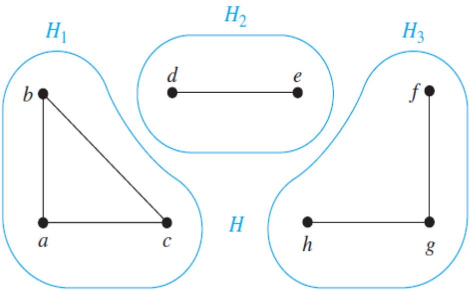

Figure 20: The Graph H and Its Connected Components H_1,H_2, and H_3.

Example 3 (5 in book)

What are the connected components of the graph H shown in Figure 20?
Solution: The graph H is the union of three disjoint connected subgraphs H_1,H_2, and H_3, shown in Figure 20. These three subgraphs are the connected components of H.

Connectivity.

Sometimes the removal from a graph of a vertex and all incident edges produces a subgraph with more connected components. Such vertices are called **cut vertices** (or **articulation points**). The removal of a cut vertex from a connected graph produces a subgraph that is not connected. Analogously, an edge whose removal produces a graph with more connected components than in the original graph is called a **cut edge** or **bridge**. Note that in a graph representing a computer network, a cut vertex and a cut edge represent an essential router and an essential link that cannot fail for all computers to be able to communicate.

Connectivity.

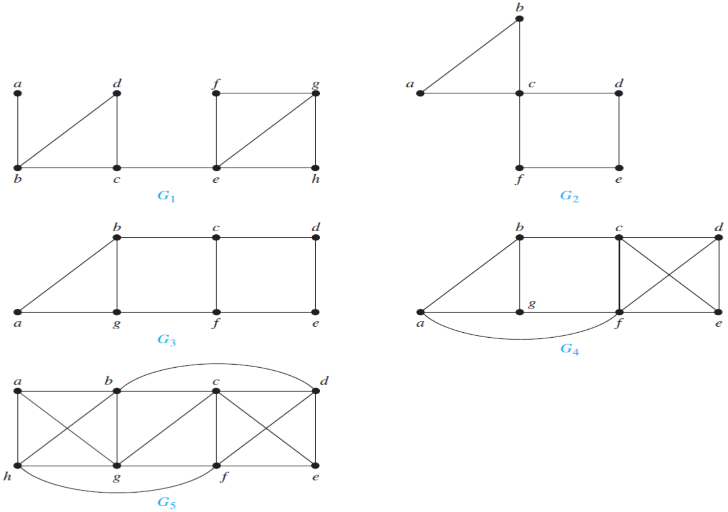

Figure 21: Some Connected Graphs.

Connectivity.

Paths and Isomorphism: There are several ways that paths and circuits can help determine whether two graphs are isomorphic. For example, the existence of a simple circuit of a particular length is a useful invariant that can be used to show that two graphs are not isomorphic. In addition, paths can be used to construct mappings that may be isomorphisms. As we mentioned, a useful isomorphic invariant for simple graphs is the existence of a simple circuit of length k, where k is a positive integer greater than 2. Example 6 illustrates how this invariant can be used to show that two graphs are not isomorphic.

Example 6 (13 in book)

Determine whether the graphs G and H shown in Figure 23 are isomorphic.

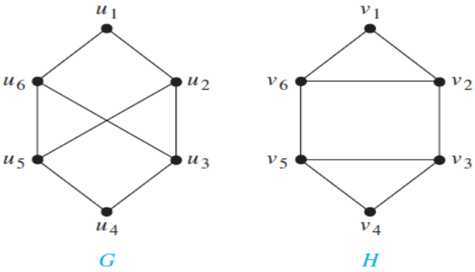

Figure 22: The Graphs G and H.

Solution: Both G and H have six vertices and eight edges. Each has four vertices of degree three, and two vertices of degree two. So, the three invariants-number of vertices, number of edges, and degrees of vertices all agree for the two graphs. However, H has a simple circuit of length three, namely, v_1, v_2, v_6, v_1, whereas G has no simple circuit of length three, as can be determined by inspection (all simple circuits in G have length at least four). Because the existence of a simple circuit of length three is an isomorphic invariant, G and H are not isomorphic.

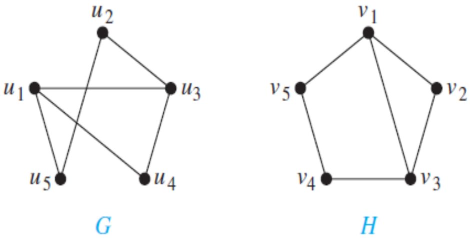

Figure 23: The Graphs G and H.

Example 7 (14 in book)

Determine whether the graphs G and H shown in Figure 24 are isomorphic.
Solution: Both G and H have five vertices and six edges, both have two vertices of degree three and three vertices of degree two, and both have a simple circuit of length three, a simple circuit of length four, and a simple circuit of length five. Because all these isomorphic invariants agree, G and H may be isomorphic.

Connectivity.

Counting Paths Between Vertices: The number of paths between two vertices in a graph can be determined using its adjacency matrix.

THEOREM 2

Let G be a graph with adjacency matrix A with respect to the ordering $v_1, v_2, \cdots, v_n$ of the vertices of the graph (with directed or undirected edges, and loops allowed). The number of different paths of length r from v_i to v_j , where r is a positive integer, equals the $(i, j)^{th}$ entry of A^r.

Connectivity.

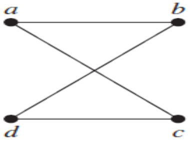

Figure 24: The Graph G.

Example 8 (15 in book)

How many paths of length four are there from a to d in the simple graph G in Figure 25?

Solution: The adjacency matrix of G (ordering the vertices as a, b, c, d) is

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \text{ and } A^4 \begin{bmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{bmatrix}$$

Hence, the number of paths of length four from a to d is 8 (the $(1, 4)$ th entry of A^4.

Connectivity.

1. a, b, a, b, d
2. a, b, a, c, d
3. a, b, d, b, d
4. a, b, d, c, d
5. a, c, a, b, d
6. a, c, a, c, d
7. a, c, d, b, d
8. a, c, d, c, d