Mathematical concepts for computer science

Graph Isomorphism

- **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.**

Representing Graphs

- One way to represent a graph without multiple edges is to **list all the edges of this graph**.
- Another way to represent a graph with no multiple edges is to use **adjacency lists**, **which specify the vertices that are adjacent to each vertex** of the graph.

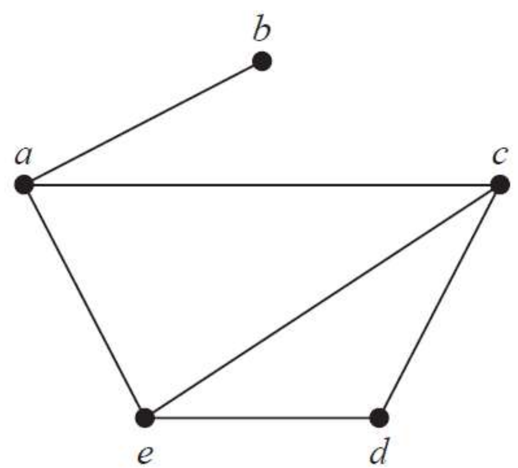


FIGURE 1 A Simple Graph.

TABLE 1 An Adjacency List for a Simple Graph.

Vertex	*Adjacent Vertices*
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

Representing Graphs

- One way to represent a graph without multiple edges is to **list all the edges of this graph**.
- Another way to represent a graph with no multiple edges is to use **adjacency lists**, **which specify the vertices that are adjacent to each vertex** of the graph.

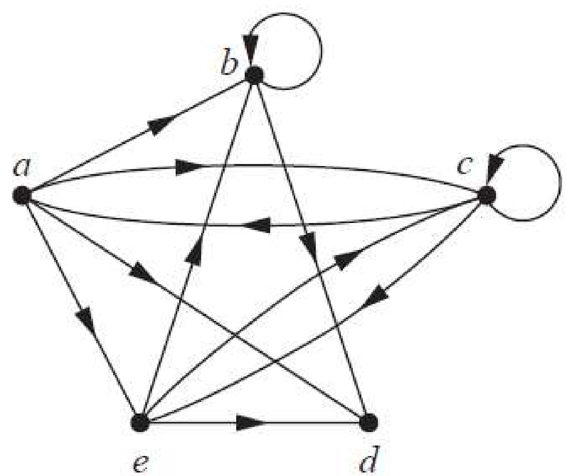


FIGURE 2 A Directed Graph.

Representing Graphs

- One way to represent a graph without multiple edges is to **list all the edges of this graph**.
- Another way to represent a graph with no multiple edges is to use **adjacency lists**, **which specify the vertices that are adjacent to each vertex** of the graph.

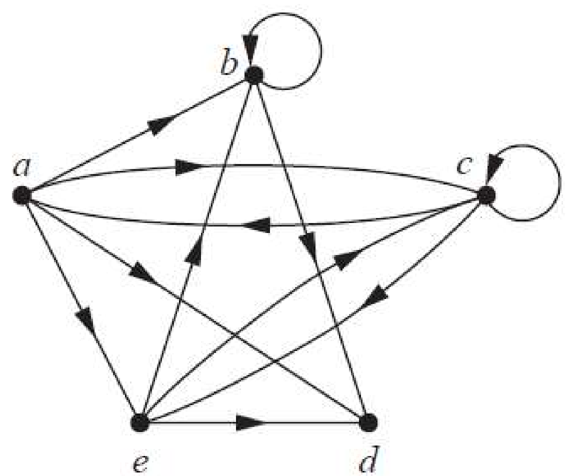


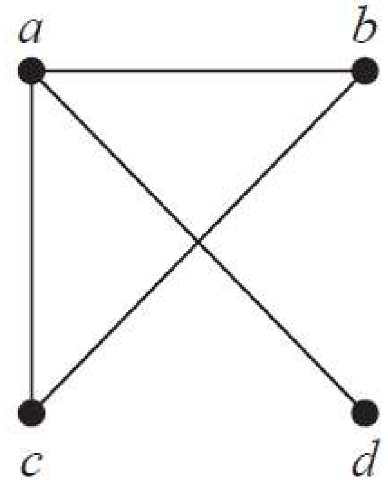
FIGURE 2 A Directed Graph.

TABLE 2 An Adjacency List for a Directed Graph.

Initial Vertex	*Terminal Vertices*
a	b, c, d, e
b	b, d
c	a, c, e
d	
e	b, c, d

Adjacency Matrices

- **Matrices** are commonly used to represent graphs

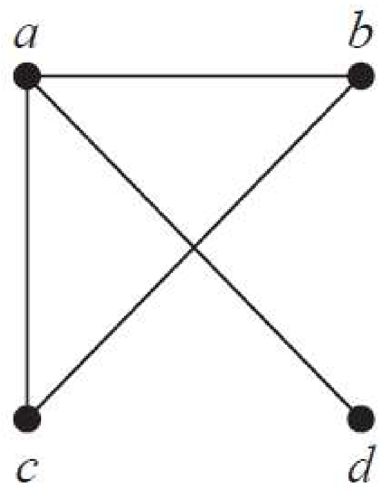


$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

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

Adjacency Matrices

- **Matrices** are commonly used to represent graphs



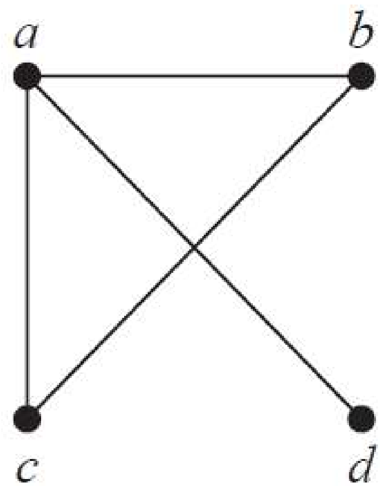
$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

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

Draw a graph with the adjacency matrix

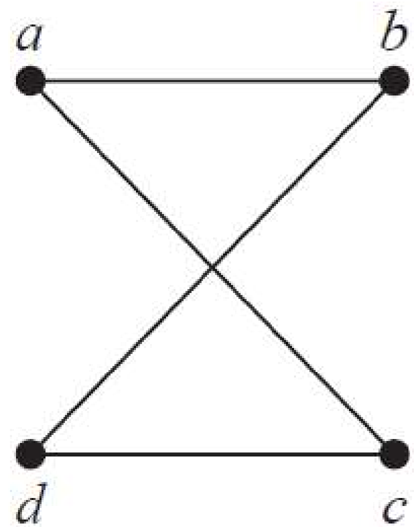
Adjacency Matrices

- **Matrices** are commonly used to represent graphs



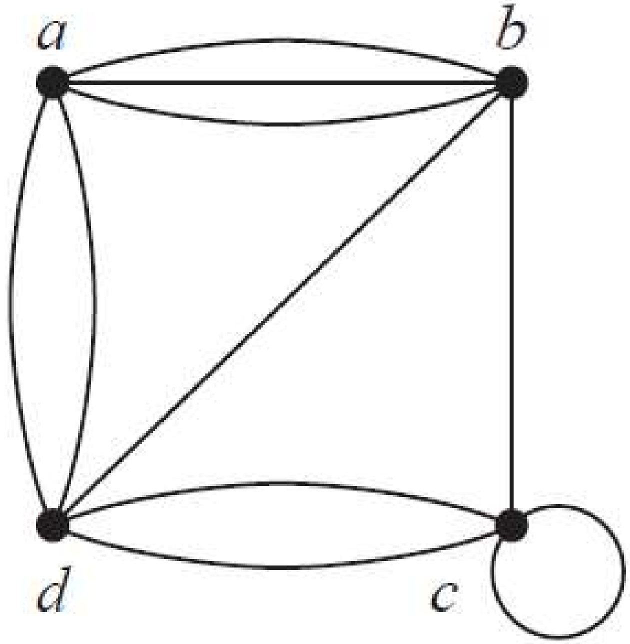
$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

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



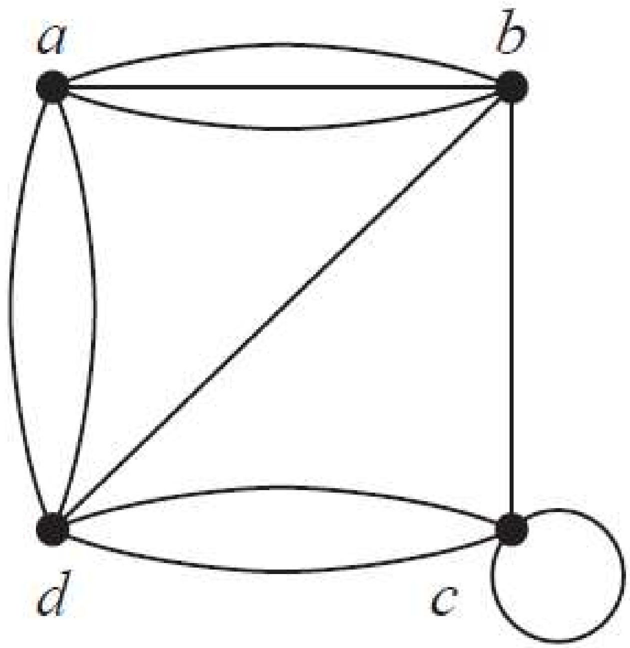
Adjacency Matrices

- Use an adjacency matrix to represent the pseudo-graph



Adjacency Matrices

- Use an adjacency matrix to represent the pseudo-graph



$$\begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

TRADE-OFFS BETWEEN ADJACENCY LISTS AND ADJACENCY MATRICES

- When a simple graph contains **relatively few edges**, that is, when it is **sparse,** it is usually **preferable to use adjacency lists** rather than an adjacency matrix to represent the graph.
- If a simple graph is **dense,** that is, suppose that **it contains many edges,** such as a graph that contains more than half of all possible edges. In this case, using an **adjacency matrix** to represent the graph

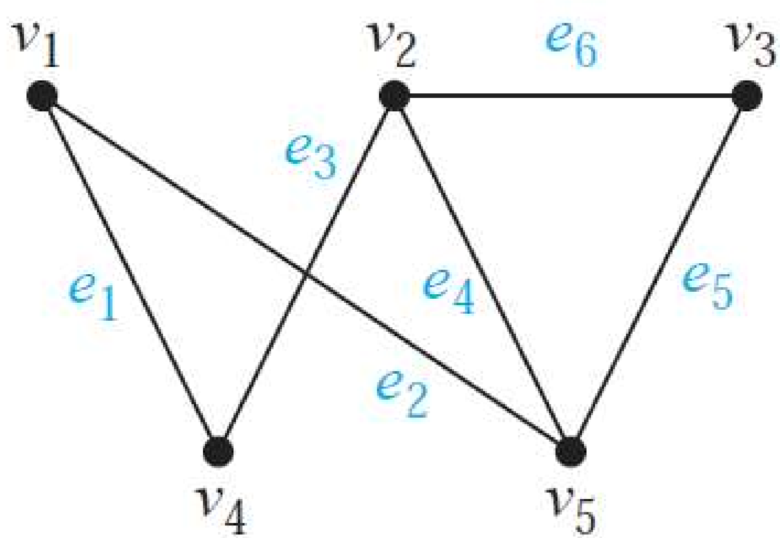
Incidence Matrices

- Let G = (V ,E) be an undirected graph. Suppose that *v1, v2, . . . , vn* are the vertices and *e1, e2, . . . , em* are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the *n* × *m* matrix **M = [*mij*],** where

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

Incidence Matrices

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

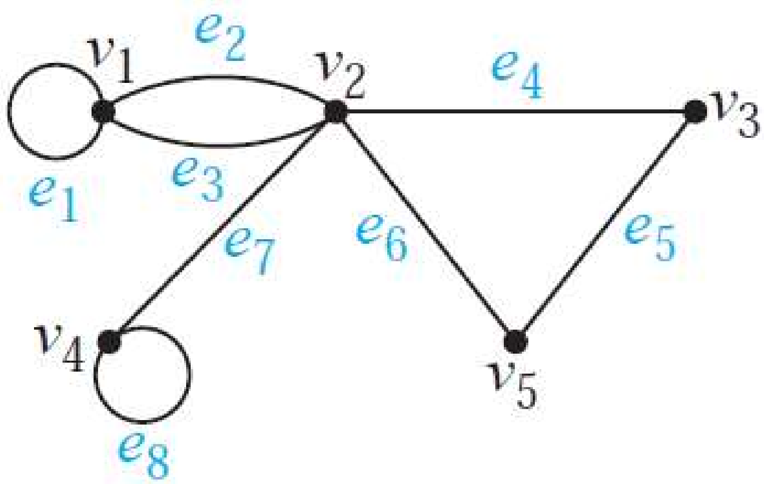


$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}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

Incidence Matrices

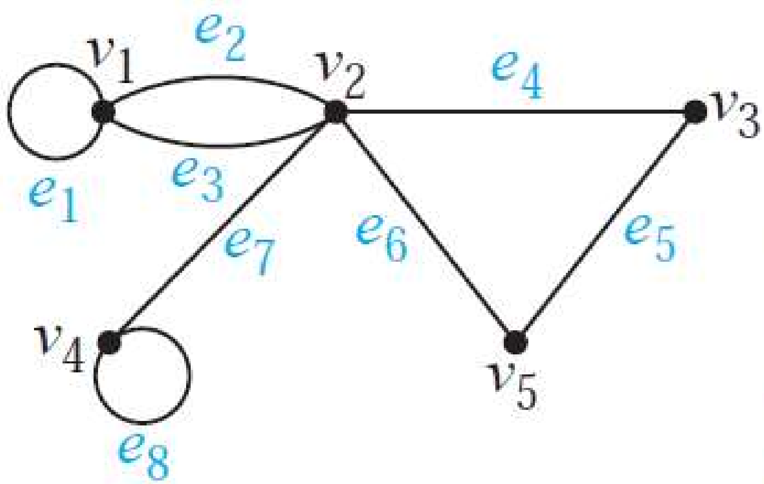
$$m_{ij} = \begin{cases} 1 & \text{when edge } e_j \text{ is incident with } v_i, \\ 0 & \text{otherwise.} \end{cases}$$

Represent the pseudo-graph using an incidence matrix



Incidence Matrices

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



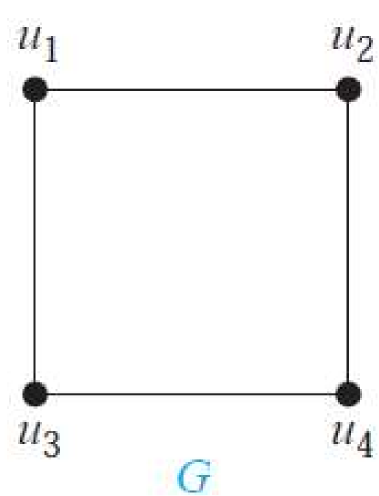
$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

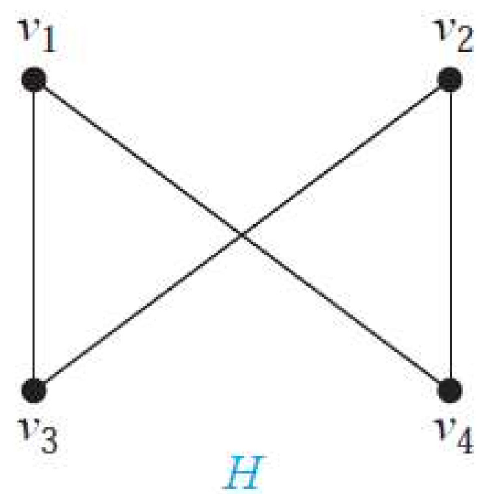
Isomorphism of Graphs

- The simple graphs **G1 = (V1,E1)** and **G2 = (V2,E2)** are isomorphic **if there exists a one to one and onto function** f from V1 to V2 with the property that **a and b are adjacent in G1 if and only if f (a) and f (b) are adjacent in G2**, for all a and b in V1. Such a function f is called an **isomorphism**.
- Two simple graphs that are not isomorphic are called non-isomorphic.

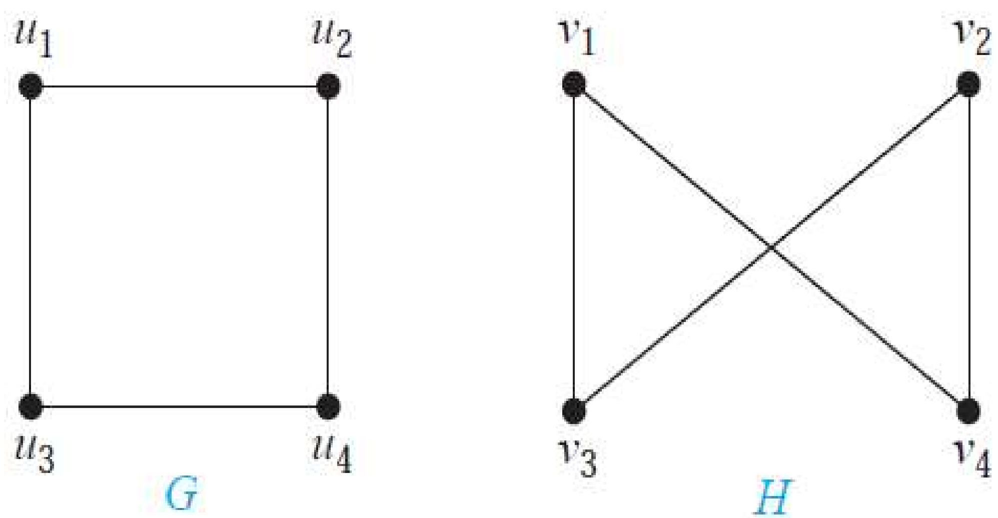
Isomorphism of Graphs

- Show that the graphs G = (V ,E) and H = (W, F) are isomorphic





Isomorphism of Graphs



Step 1: **one to one and on to**

The function f with **f (u1) = v1, f (u2) = v4, f (u3) = v3, and f (u4) = v2** is a one to one correspondence between V and W.

Step 2: **preserves adjacency**

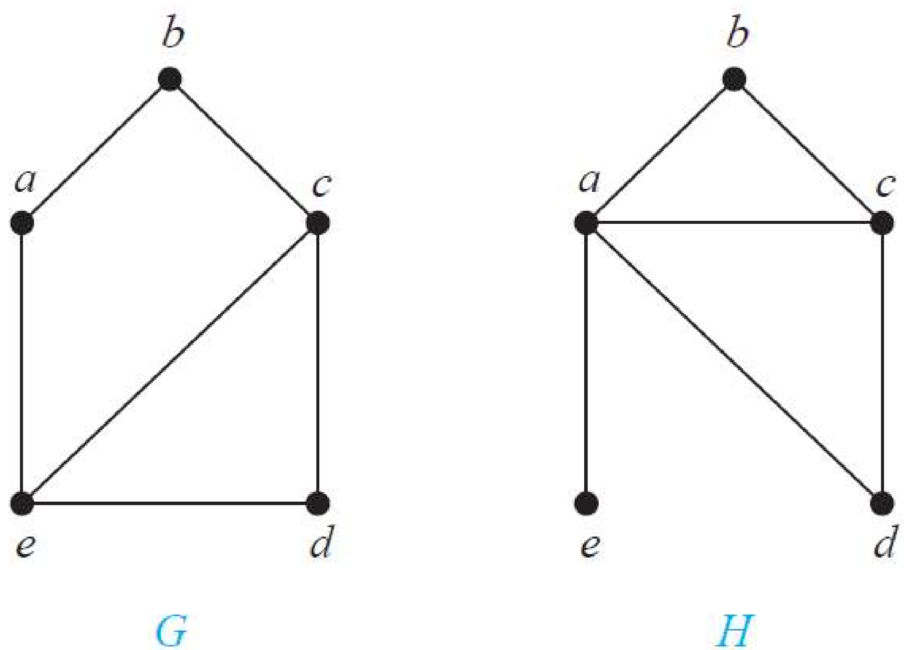
Adjacent vertices in G are u1 and u2, **u1 and u3**, u2 and u4, and u3 and u4, and each of the pairs f (u1) = v1 and f (u2) = v4, **f (u1) = v1 and f (u3) = v3**, f (u2) = v4 and f (u4) = v2, and **f (u3) = v3 and f (u4) = v2** consists of two adjacent vertices in H.

Determining whether Two Simple Graphs are Isomorphic

- There are **n! possible one-to-one correspondences** between the vertex sets of two simple graphs with **n vertices**. Testing each such correspondence to see whether it preserves adjacency and non-adjacency is **impractical if n is at all large**.
- Sometimes we can show that **two graphs are not isomorphic if we can find a property only one of the two graphs has**, **but that is preserved by isomorphism**.
- A property preserved by isomorphism of graphs is called a graph invariant.
- For instance, **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.

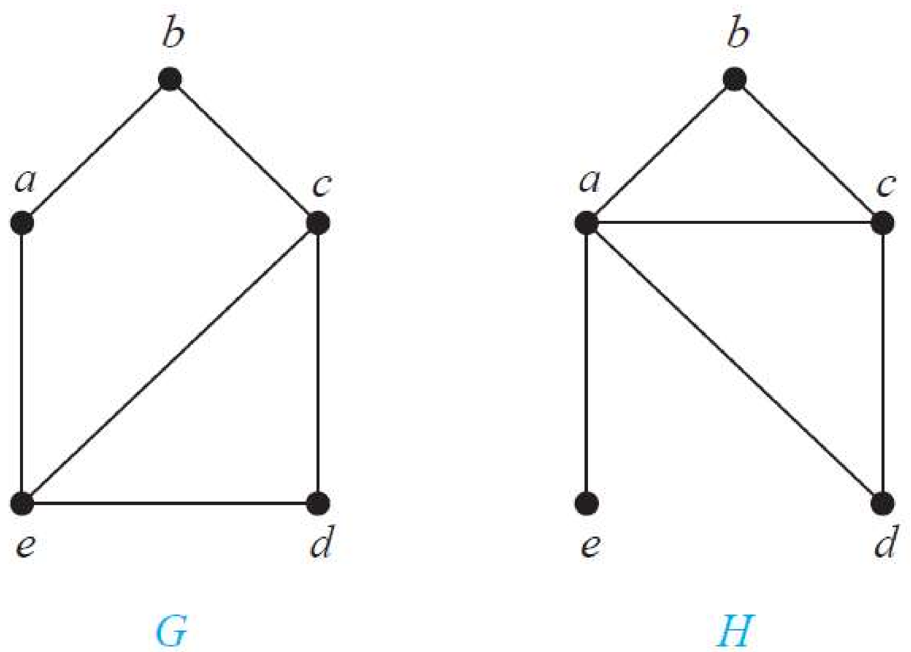
Determining whether Two Simple Graphs are Isomorphic

- Show that the graphs displayed are not isomorphic.



Determining whether Two Simple Graphs are Isomorphic

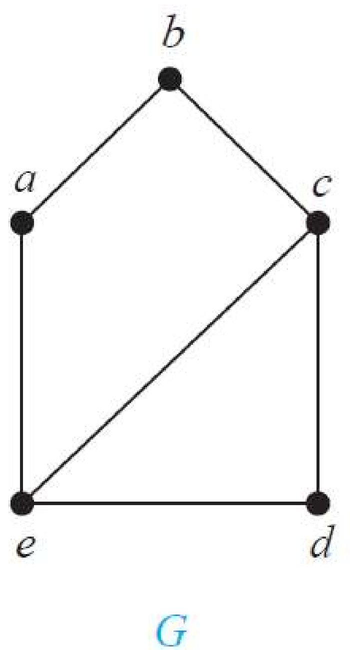
- Show that the graphs displayed are not isomorphic.

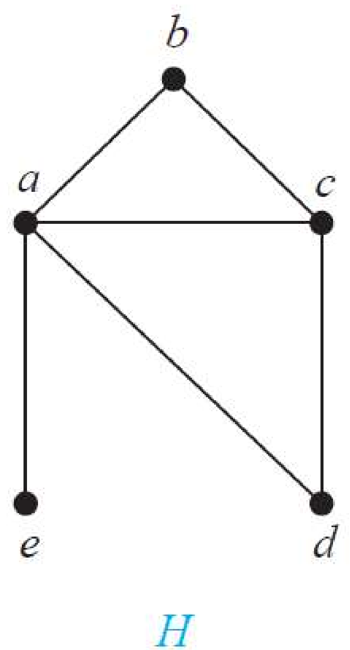


Both G and H have five vertices and six edges. However, H has a vertex of degree one,
namely, e, whereas G has no vertices of degree one. It follows that G and H are not isomorphic.

Determining whether Two Simple Graphs are Isomorphic

- Show that the graphs displayed are not isomorphic.

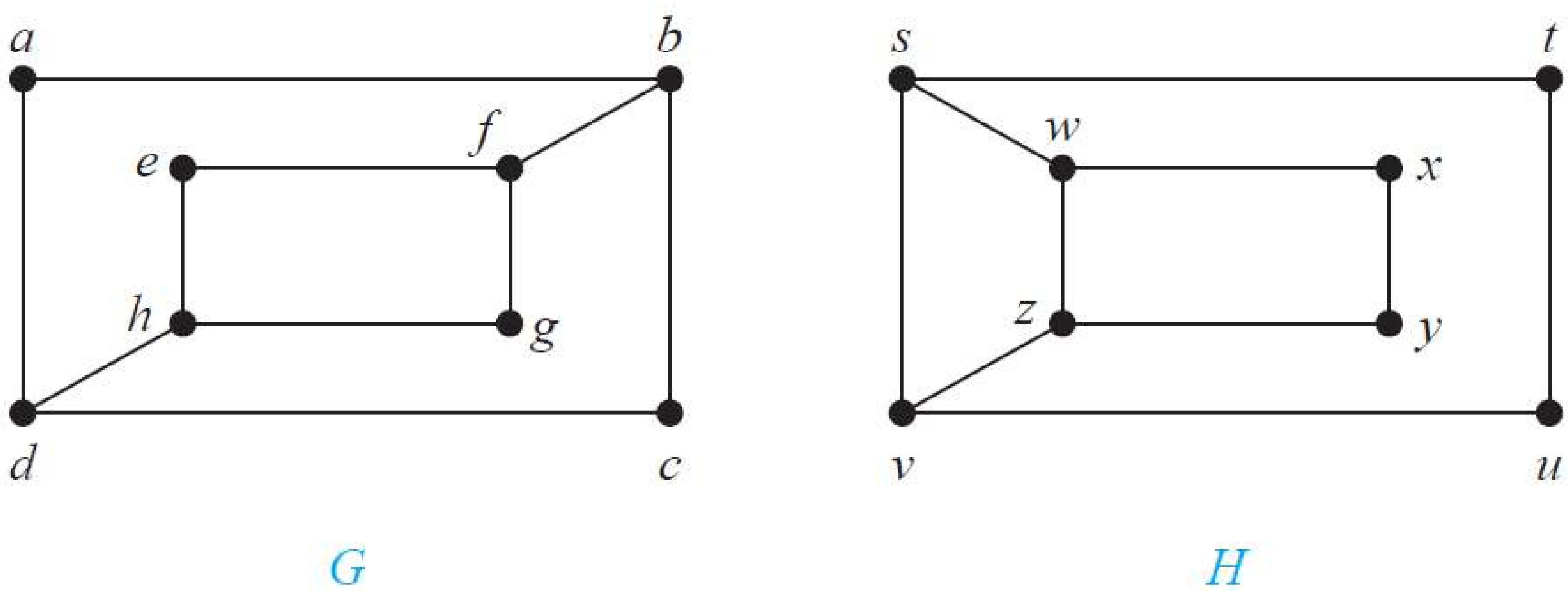




Both G and H have five vertices and six edges. However, H has a vertex of degree one,
namely, e, whereas G has no vertices of degree one. It follows that G and H are not isomorphic.

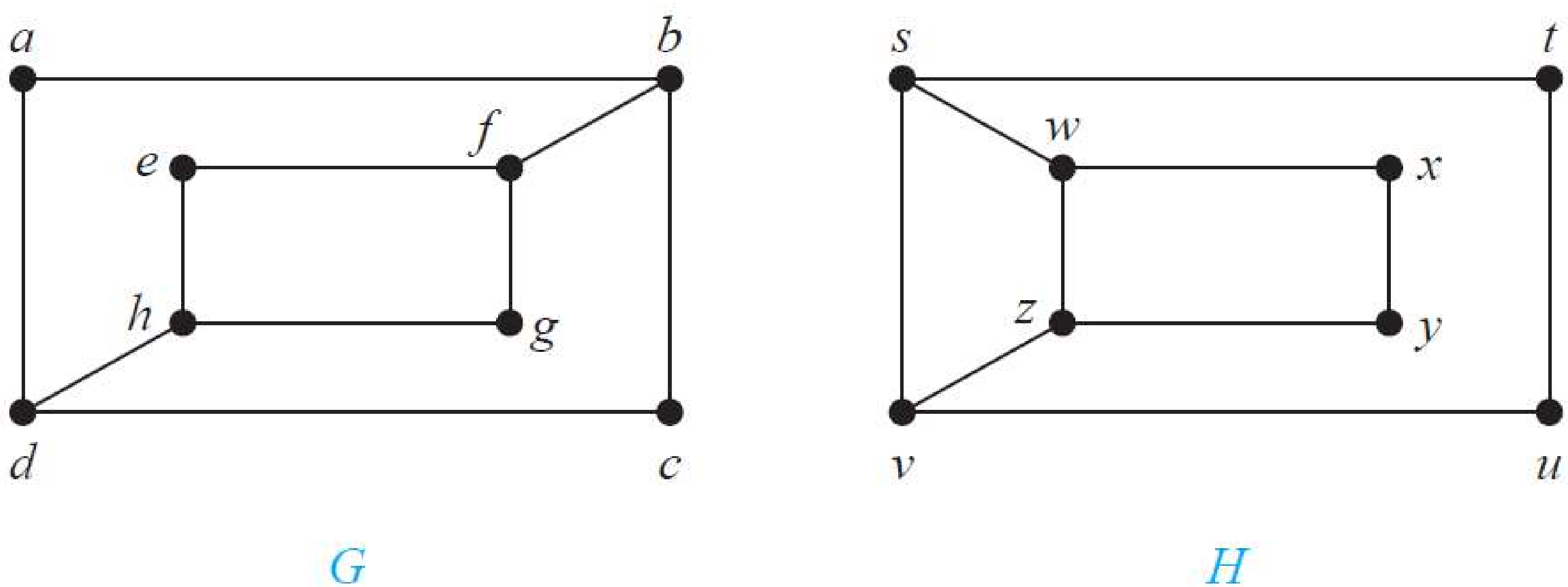
Determining whether Two Simple Graphs are Isomorphic

- Determine whether the graphs are isomorphic



Determining whether Two Simple Graphs are Isomorphic

- Determine whether the graphs are isomorphic



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.

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**.

Paths and walks

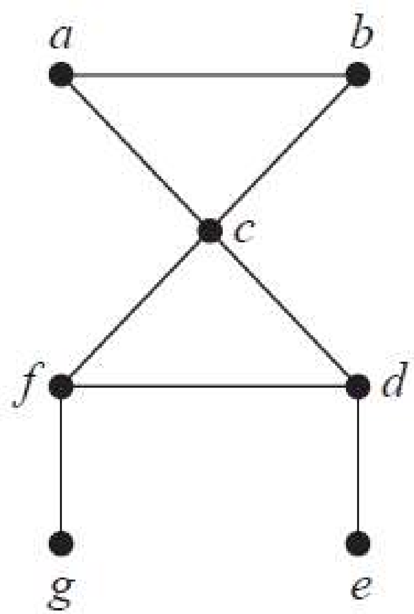
- Informally, a **path is a sequence of edges that begins at a vertex of a graph and travels from vertex to vertex along edges of the graph**.
- Let **n** be a nonnegative integer and **G** an undirected graph. A **path of length n from u to v** in G is a **sequence of n edges $e_1, \ldots, e_n$ of G** 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.

Paths and walks

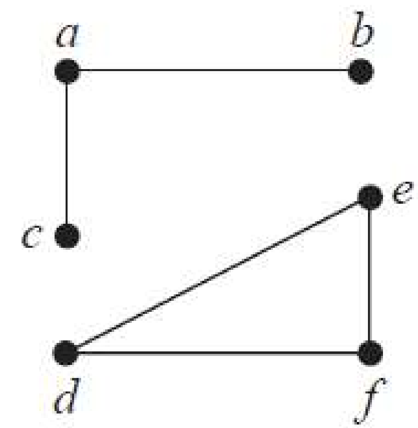
- When the **graph is simple**, we denote this path by its vertex sequence **x0, x1, . . . , xn.**
- The path is a circuit if it **begins and ends at the same vertex**, that is, if u = v, and has **length greater than zero**.
- A **path or circuit is simple** if it **does not contain the same edge more than once**.
- **Trail** is used to denote a **walk** that has **no repeated edge**

Connectedness in Undirected Graphs

- 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.



G_1



G_2

THEOREM

- There is a simple path between every pair of distinct vertices of a connected undirected graph.
- In graph theory a **simple path** is a path in a graph **which does not have repeating vertices**.

THEOREM

- ***Proof:***

Let **u** and **v** be two distinct vertices of the connected undirected graph **G = (V ,E)**. Because G is connected, there is at least one path between u and v. Let $x_0, x_1, \ldots, x_n$, where $x_0 = u$ and $x_n = v$, be the vertex sequence of a path of least length. This path of least length is simple.

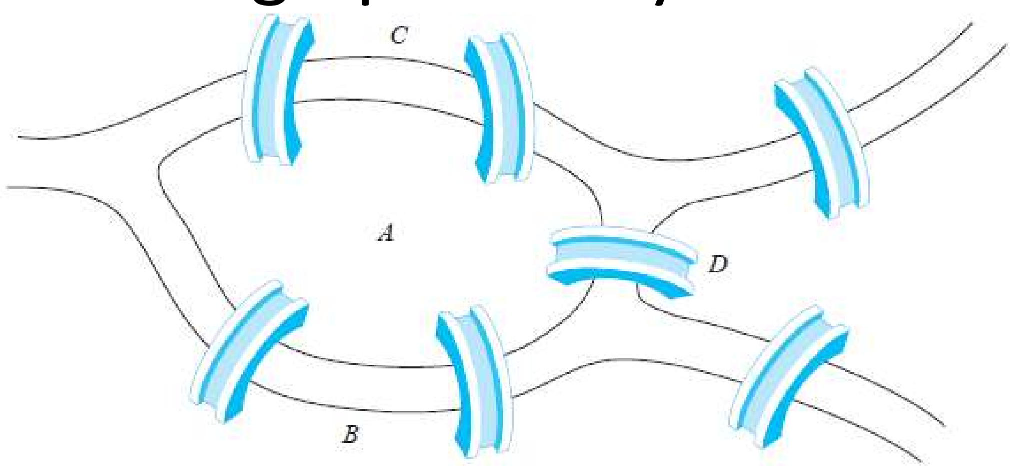
To see this, suppose it is not simple. Then $x_i = x_j$ for some i and j with $0 \leq i < j$. This means that there is a path from u to v of shorter length with vertex sequence $x_0, x_1, \ldots, x_{i-1}, x_j, \ldots, x_n$ obtained by deleting the edges corresponding to the vertex sequence $x_i, \ldots, x_{j-1}$.

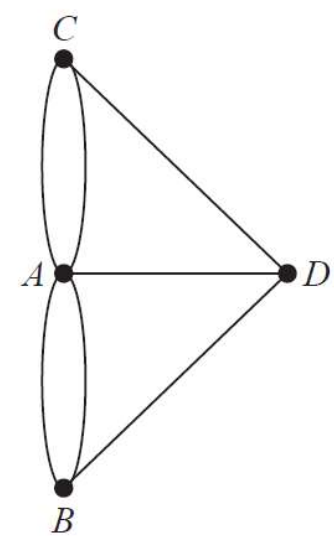
Euler and Hamilton Paths

- Can we travel along the edges of a graph starting at a vertex and returning to it by **traversing each edge** of the graph exactly once? – ***Euler circuit***
- Similarly, can we travel along the edges of a graph starting at a vertex and returning to it while **visiting each vertex** of the graph exactly once? - ***Hamilton circuit***

The Seven Bridges of Königsberg.

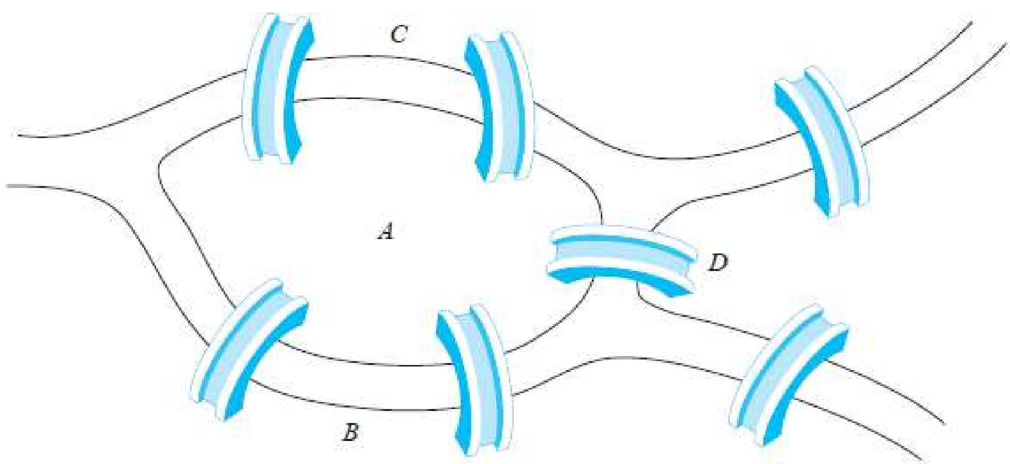
- whether it was possible to start at some location in the town, travel across all the bridges once without crossing any bridge twice, and return to the starting point.
- The Swiss mathematician Leonhard Euler solved this problem. His solution, published in 1736, may be the first use of graph theory.

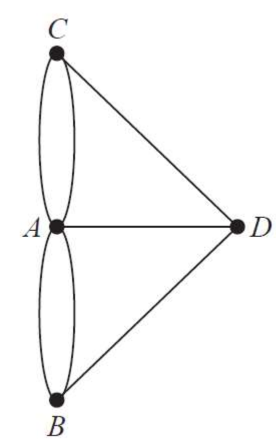




The Seven Bridges of Königsberg.

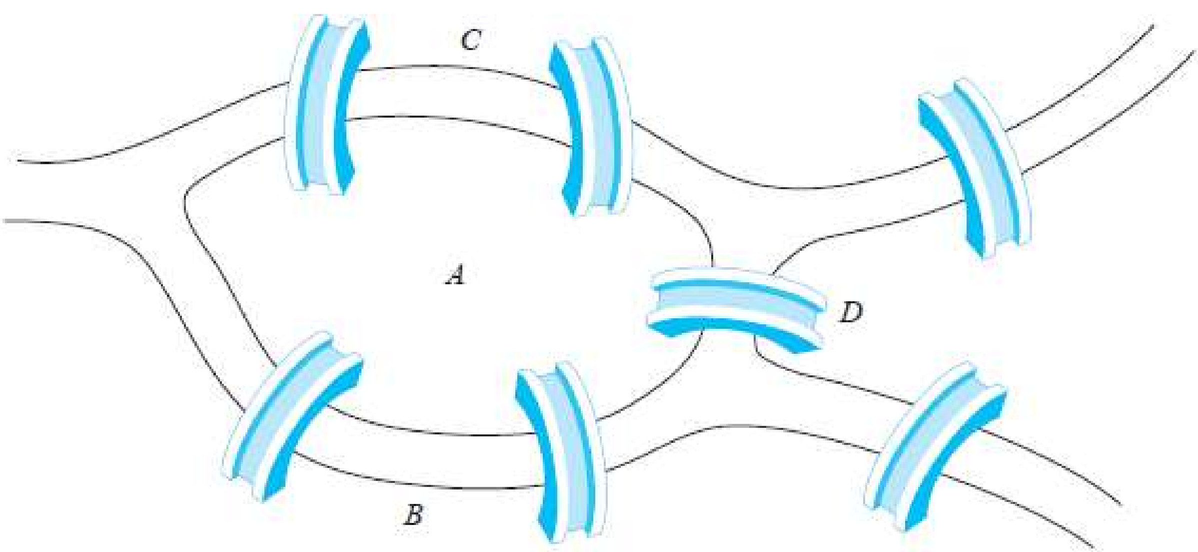
- Euler's argument shows that a necessary condition for the walk of the desired form is that the graph be connected and have exactly zero or two nodes of odd degree. Such a walk is now called an ***Eulerian path*** or ***Euler walk*** in his honor.
- if there are nodes of odd degree, then any Eulerian path will start at one of them and end at the other.

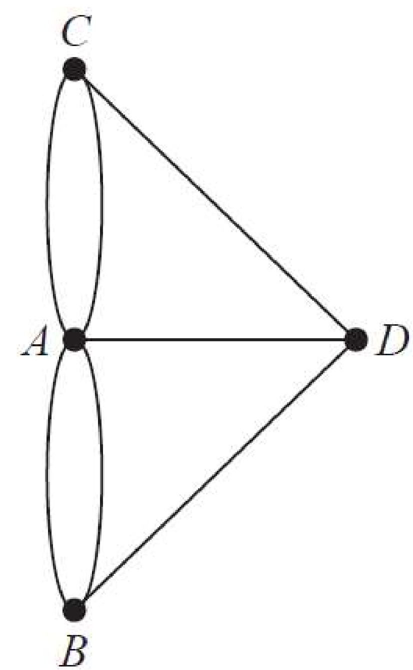




The Seven Bridges of Königsberg.

- ***Eulerian circuit*** or an ***Euler tour***. Such a circuit exists if, and only if, the graph is **connected**, and **there are no nodes of odd degree at all**.
- All Eulerian circuits are also Eulerian paths, but **not all Eulerian paths are Eulerian circuits**.

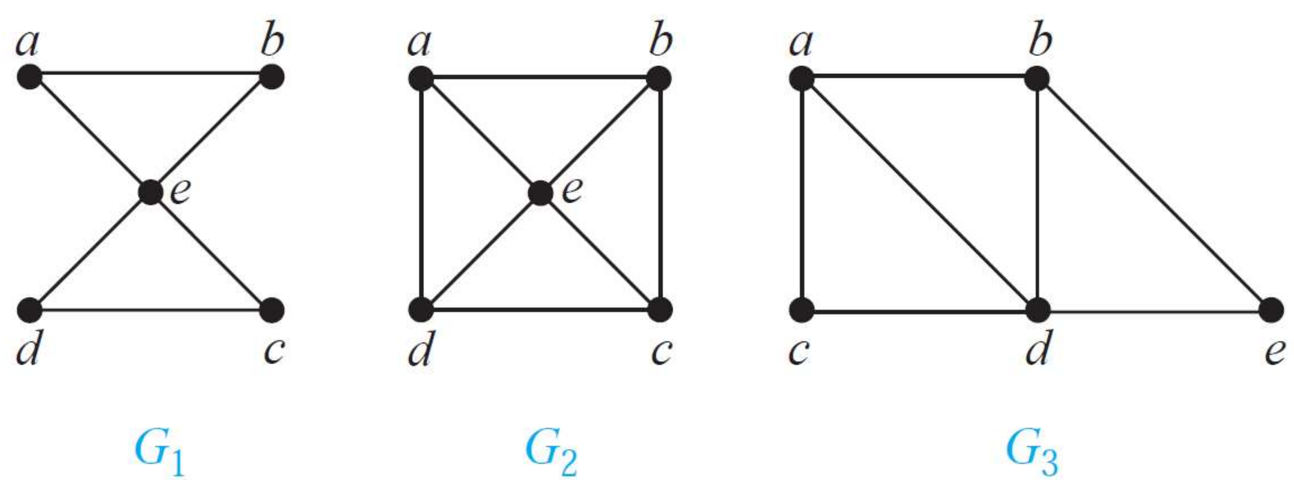




Euler circuit and Euler path

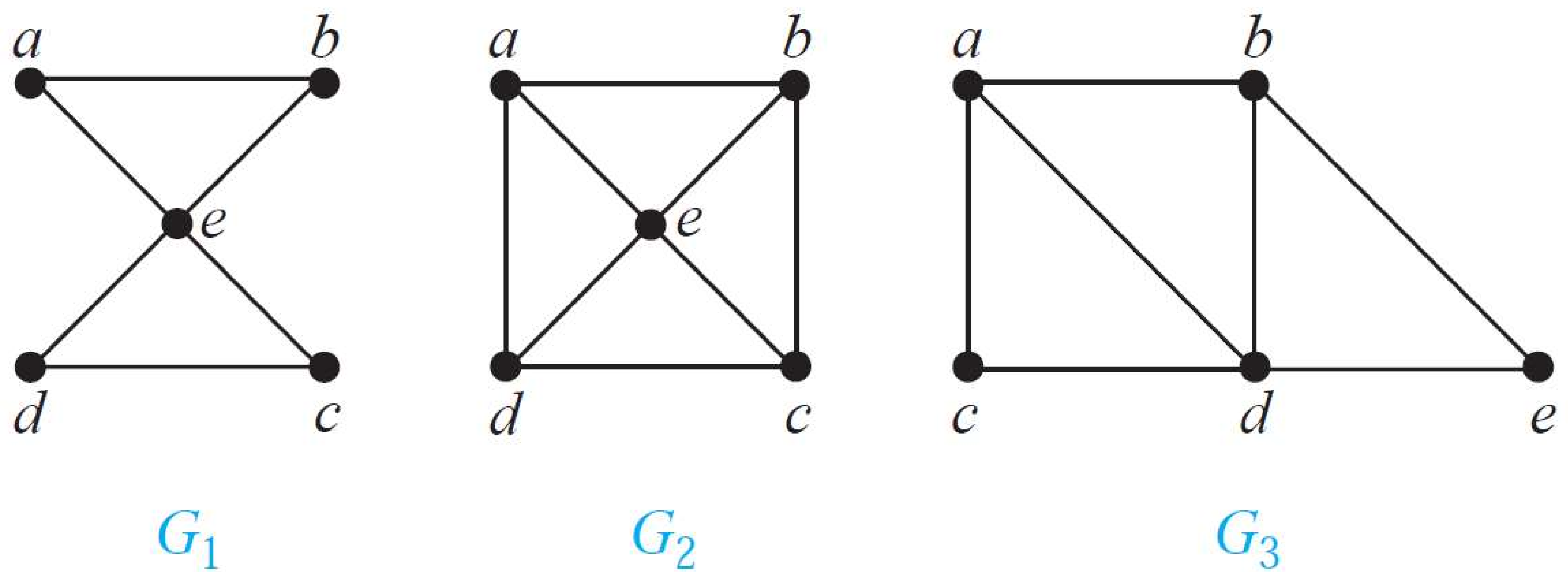
- An **Euler circuit** in a graph G is a **circuit containing every edge** of G and **traversing each edge** of the graph exactly once.
- An **Euler path** in G is a **path** containing **every edge** of G and **traversing each edge** of the graph exactly once.

Which of the undirected graphs in Figure have an Euler circuit?
Of those that do not, which have an Euler path?



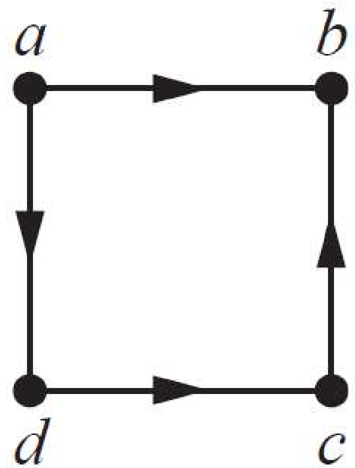
Euler circuit and Euler path

- The graph G1 has an Euler circuit, for example, **a, e, c, d, e, b, a**.
- G3 has an Euler path, namely, **a, c, d, e, b, d, a, b**.

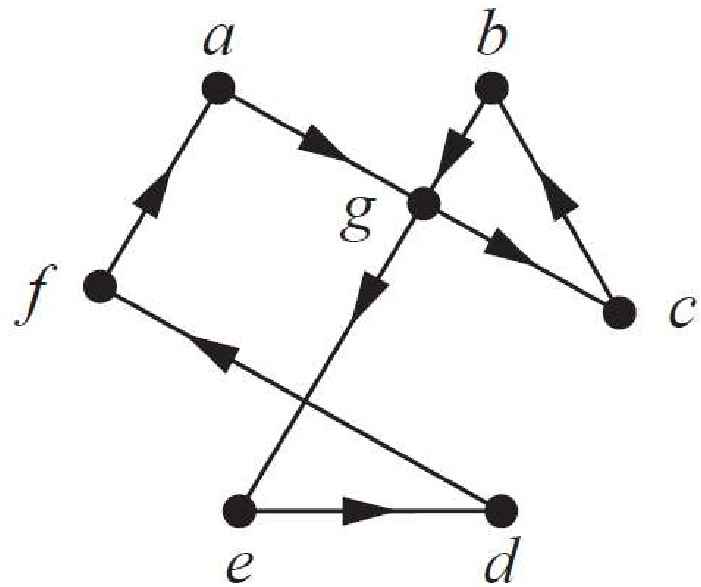


Euler circuit and Euler path

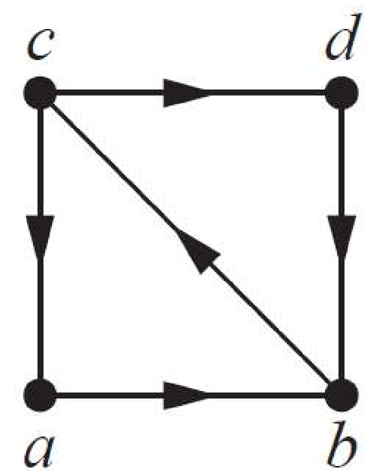
- Which of the directed graphs in Figure have an Euler circuit?
- Of those that do not, which have an Euler path?



H_1



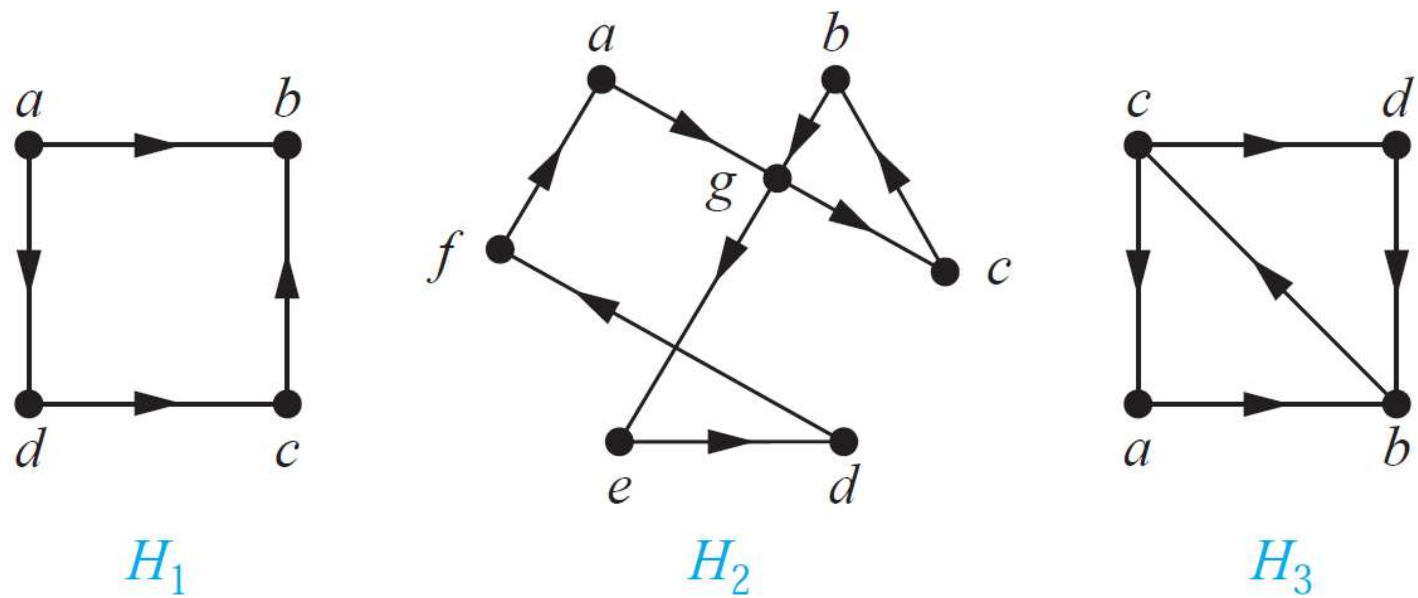
H_2



H_3

Euler circuit and Euler path

- The graph **H2 has an Euler circuit**, for example, **a, g, c, b, g, e, d, f, a**.
- Neither H1 nor H3 has an Euler circuit.
- H3 has an Euler path, namely, **c, a, b, c, d, b**, but H1 does not.

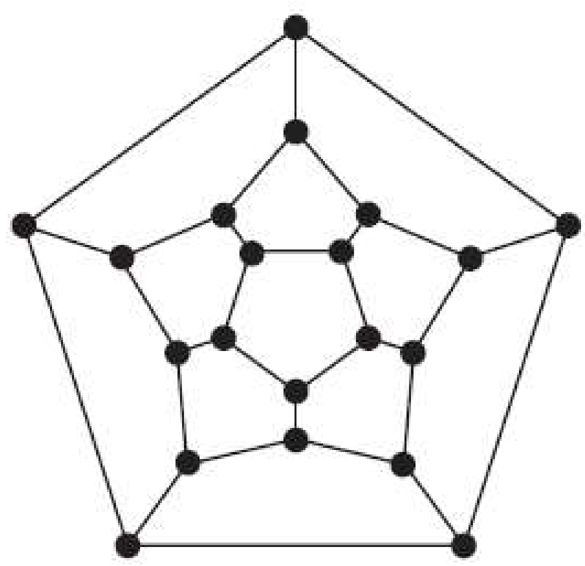


Hamilton Paths and Circuits

- 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, the simple path $x0, x1, \ldots, xn-1, xn$ in the graph $G = (V, E)$ is a Hamilton path if $V = \{x0, x1, \ldots, xn-1, xn\}$ and **$xi = xj$** for $0 \leq i < j \leq n$, and the simple circuit $x0, x1, \ldots, xn-1, xn, x0$ (with $n > 0$) is a Hamilton circuit if $x0, x1, \ldots, xn-1, xn$ is a Hamilton path.

Hamilton Paths and Circuits

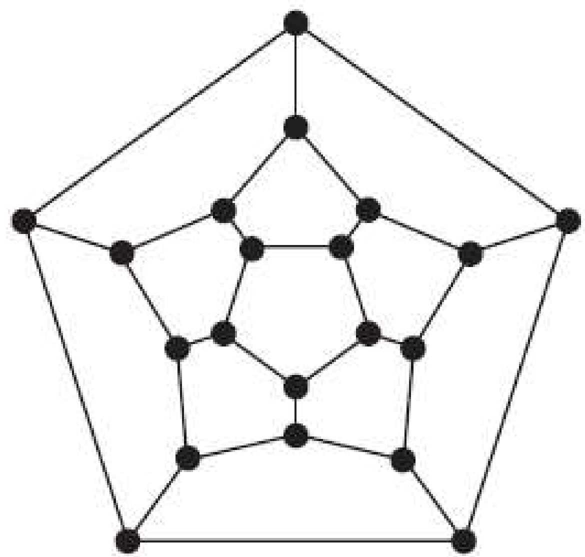
- A simple path in a graph G that passes through **every vertex** exactly once is called a **Hamilton path**, and a simple **circuit** in a graph G that passes through **every vertex exactly once** is called a **Hamilton circuit**.

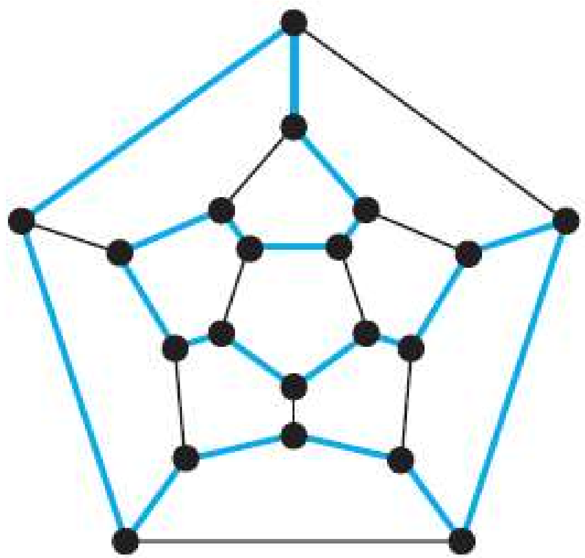


Is there any Hamilton circuit ?

Hamilton Paths and Circuits

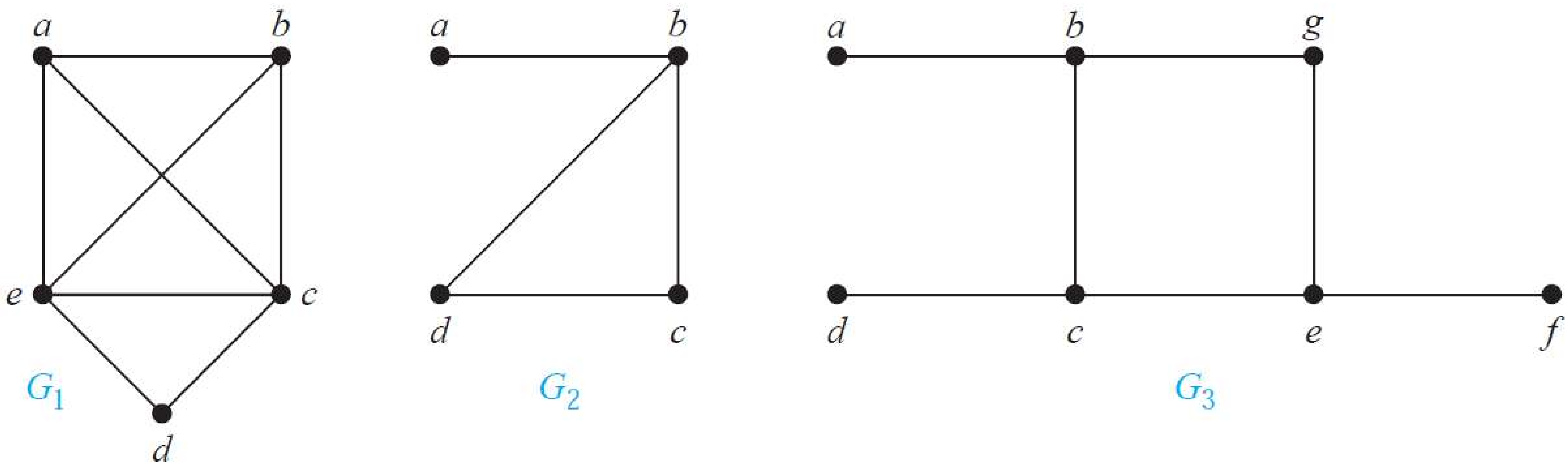
- A simple path in a graph G that passes through **every vertex** exactly once is called a **Hamilton path**, and a simple **circuit** in a graph G that passes through **every vertex exactly once** is called a **Hamilton circuit**.





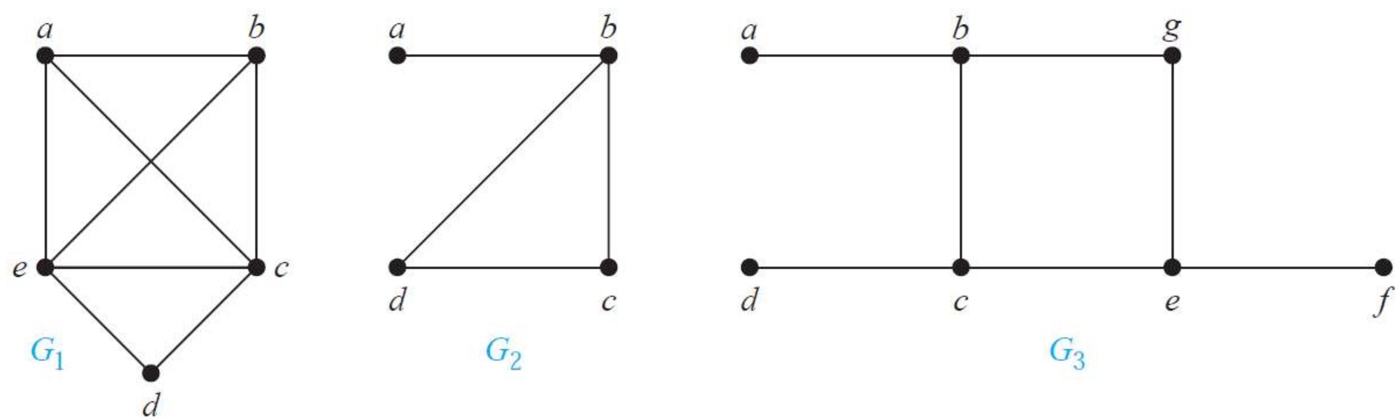
Hamilton Paths and Circuits

- Which of the simple graphs in Figure have a Hamilton circuit or, if not, a Hamilton path?



Hamilton Paths and Circuits

- G1 has a **Hamilton circuit: a, b, c, d, e, a**.
- There is no Hamilton circuit in G2, but G2 does have a **Hamilton path**, namely, **a, b, c, d**.
- **G3 has neither a Hamilton circuit nor a Hamilton path**, because any path containing all vertices must contain one of the edges {a, b}, {e, f }, and {c, d} more than once.



sufficient conditions for Hamilton Circuits

- **DIRAC'S THEOREM**
  - If G is a simple graph with n vertices with $n \geq 3$ such that the **degree of every vertex in G is at least n/2**, then G has a Hamilton circuit.

- **ORE'S THEOREM**
  - **If G is a simple graph with n vertices with $n \geq 3$ such that** **$deg(u) + deg(v) \geq n$** for every pair of nonadjacent vertices u and v in G, then G has a Hamilton circuit.

Applications of Hamilton Circuits

- **The Traveling Salesperson Problem**
  - A traveling salesperson wants to visit each of n cities exactly once and return to his starting point in a minimum cost.

Reference

- **Rosen, Kenneth H., and Kamala Krithivasan. Discrete mathematics and its applications: with combinatorics and graph theory. Tata McGraw-Hill Education, 2016.**