CHAPTER 9
GRAPHS

Outline

- **Content**
  - Graphs and graph models
  - Graph terminology and special types of graphs
  - Representing graphs and graph isomorphism
  - Connectivity
  - Shortest-path problems
  - Euler and Hamilton paths
  - Planar graphs
  - Graph coloring
- **Reading**
  - Chapter 9

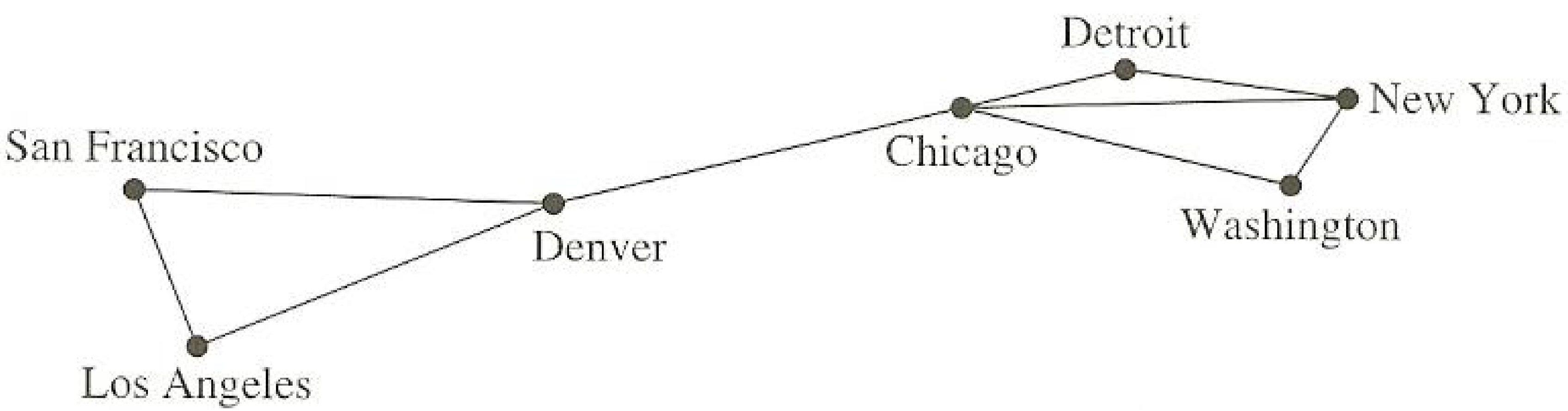
Most of the following slides are by courtesy of Prof. J.-D. Huang and Prof. H.-C. Yen

Graphs

- **Define a graph *G* = (*V*, *E*) where *V* is a nonempty set of vertices (or nodes) and *E* a set of edges. Each edge has one or two vertices associated with it, called its endpoints. An edge connects its endpoints.**
  - If the set of vertices *V* is infinite, we say *G* is an infinite graph; otherwise, *G* is a finite graph.
  - An edge can represent some binary relation!
  - Edges can be directed or undirected
    - Directed: asymmetric
    - Undirected: symmetric

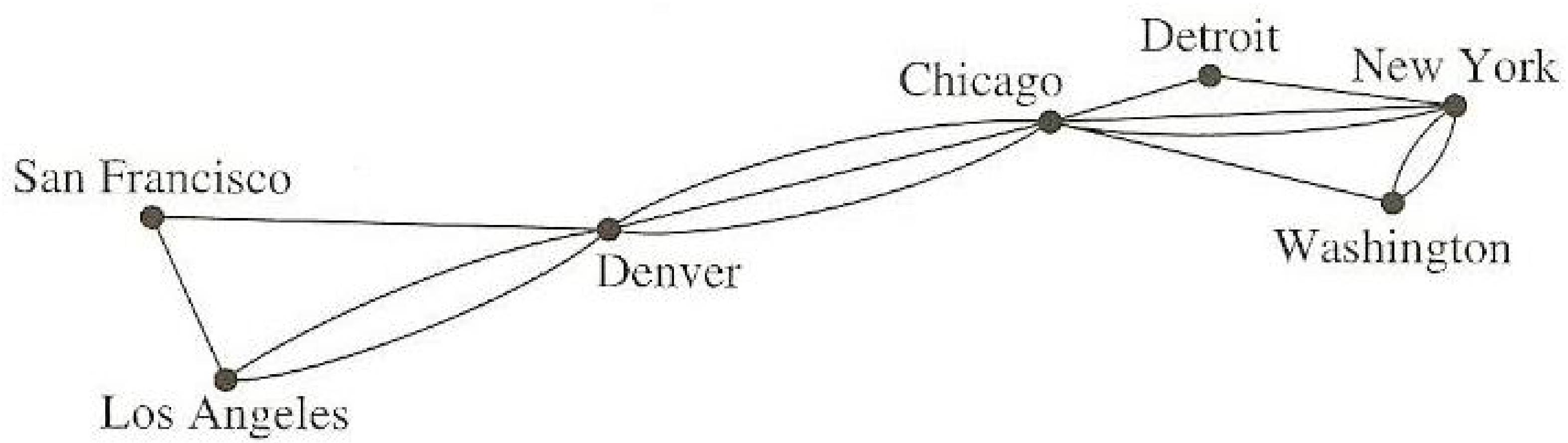
Simple Graphs

- **A simple graph is a graph where each edge connects two different vertices and no two edges connect the same pair of vertices.**
  - No loop
  - Any pair of vertices have no more than one edge



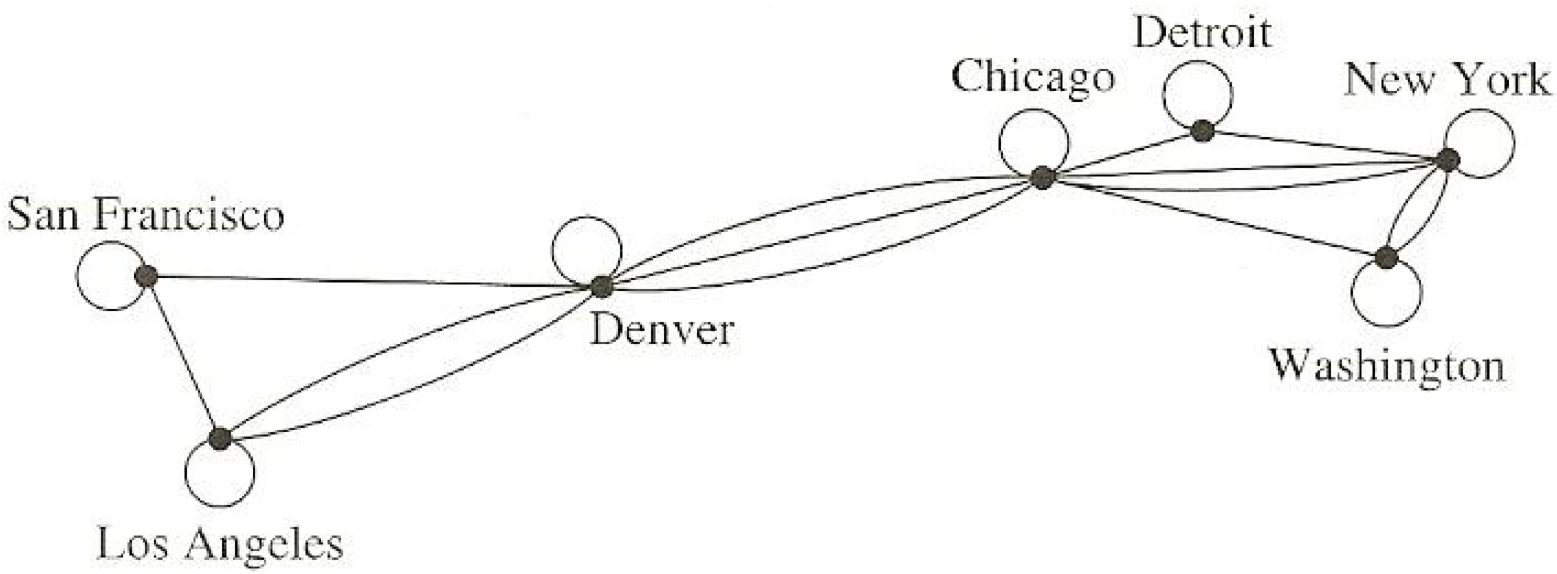
Multigraphs

- Edges connect the same vertices are called **multiple** (or **parallel**) **edges**.
- A **multigraph** is a graph which may have multiple edges. When there are *m* different edges connecting the same pair of vertices, we also say there is an edge of multiplicity *m*.



Pseudographs

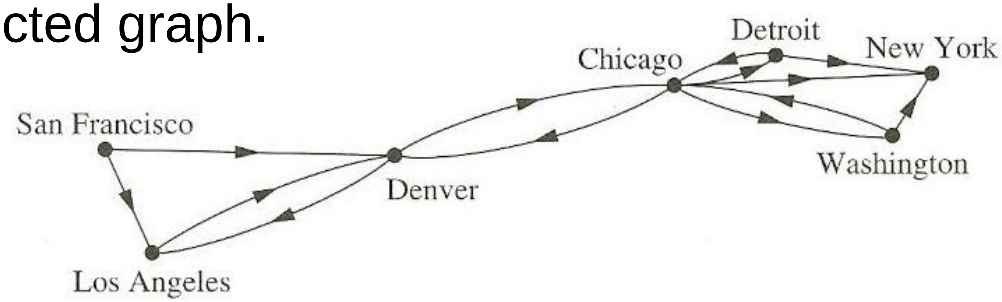
- A **loop** is an edge connecting a vertex.
- A **pseudograph** is a graph which may have **loops** and **multiple** edges.



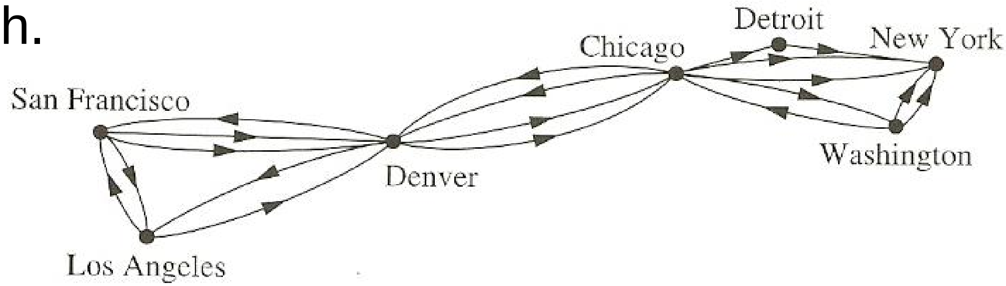
Directed Graphs

- **A directed graph (or digraph) $G = (V, E)$ consists of a nonempty set of vertices V and a set of directed edges (or arcs) E. Each directed edge starts at a vertex and ends at a vertex.**

  - A directed graph with no loops nor multiple directed edges is called a simple directed graph.



  - A directed graph that may have multiple directed edges is called a directed multigraph.



Graph Terminologies

- **Comparison**
  - Keep in mind this terminology is not fully standardized

Type	Edges	Multiple Edges?	Loops?
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Directed graph	Directed	No	Yes
Directed multigraph	Directed	Yes	Yes

- **Graphs are used in many models. Once we have modeled our problems as graphs, many graph algorithms can help us solve these problems.**

Adjacency and Degree (Undirected Graphs)

- **Let $G = (V, E)$ be an undirected graph and $u, v \in V$. We say u and v are adjacent (or neighbors) in G if $\{u, v\} \in E$.**
- **If $e = \{u, v\}$, we say e is incident with the vertices u and v. The edge e is also said to connect u and v.**
- **The vertices u and v are the endpoints of the edge $\{u, v\}$.**

- **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.**
- **The degree of the vertex v is denoted by deg(v).**
- **A vertex of degree zero is called isolated. A vertex of degree one is called pendant.**

Handshaking Theorem (1/2)

- **(The Handshaking Theorem) Let *G* = (*V*, *E*) be an undirected graph. Then**

$$2|E| = \sum_{v \in V} \deg(v).$$

  - Proof by induction on the number of edges |*E*|.

- **Degree Property: An undirected graph has an even number of vertices of odd degree**
  - Why?

$$2|E| = \sum_{v \in V} \deg(v) = \sum_{v \in V_{even}} \deg(v) + \sum_{v \in V_{odd}} \deg(v)$$

Handshaking Theorem (2/2)

- **Pf:**

  We prove by induction on the number of edges $|E|$.
  BASIS STEP. $|E| = 0$. Obvious, since $\deg(v) = 0$ for all $v \in V$.
  INDUCTIVE STEP. Assume the statement holds for $|E| = k$. Consider any graph G with $|E| = k + 1$. Remove an edge e from G and construct $G' = (V, E - \{e\})$. Then $|E - \{e\}| = k$. We have $2|E| - 2 = \sum_{v \in V} \deg_{G'}(v)$ by inductive hypothesis. We have the following two cases:

  - e is not a loop. Consider the endpoints u, v of e. $\deg_G(u) = \deg_{G'}(u) + 1$, $\deg_G(v) = \deg_{G'}(v) + 1$, but $\deg_G(x) = \deg_{G'}(x)$ for other vertices x. Hence $\sum_{v \in V} \deg_G(v) = 2 + \sum_{v \in V} \deg_{G'}(v) = 2|E|$.
  - e is a loop. Consider the endpoint u of e. We have $\deg_G(u) = \deg_{G'}(u) + 2$. We have $\sum_{v \in V} \deg_G(v) = \sum_{v \in V} \deg_{G'}(v) + 2 = 2|E|$.

Adjacency and Degree (Directed Graphs)

- Let $G = (V, E)$ be a directed graph and $(u, v) \in E$. We say that u is adjacent to v and v is adjacent from u. The vertex u is the initial vertex of (u, v), and v is the terminal or end vertex of (u, v).

- Let $G = (V, E)$ be a directed graph and $v \in V$.

  The in-degree of v, $\deg^-(v)$, is the number of edges with v as terminal vertex.

  The out-degree of v, $\deg^+(v)$, is the number of edges with v as initial vertex.

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

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

Proof of Degree Property

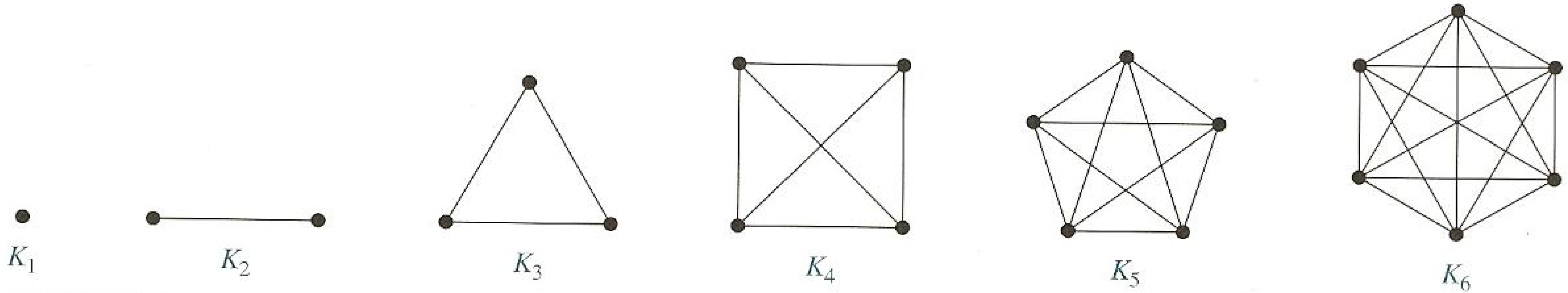
- **Pf:**

We prove by induction on $|E|$.
BASIS STEP: $|E| = 0$. Observe that $\deg^-(v) = \deg^+(v) = 0$ for all $v \in V$.
INDUCTIVE STEP: Assume the statement holds for $|E| = k$. Let $e \in E$, consider $G' = (V, E - \{e\})$. By inductive hypothesis, we have $\sum_{v \in V} \deg^-_{G'}(v) = \sum_{v \in V} \deg^+_{G'}(v) = |E| - 1$. There are two cases:

- $e = (u, v)$ with $u \neq v$. Then $\deg^-_G(v) = \deg^-_{G'}(v) + 1$ and $\deg^-_G(x) = \deg^-_{G'}(x)$ for all $x \neq v$. Similarly, we have $\deg^+_G(u) = \deg^+_{G'}(u) + 1$ and $\deg^+_G(x) = \deg^+_{G'}(x)$ for all $x \neq u$. The result follows.
- $e = (u, u)$. Then $\deg^-_G(u) = \deg^-_{G'}(u) + 1$, $\deg^+_G(u) = \deg^+_{G'}(u) + 1$ and all other terms remain the same. The result follows as well.

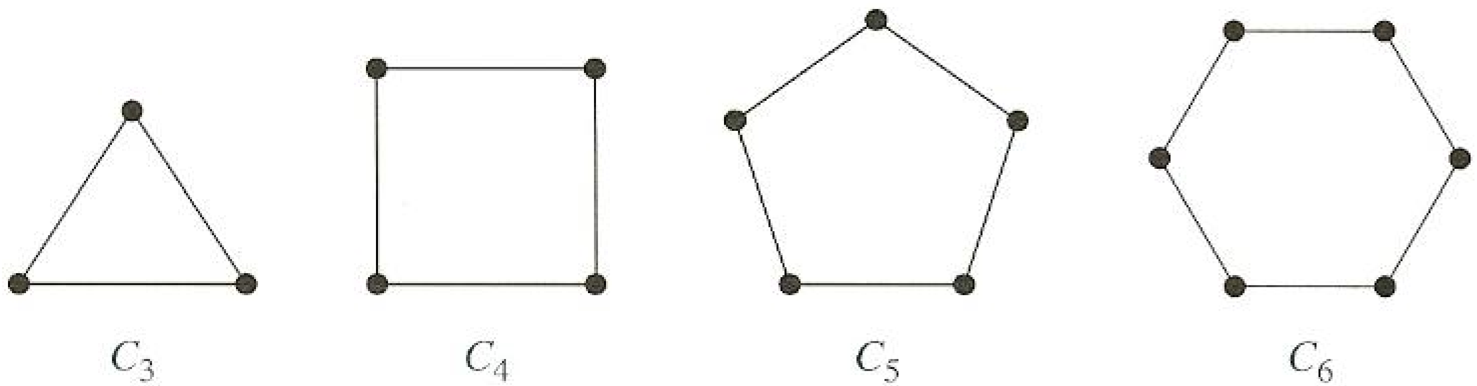
Complete Graphs

- Let $K_n = (V, E)$ be a graph with
  $|V| = n$ and
  $E = \{\{u, v\} : u, v \in V, u \neq v\}$.
  We say K_n is the **complete graph** on n vertices.



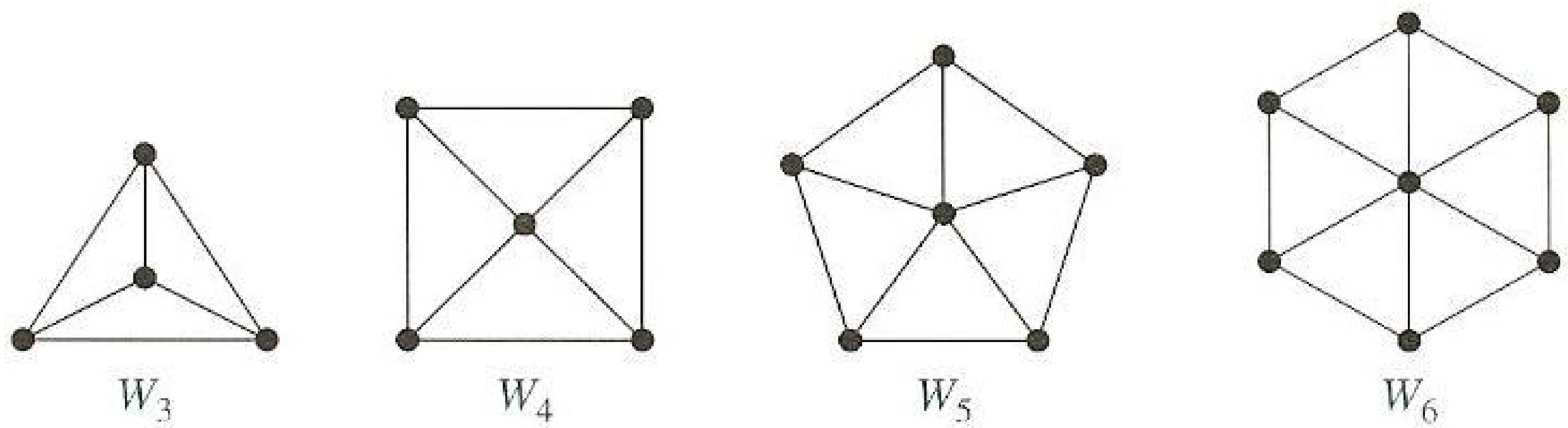
Cycles

- **Let C_n = (V, E) be a graph with $n \geq 2$, $V = \{v_0, v_1, \ldots, v_{n-1}\}$ and $E = \{\{v_i, v_{i+1 \bmod n}\} : 0 \leq i < n\}$. We say C_n is the cycle of size n.**
  - n edges: $\{v_0, v_1\}$, $\{v_1, v_2\}$, …, $\{v_{n-2}, v_{n-1}\}$ and $\{v_{n-1}, v_0\}$



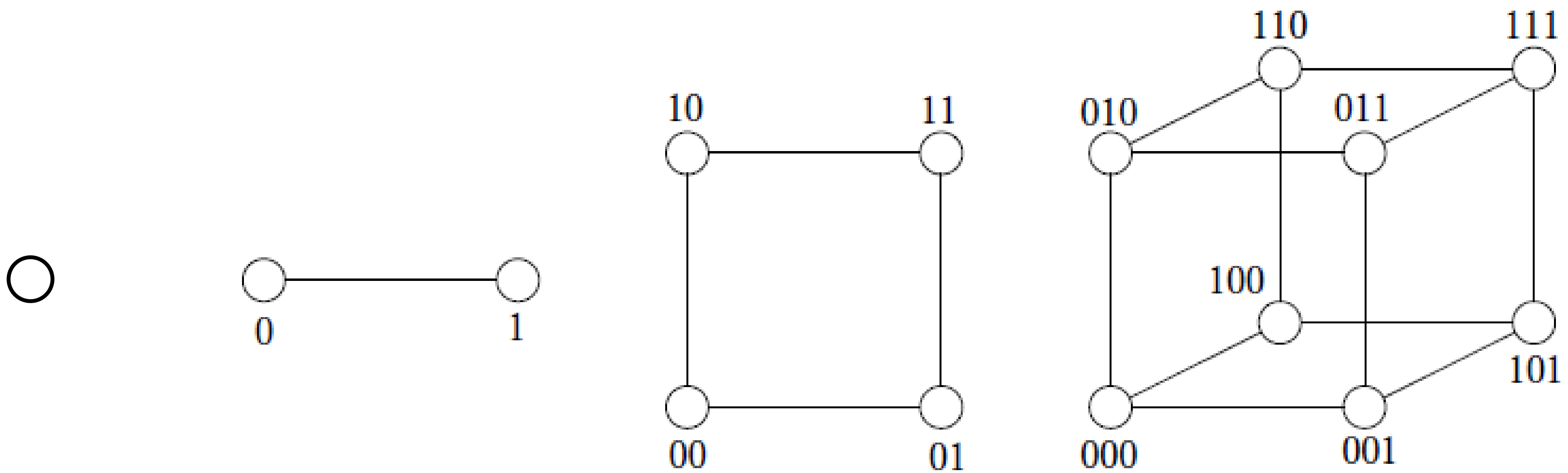
Wheels

- **The wheel W_n is obtained by adding an extra vertex to the cycle C_n, $n \geq 3$, and connect this new vertex to each of the n vertices in C_n by new edges**



Cubes (Hypercubes)

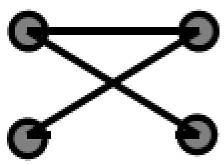
- Let $Q_n = (V, E)$ be a graph with
  $V = \{v_0, v_1, \ldots, v_{2^n-1}\}$ and
  $E = \{\{v_i, v_j\} : 0 \leq i, j < 2^n, (i)_2$ differs from $(j)_2$ by one position $\}$.
  We say Q_n is the ***n*-cube**.

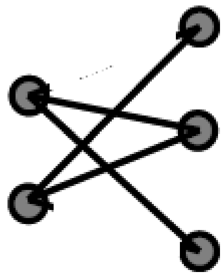


Q_0 Q_1 Q_2 Q_3

Bipartite Graphs

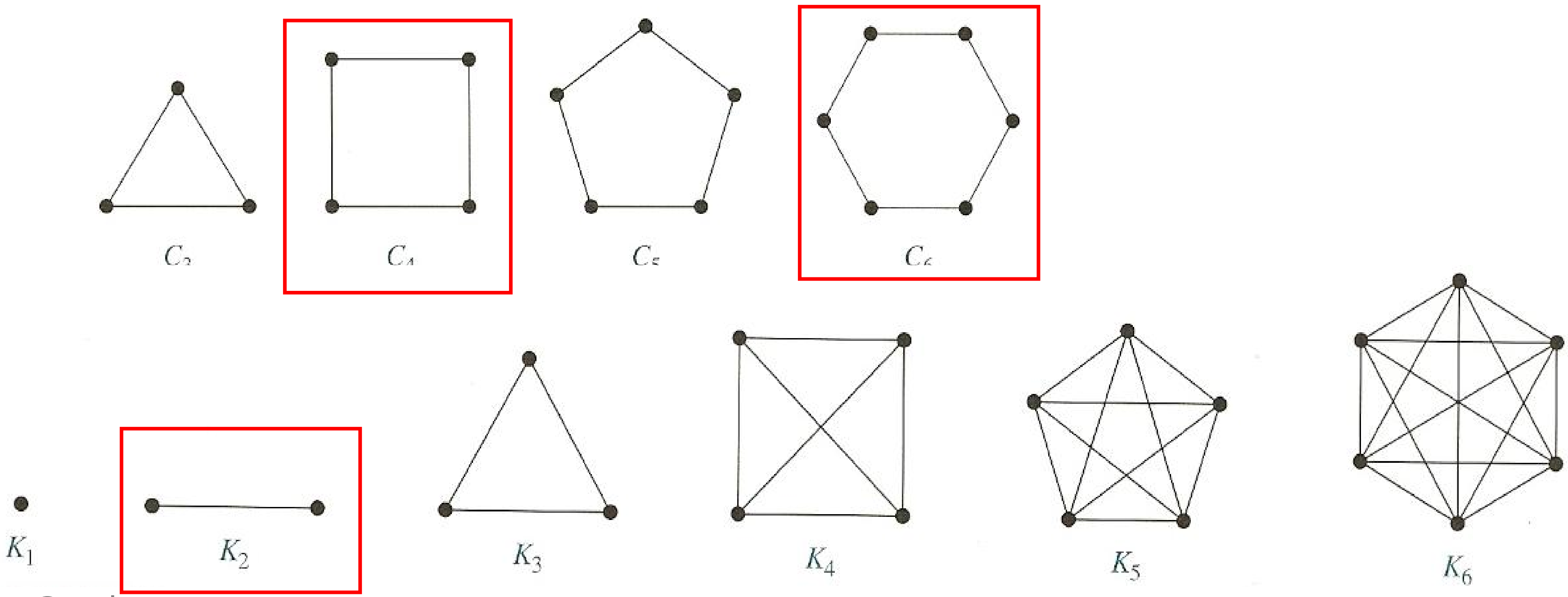
- **Let $G = (V, E)$ be a graph with $V = V_0 \cup V_1$ and $V_0 \cap V_1 = \varnothing$ such that $E \subseteq \{\{v_0, v_1\} : v_0 \in V_0, v_1 \in V_1\}$. Then we say G is bipartite.**





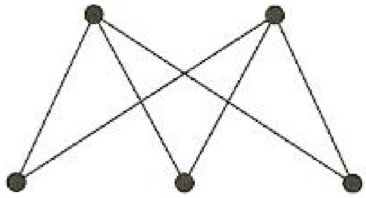
Bipartite Graphs

- Let $G = (V, E)$ be a graph with $V = V_0 \cup V_1$ and $V_0 \cap V_1 = \varnothing$ such that $E \subseteq \{\{v_0, v_1\} : v_0 \in V_0, v_1 \in V_1\}$. Then we say G is **bipartite**.
  - E.g., C_4, C_6 are bipartite but C_3, C_5 are not.

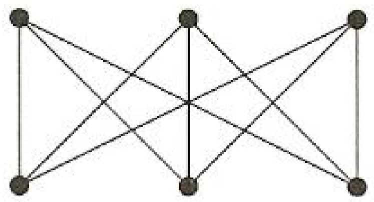


Complete Bipartite Graphs

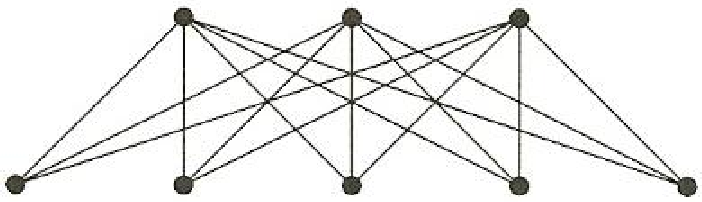
- Let $K_{m,n} = (V, E)$ be a bipartite graph with $V = V_0 \cup V_1$ and $V_0 \cap V_1 = \varnothing$, and $|V_0| = m$, $|V_1| = n$. Furthermore, $E = \{\{v_0, v_1\} : v_0 \in V_0, v_1 \in V_1\}$. We say $K_{m,n}$ is the **complete bipartite graph**.



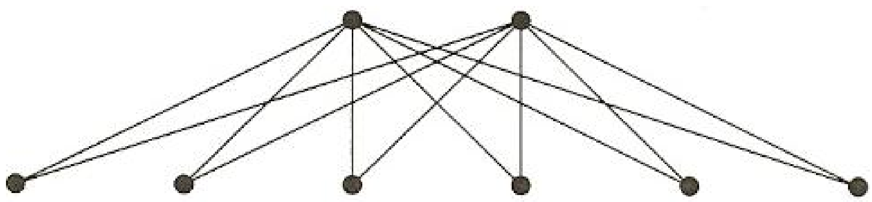
$K_{2,3}$



$K_{3,3}$



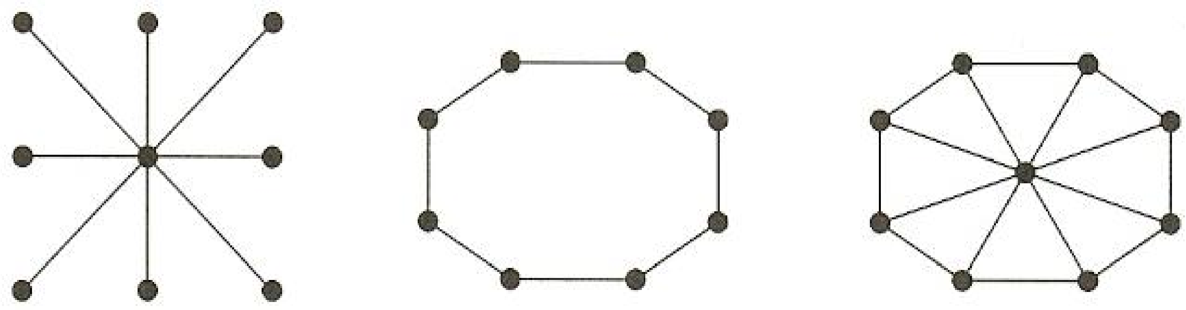
$K_{3,5}$



$K_{2,6}$

Special Types of Graphs

Common LAN Topology



(a) Star

(b) Ring

(c) Hybrid

Interconnection Networks

FIGURE 11 A Linear Array for Six Processors.

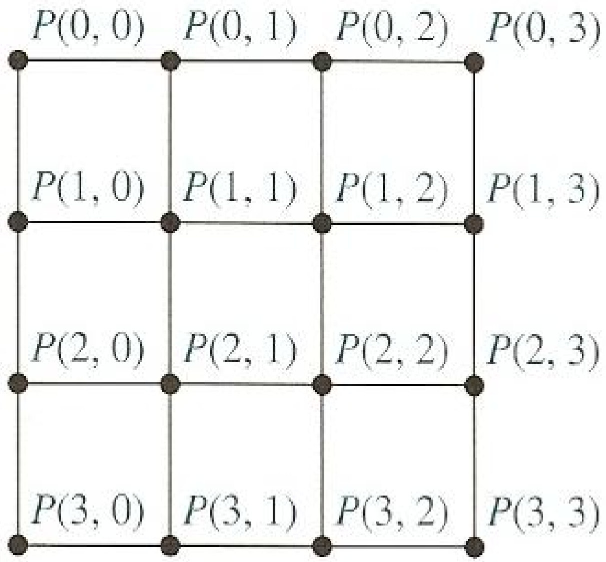


FIGURE 12 A Mesh Network for 16 Processors.

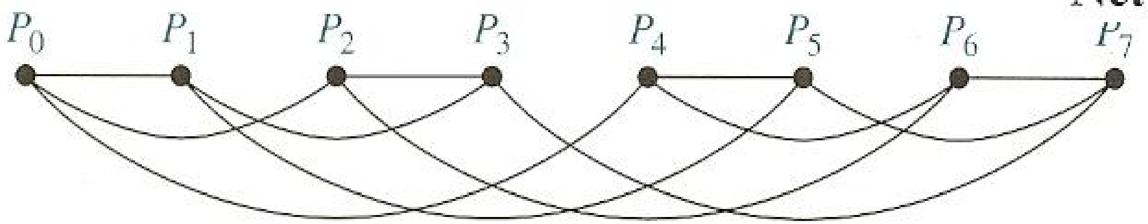


FIGURE 13 A Hypercube Network for Eight Processors.

Subgraphs

- **Let $G = (V, E)$ be a graph. If $H = (W, F)$ with $W \subseteq V$ and $F \subseteq E$ is a graph, then we say H is a subgraph of G.**
  - E.g., K_3 is a subgraph of K_4. C_4 is a subgraph of Q_3.

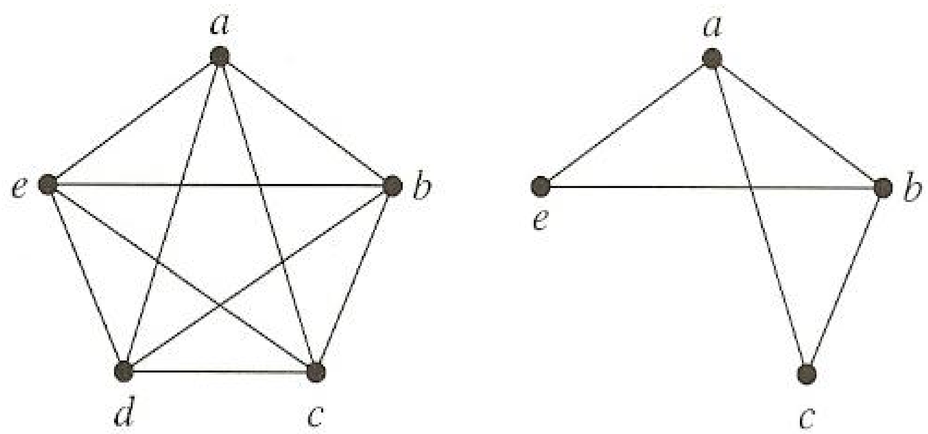


FIGURE 14 A Subgraph of K_5.

Union of Graphs

- Let $G_0 = (V_0, E_0)$ and $G_1 = (V_1, E_1)$ be graphs.
  Define the union $G_0 \cup G_1 = (V, E)$ of G_0 and G_1 with
  $V = V_0 \cup V_1$ and $E = E_0 \cup E_1$.

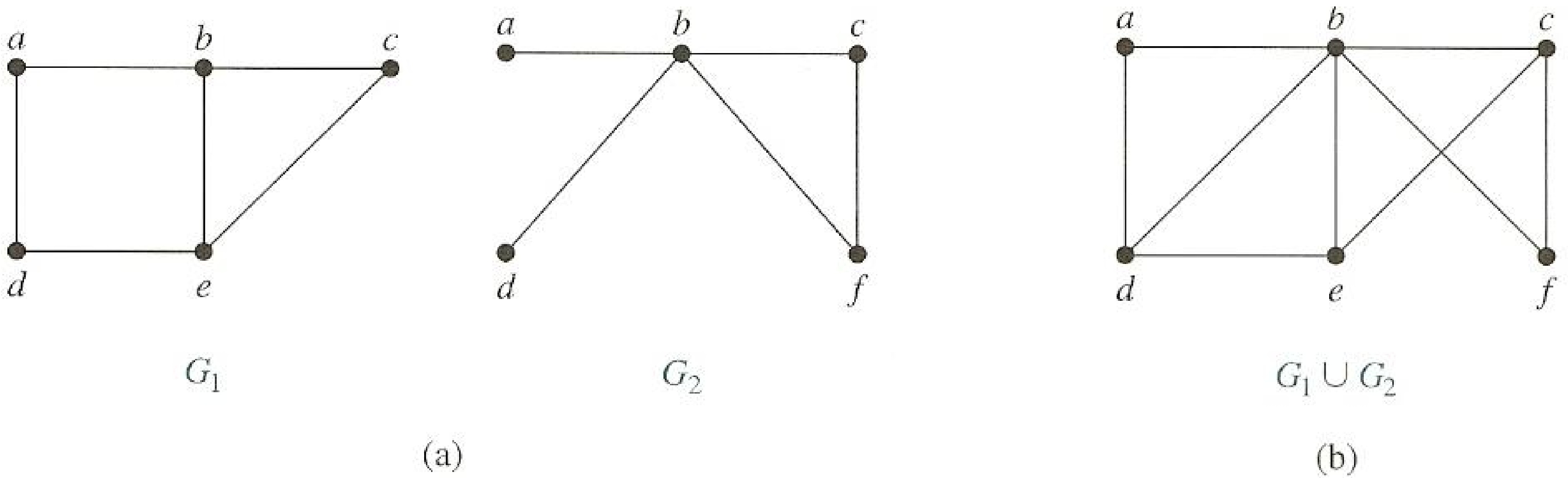


FIGURE 15 (a) The Simple Graphs G_1 and G_2; (b) Their Union $G_1 \cup G_2$.

Adjacency Lists for Simple Graphs

- **Adjacency list: We can represent a graph by a list of adjacent vertices for each vertex.**

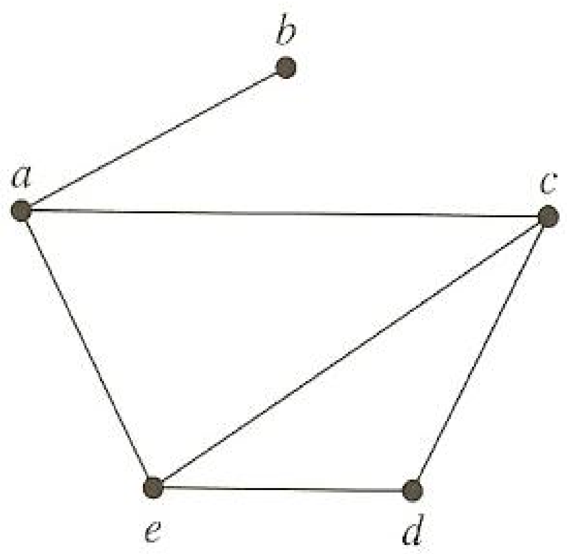


FIGURE 1 A Simple Graph.

TABLE 1 An Edge 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

Adjacency Lists for Directed Graphs

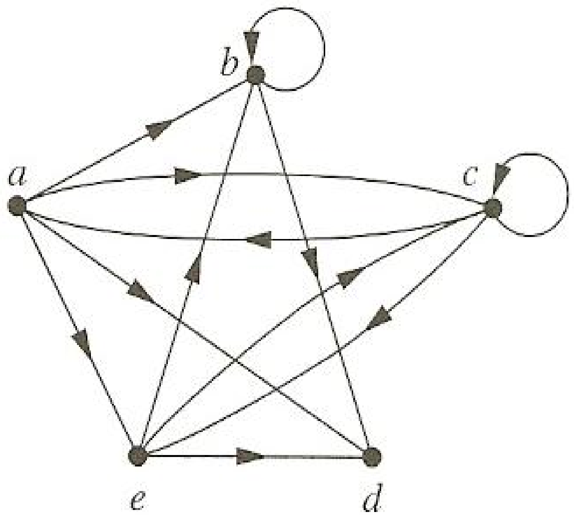


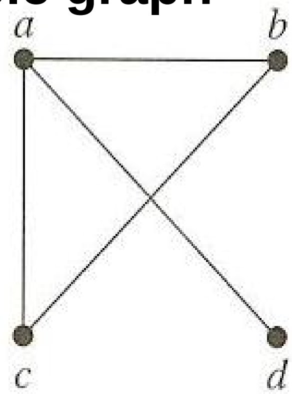
FIGURE 2 **A Directed Graph.**

TABLE 2 **An Edge 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

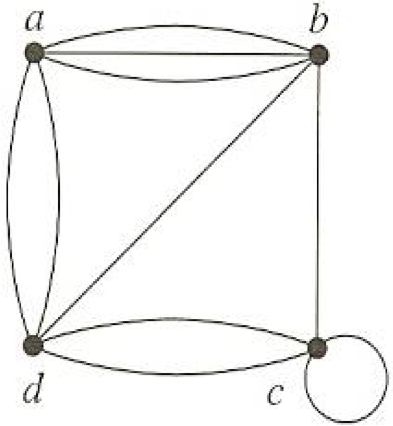
Adjacency Matrices

- **Simple graph**



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

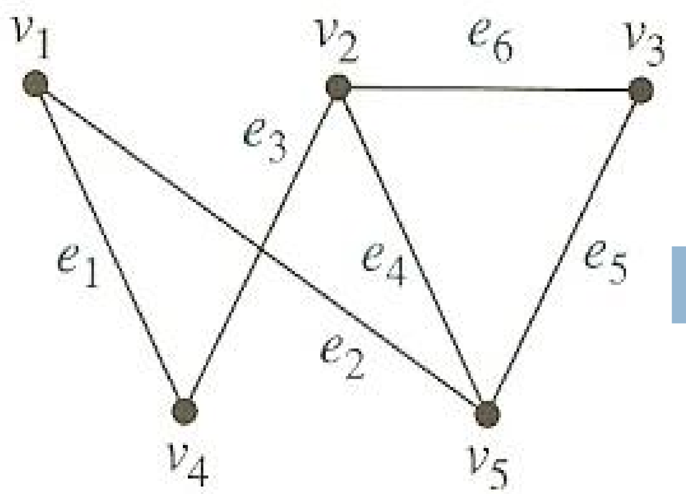
- **Pseudograph**
  - Easily specify multiplicity



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

Still symmetric

Incidence Matrices



	e_1	e_2	e_3	e_4	e_5	e_6
v_1	1	1	0	0	0	0
v_2	0	0	1	1	0	1
v_3	0	0	0	0	1	1
v_4	1	0	1	0	0	0
v_5	0	1	0	1	1	0

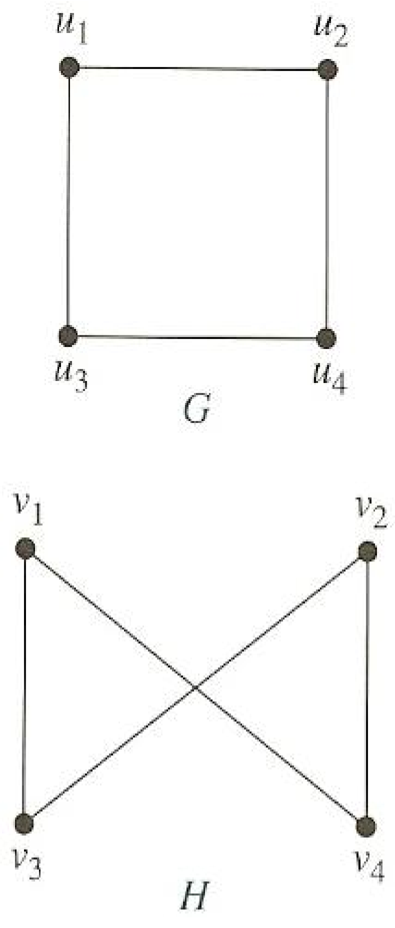
Isomorphism

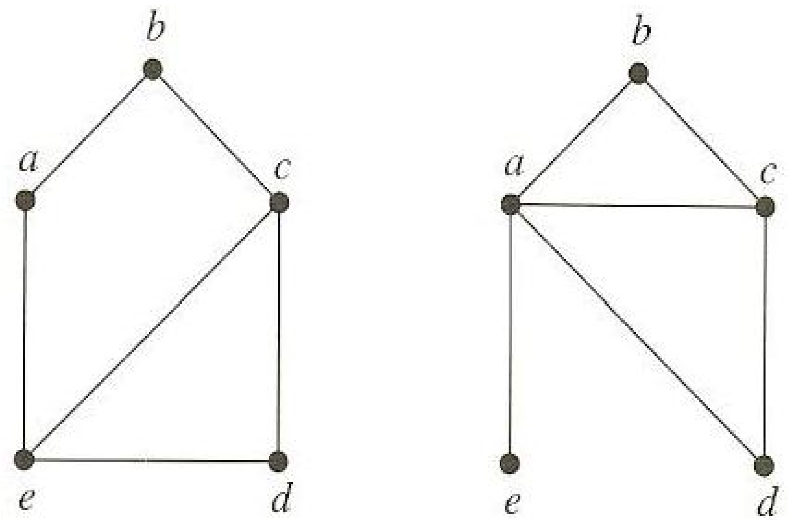
- **Let $G_0 = (V_0, E_0)$ and $G_1 = (V_1, E_1)$ be graphs. We say G_0 and G_1 are isomorphic if there exists a bijection $f : V_0 \rightarrow V_1$ such that $\{u, v\} \in E_0$ iff $\{f(u), f(v)\} \in E_1$**

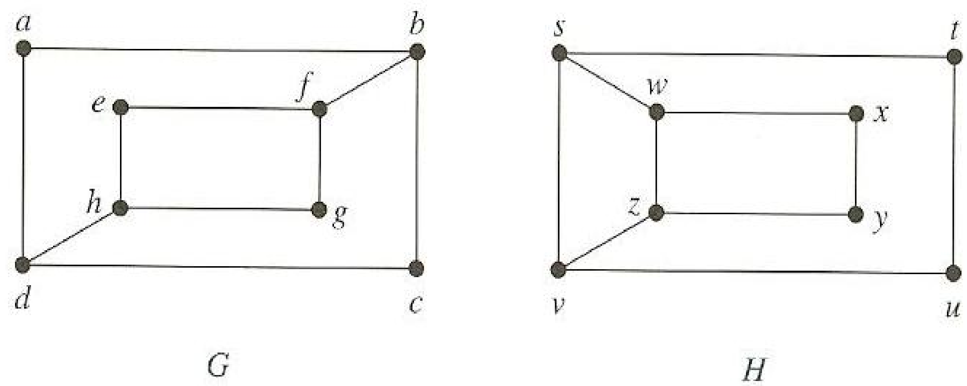
- ***f* is the "renaming" function that makes the two graphs identical**
- **Isomorphism of simple graphs is an equivalence relation**

Examples

- **Isomorphic or not?**







Isomorphism Check

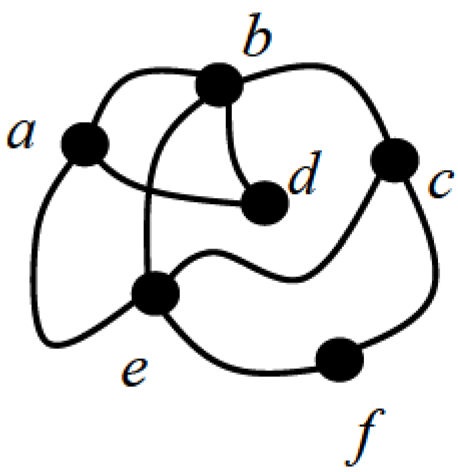
- **Given two graphs G_0 and G_1 (in some representation), determine whether G_0 is isomorphic to G_1 is called the graph isomorphism problem. So far, we do not know if the problem is tractable.**

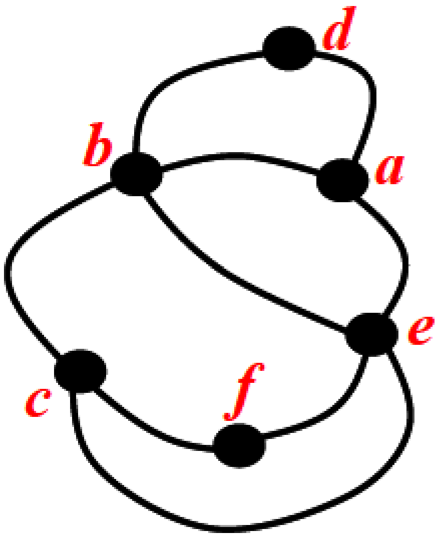
- **It is difficult to check if 2 graphs are isomorphic**
  - n! possible bijective functions

- **Some simple invariants if 2 graphs are isomorphic**
  - same number of vertices
  - same number of edges
  - the degrees of vertices must be the same

Example

- **If isomorphic, label the 2nd graph to show the isomorphism, else identify difference**





Paths

- **Let $G = (V, E)$ be a simple graph. A path of length n from v_0 to v_n is a sequence of n edges**

  $$(\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\})$$

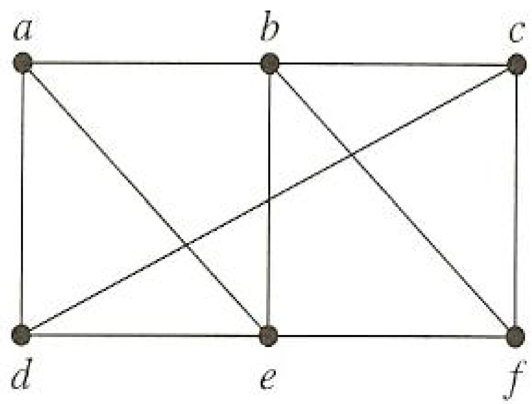
  where $\{v_i, v_{i+1}\} \in E$ for $0 \le i < n$.

  We denote the path by $v_0, v_1, \ldots, v_n$.
  - The path is a circuit (cycle) if $v_0 = v_n$ and $n > 0$.
  - A path is simple if it does not contain the same vertex more than once.
  - An undirected graph is called **connected** if there is a **path** between every pair of distinct vertices.

- **Our definition is a special case of the one in the textbook because we do not consider multigraphs. However, it should be straightforward to extend the definition to multigraphs. We can also generalize the definition to directed graphs.**

Examples

- **a, d, c, f, e is a simple path of length 4**
- **d, e, c, a is NOT a path**
  - {e, c} is not an edge
- **b, c, f, e, b is a circuit of length 4**
- **a, b, e, d, a, b is a path of length 5**
  - However, it is NOT a simple path



Connectivity

- **Theorem: There is a simple path between every pair of distinct vertices of a connected undirected graph.**
- **Pf:**

By the definition of connectivity, we have a path (not necessarily simple) between any two distinct vertices. It remains to show that a simple path does exist.
Consider the following path between v_0 and v_n

$$(\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{i-1}, v_i\}, \ldots, \{v_j, v_{j+1}\}, \ldots, \{v_{n-1}, v_n\})$$

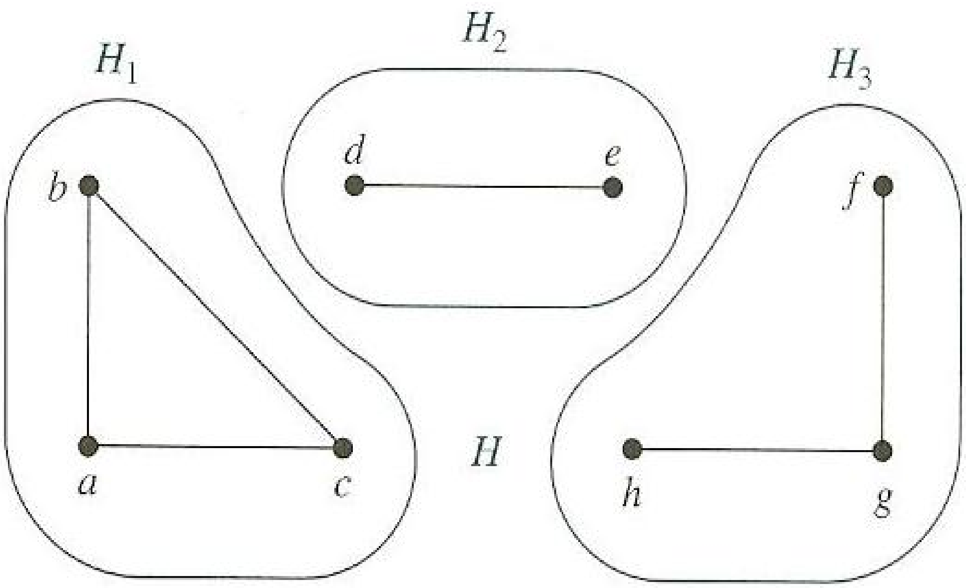
with $v_i = v_j$. We see that the following is another path between v_0 and v_n

$$(\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{i-1}, v_i\}, \{v_j, v_{j+1}\}, \ldots, \{v_{n-1}, v_n\}).$$

Repeat this process, we can obtain a required simple path. □

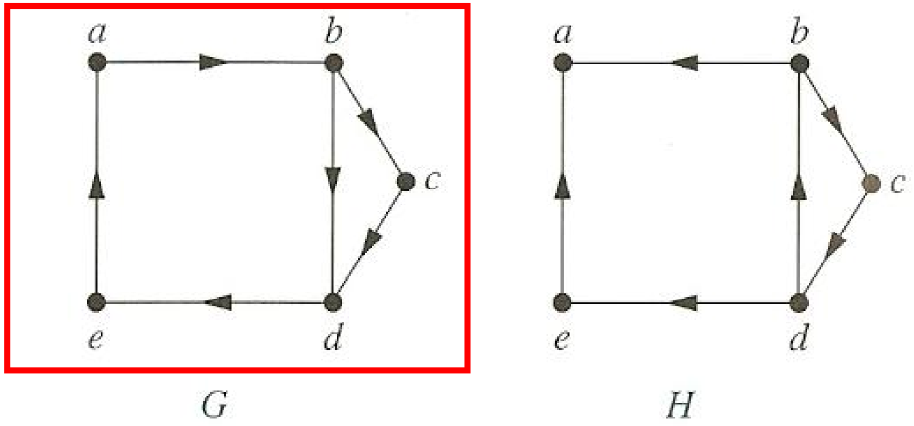
Connected Components

- **If a undirected graph is not connected, it is the union of several disjoint connected subgraphs. We call such connected subgraph the connected components of the graph.**



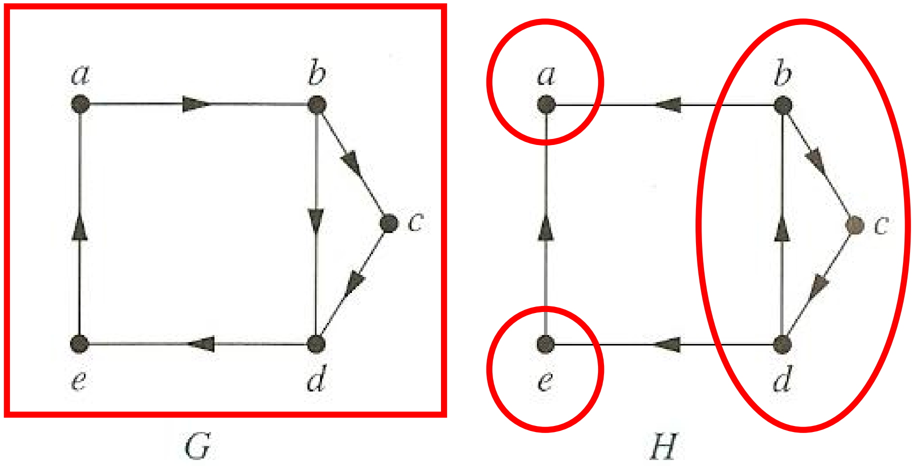
Connectivity in Directed Graphs

- **A directed graph is weakly connected if there is a path between every 2 distinct vertices in the underlying undirected graph**
- **A directed graph is strongly connected if there is a path from *a* to *b* and another path from *b* to *a* whenever *a* and b are 2 distinct vertices**
  - Mutually connected
  - The underlying undirected graph for a directed graph is defined by removing the direction of edges. Let $G = (V, E)$ be a directed graph. Define its underlying undirected graph $G' = (V, E')$ where (v, u) or $(u, v) \in E$ if and only if $\{u, v\} \in E'$



Strongly Connected Components

- The **maximal** strongly connected subgraphs are called the **strongly connected components**



Counting Paths between Vertices

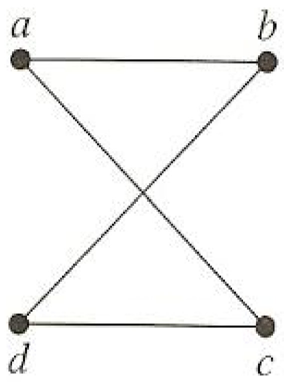
- **Theorem: Let *G* be a graph with adjacency matrix *A*. The number of different paths of length *r* from v_i to v_j equals (*i*, *j*)-th entry of A^r**
- **Pf:**

We prove by induction on r.
BASIS STEP: $r = 1$. Obvious, by the definition of adjacency matrix.
INDUCTIVE STEP: Assume the theorem holds for $r = k$. Let $\mathbf{A} = [a_{ik}]_{n\times n}$ and $\mathbf{A}^k = [b_{kj}]_{n\times n}$. Then $\mathbf{A}^{k+1} = [c_{ij}]_{n\times n}$ where $c_{ij} = \sum_{k=0}^{n-1} a_{ik} b_{kj}$.
By inductive hypothesis, there are b_{kj} different paths from v_k to v_j. If there are a_{ik} edges from v_i to v_k, there are $a_{ik} b_{kj}$ different paths of length $k + 1$ from v_i to v_j via v_k. Counting different paths from v_i to v_j via all v_k, we have c_{ij} different paths from v_i to v_j as required. □

Example

- **E.g.,**

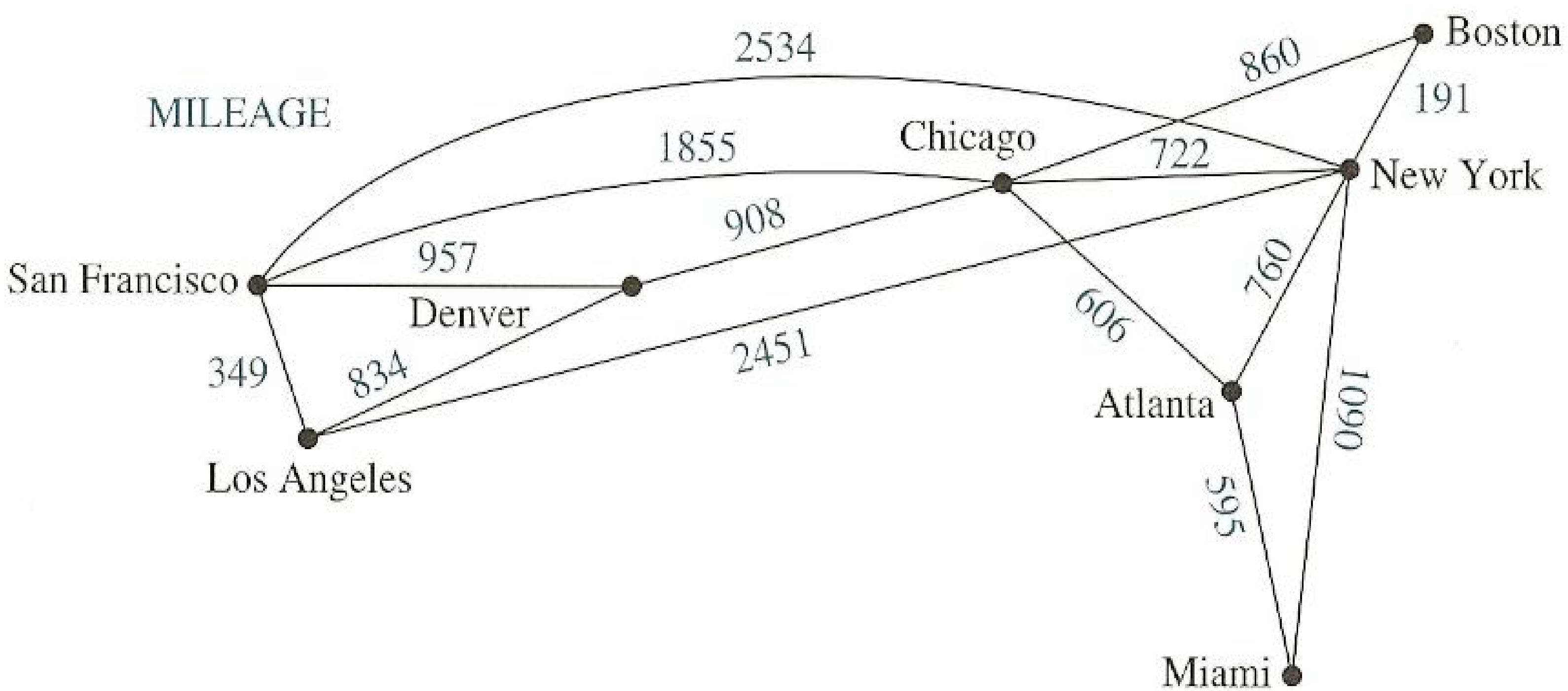
$$A=\begin{bmatrix}0&1&1&0\\1&0&0&1\\1&0&0&1\\0&1&1&0\end{bmatrix} \Rightarrow A^4=\begin{bmatrix}8&0&0&8\\0&8&8&0\\0&8&8&0\\8&0&0&8\end{bmatrix}$$



- **Think about the connectivity**
- **Think about the transitivity**

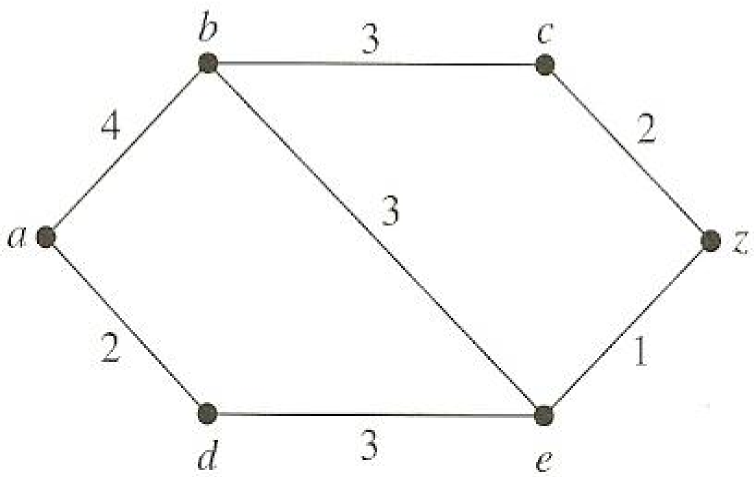
Weighted Graphs

- **Graphs that have a number assigned to each edge are called weighted graphs**



Shortest-Path Problem

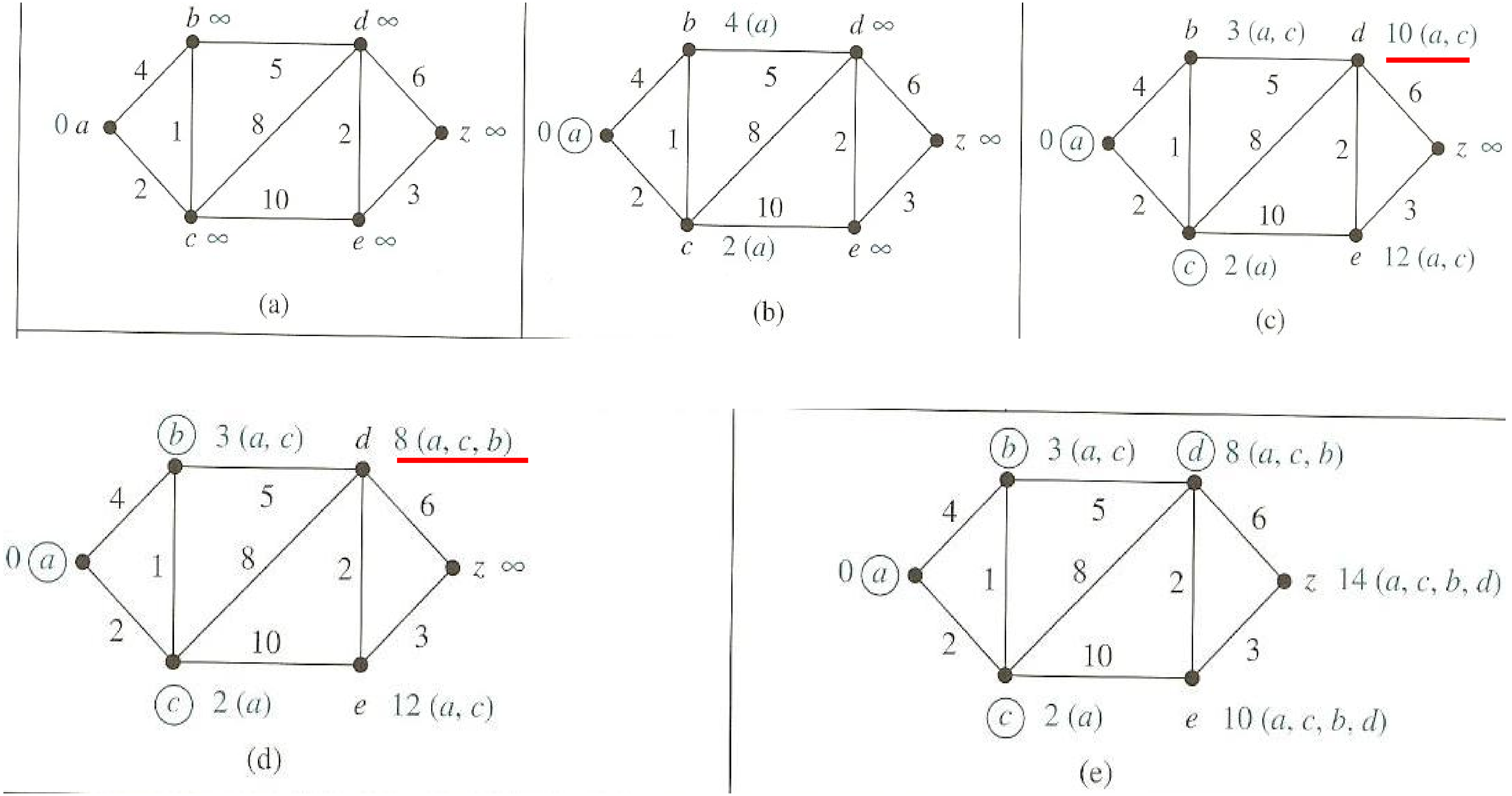
- **What is the length of the shortest path between vertex a and z?**
  - (a,d) = 2, (a,b) = 4 ⇒ d
  - (a,b) = 4, (a,d,e) = 5 ⇒ b
  - (a,d,e) = 5, (a,b,e) = 7, (a,b,c) = 7 ⇒ e
  - (a,d,e,z) = 6, (a,b,c) = 7 ⇒ z
  - (a,b,c) = 7, (a,d,e,z,c) = 8 ⇒ c



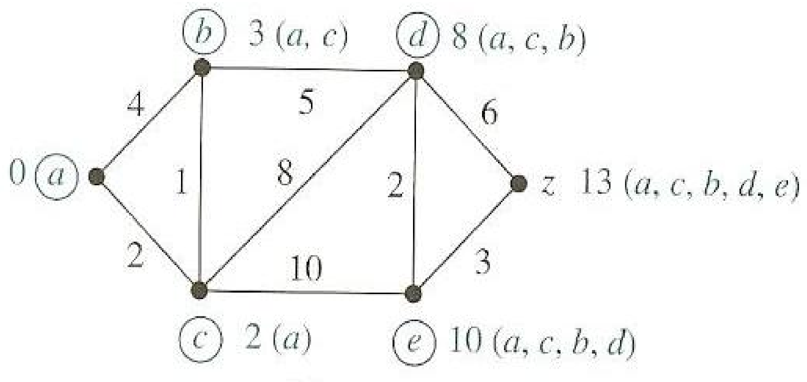
Dijkstra's Shortest-Path Algorithm

procedure *Dijkstra*(G: weighted connected simple graph, with
all weights positive)
{G has vertices $a = v_0, v_1, \ldots, v_n = z$ and weights $w(v_i, v_j)$
where $w(v_i, v_j) = \infty$ if $\{v_i, v_j\}$ is not an edge in G}
for $i := 1$ **to** n
  $L(v_i) := \infty$
$L(a) := 0$
$S := \emptyset$
{the labels are now initialized so that the label of a is 0 and all
other labels are ∞, and S is the empty set}
while $z \notin S$
begin
  $u :=$ a vertex not in S with $L(u)$ minimal
  $S := S \cup \{u\}$
  for all vertices v not in S
    if $L(u) + w(u, v) < L(v)$ **then** $L(v) := L(u) + w(u, v)$
  {this adds a vertex to S with minimal label and updates the
  labels of vertices not in S}
end {$L(z) =$ length of a shortest path from a to z}

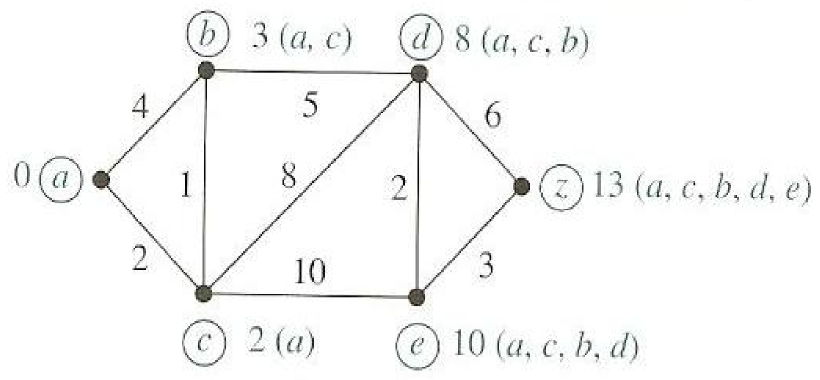
Example (1/2)


b ∞
d ∞
0 a
z ∞
c ∞
e ∞
(a)
b 4 (a)
d ∞
0 (a)
z ∞
c 2 (a)
e ∞
(b)
b 3 (a, c)
d 10 (a, c)
0 (a)
z ∞
(c) 2 (a)
e 12 (a, c)
(c)
(b) 3 (a, c)
d 8 (a, c, b)
0 (a)
z ∞
(c) 2 (a)
e 12 (a, c)
(d)
(b) 3 (a, c)
(d) 8 (a, c, b)
0 (a)
z 14 (a, c, b, d)
(c) 2 (a)
e 10 (a, c, b, d)
(e)

Example (2/2)



(f)



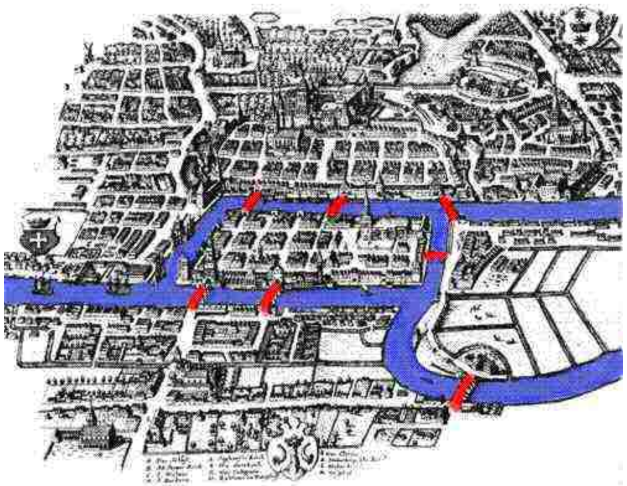
(g)

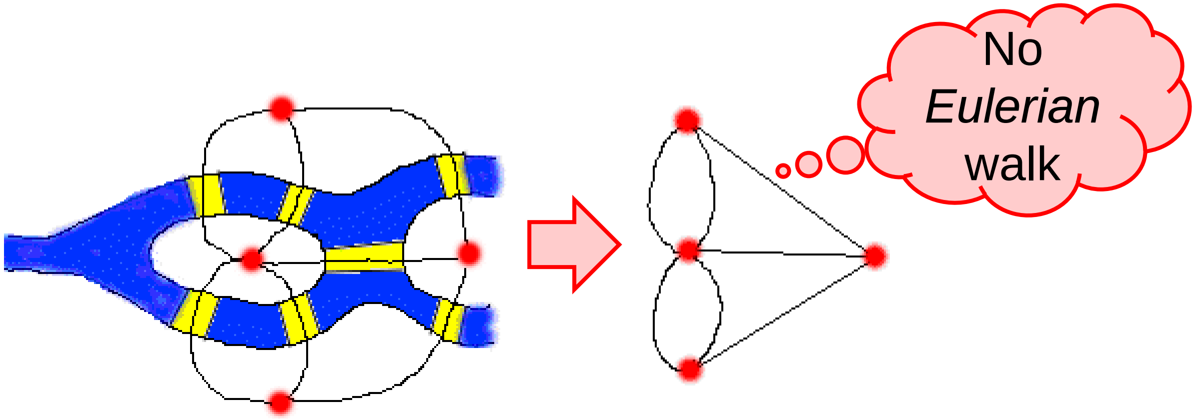
Summary of Dijkstra's Algorithm

- **It can find the length of a shortest path in a connected simple undirected weighted graph**
  - **Single-source all-destination**
  - the edge weight **MUST** be **nonnegative**
- **Its time complexity is O(n^2)**

Salute to Euler!

- **One of the most fundamental and expressive of combinatorial structures is the graph.**
  - Invented by L. Euler based on his proof on the Königsberg bridge problem (the seven bridge problem) in 1736.
    - Is it possible to walk across all the bridges exactly once and return to the starting land area?
    - Abstraction!





L. Euler, Solutioproblematis ad geometriamsituspertinentis, *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, Vol. 8, pp. 128—140, 1736 (published 1741).

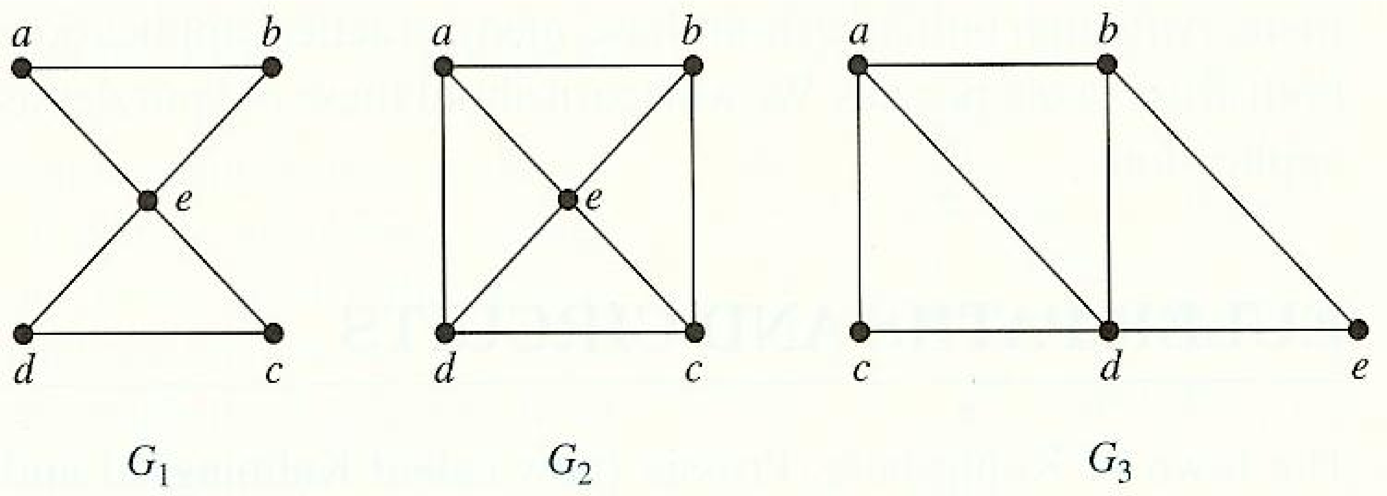
Euler and Hamilton Circuits/Paths

- **Is it possible to travel alone the edges of a graph starting at a vertex and returning to it by traversing each edge exactly once?**
  - **Euler** circuit

- **Is it possible to travel alone the edges of a graph starting at a vertex and returning to it by traversing each vertex exactly once?**
  - **Hamilton** circuit

Euler Circuits and Paths (1/2)

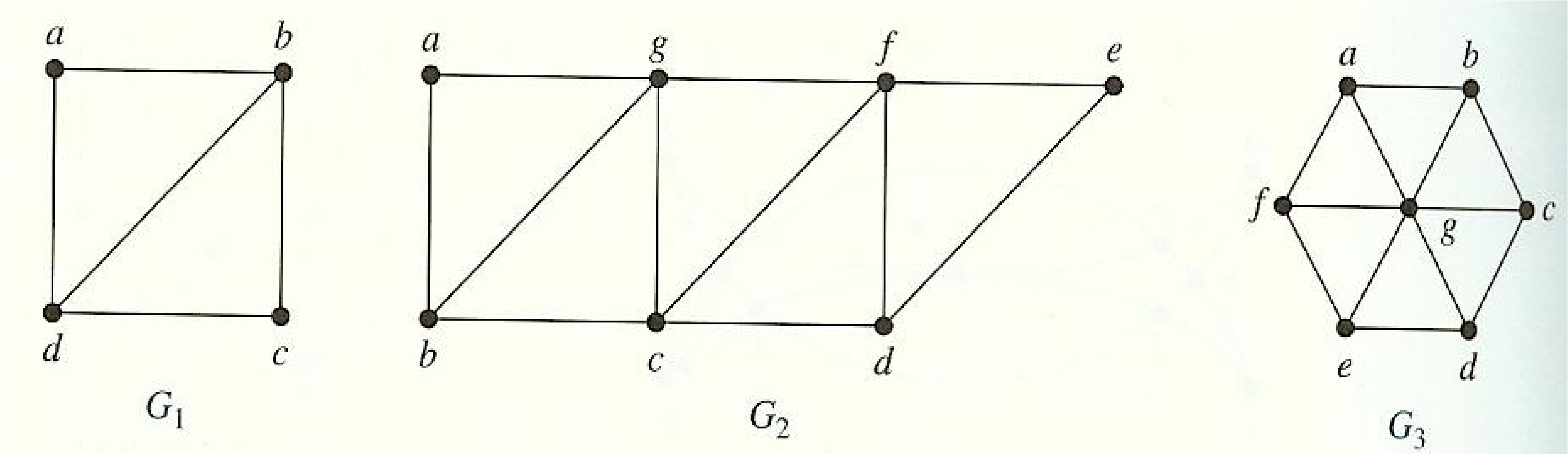
- An Euler circuit in a graph *G* is a simple circuit (cycle) containing every edge of *G*
- An Euler path in a graph *G* is a simple path containing every edge of *G*



Euler Circuits and Paths (2/2)

- **Theorem: A connected mulltigraph (or graph) has an Euler circuit iff each of its vertices has even degree**

  A connected mulltigraph (or graph) has an Euler path but not an Euler circuit iff it has exactly 2 vertices of odd degree



Hamilton Paths and Circuits

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

- **Hamilton circuit**
  - a circuit $x_0, x_1, \ldots, x_{n-1}, x_n, x_0$ is a Hamilton circuit if $x_0, x_1, \ldots, x_{n-1}, x_n$ is a Hamilton path

Hamilton's "A Voyage Round the World"

- Invented in 1857

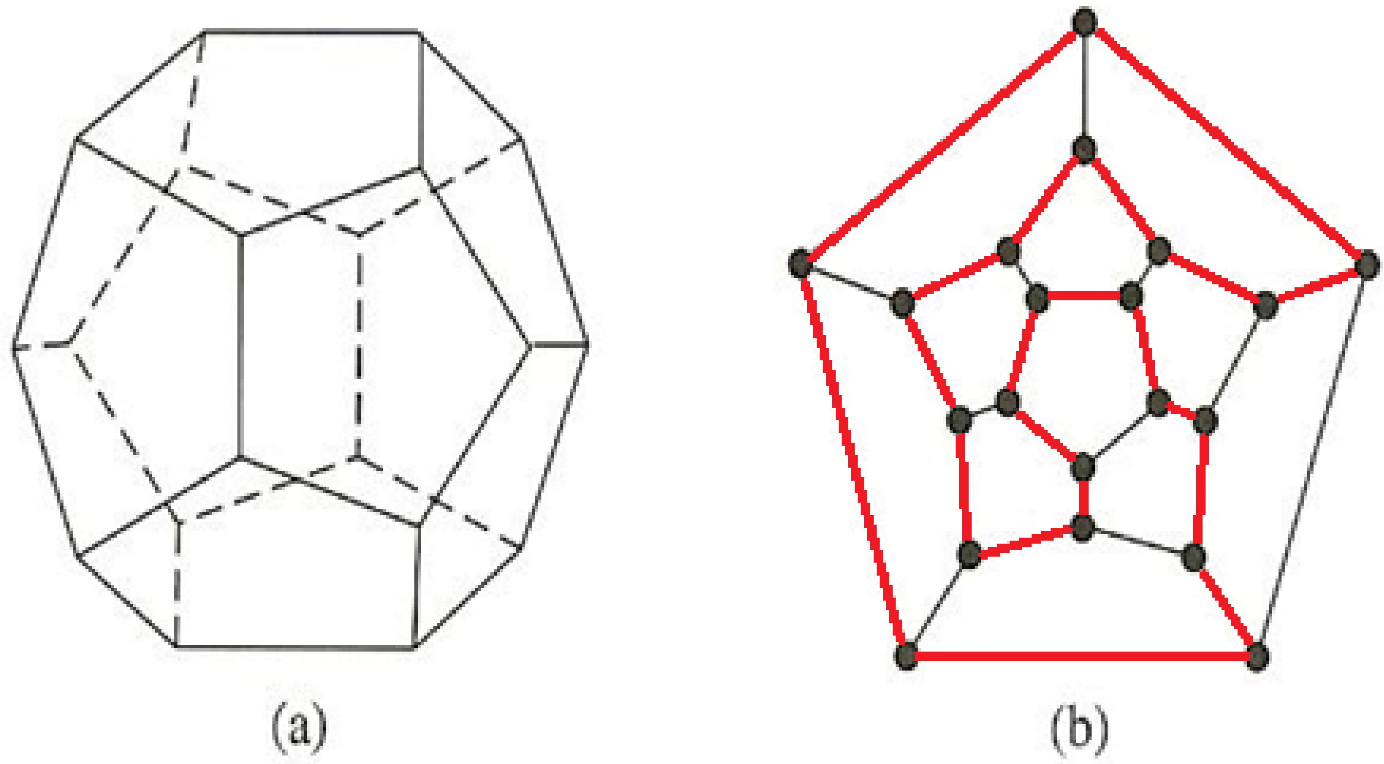


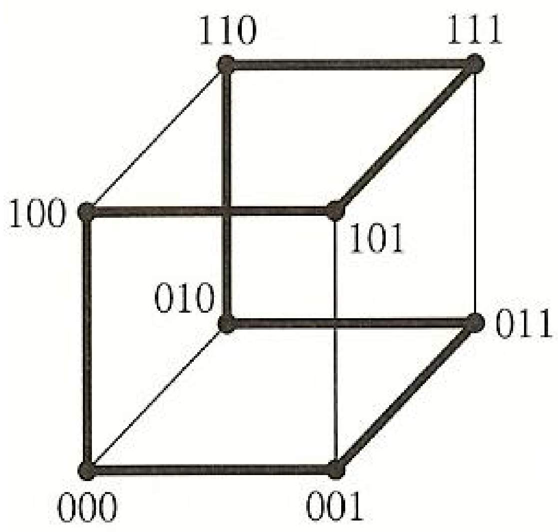
FIGURE 8 Hamilton's "A Voyage Round the World" Puzzle.

IS there a Hamilton Circuit?

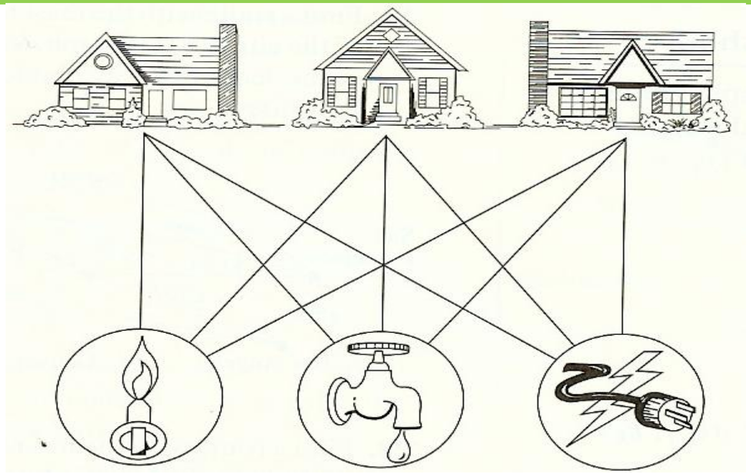
- **There are no known simple necessary and sufficient criteria**
- **However,**
  - some theorems can provide sufficient conditions
  - some properties can show there is no Hamilton circuit in a graph
    - a graph with a vertex of degree 1

Gray Codes

- **For 3-bit codes (8 different combinations)**
  - binary code (000, 001, 010, 011, 100, 101, 110, 111)
    - the largest hamming distance between 2 adjacent codes is 3 (011->100)
  - gray code (000, 001, 011, 010, 110, 111, 101 100)
    - the hamming distance between any 2 adjacent codes is always 1
- ***n*-bit Gray code ➔**
  finding a Hamilton circuit in Q_n



Planar Graphs



- A graph is called planar if it can be drawn in a plan without any edges crossing
- Caution!
  Not what you think at the first sight!

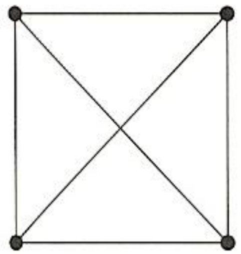


FIGURE 2 The Graph K_4.

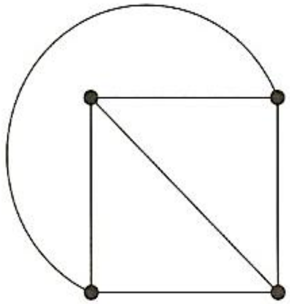


FIGURE 3 K_4 Drawn with No Crossings.

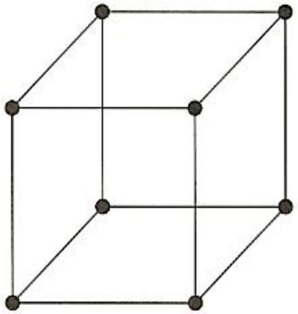


FIGURE 4 The Graph Q_3.

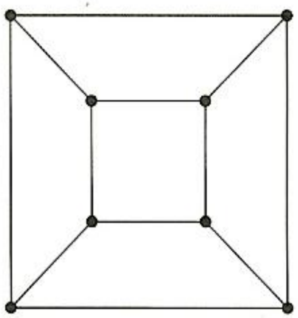
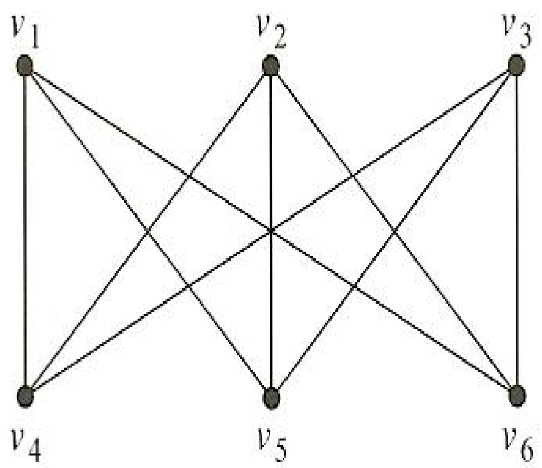
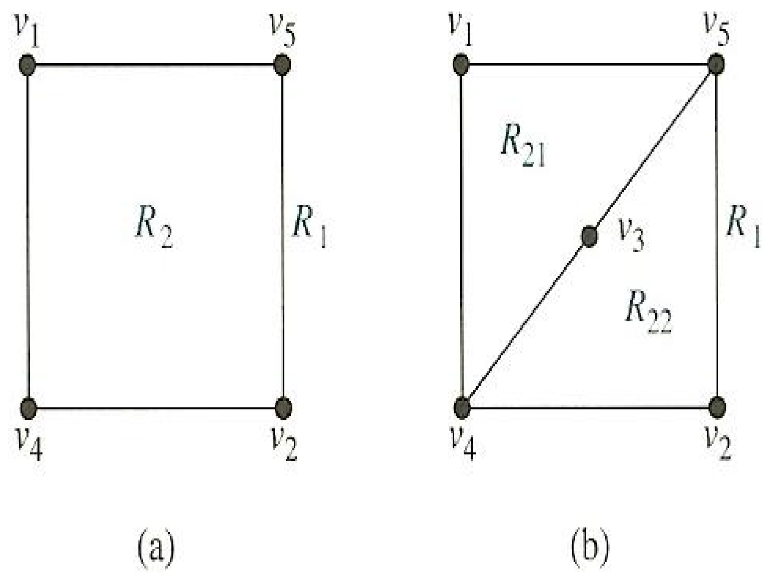


FIGURE 5 A Planar Representation of Q_3.

Planarity of Graph

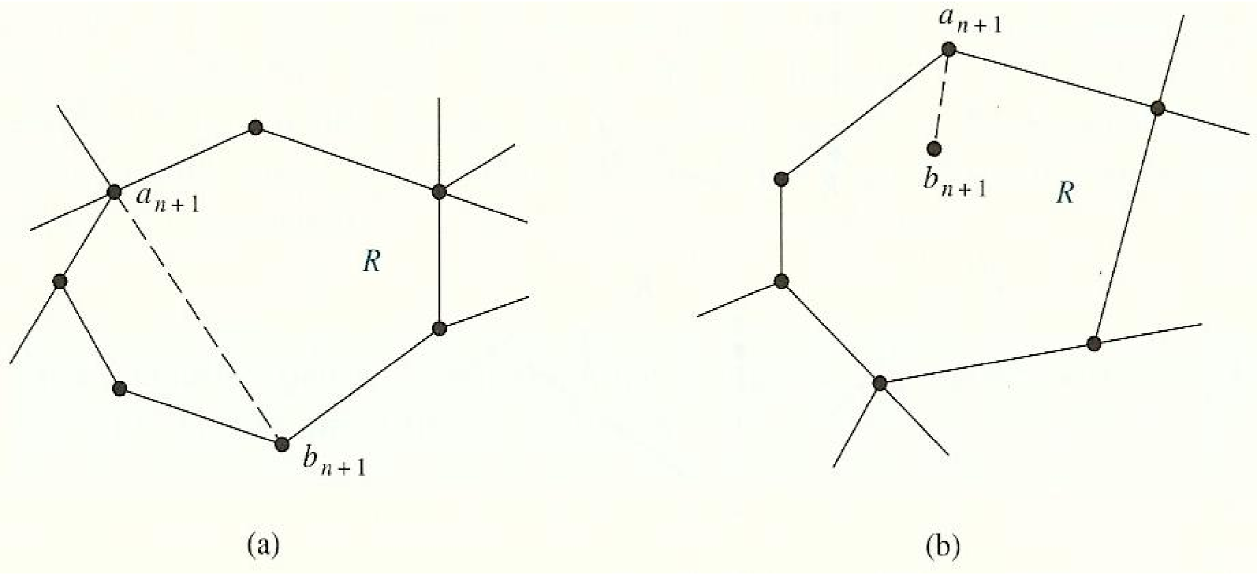
- **Is $K_{3,3}$ Planar?**





Euler's Formula

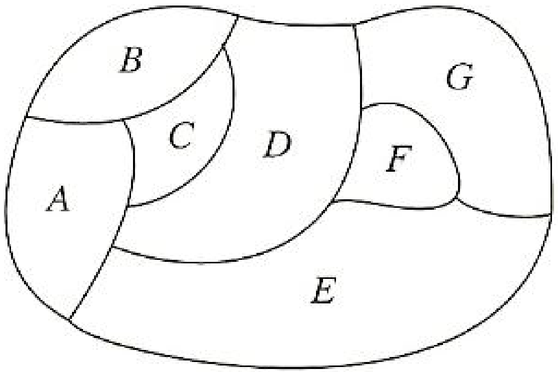
- **Euler showed that all planar representation of a graph split the plane into the same number of regions**
- **Let G be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then,**
  r = e-v+2
  - use math induction

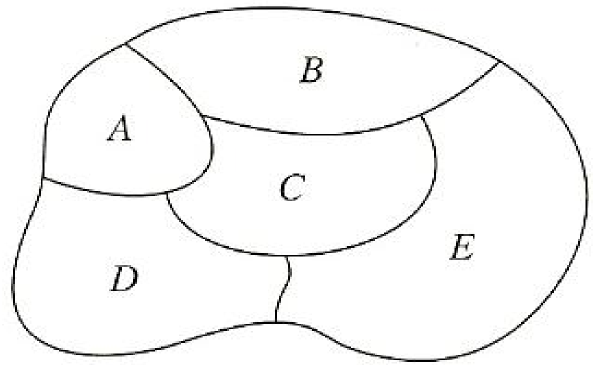


Graph Coloring (1/2)

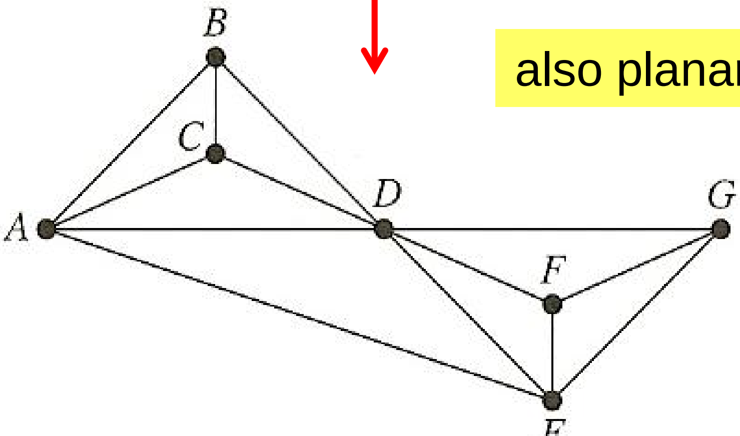
- **What is graph coloring?**

maps of regions

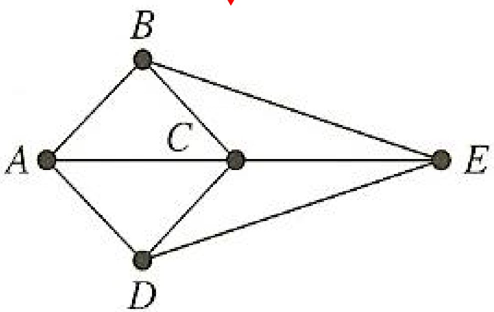




- **Dual graphs**



also planar !



Graph Coloring (2/2)

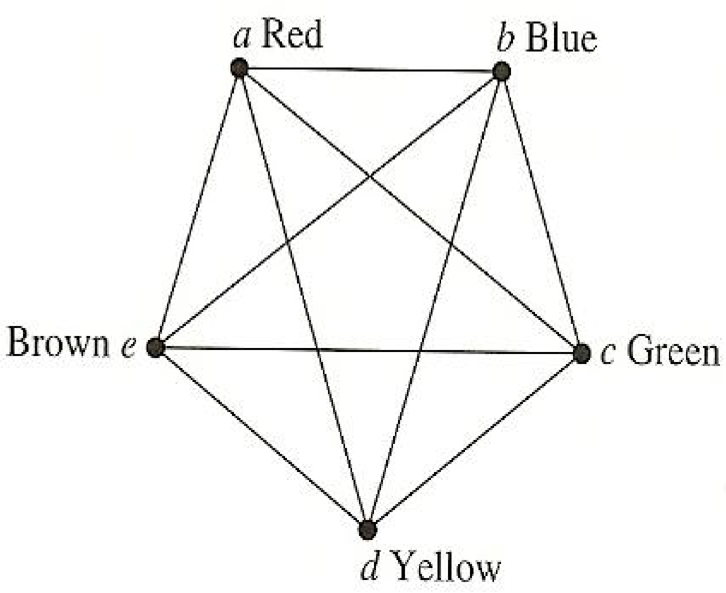
- **A coloring of a simple graph is the assignment of a color to each vertex so that no 2 adjacent vertices are assigned the same color**

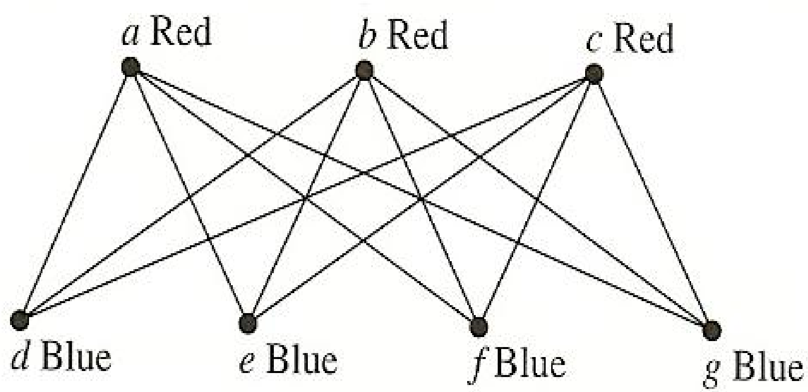
- **The chromatic number of a graph is the least number of colors needed for a coloring**

- **The Four Color Theorem**
  - the chromatic number of a **planar** graph is **$\leq$ 4**

Chromatic Numbers for K_n and $K_{m,n}$

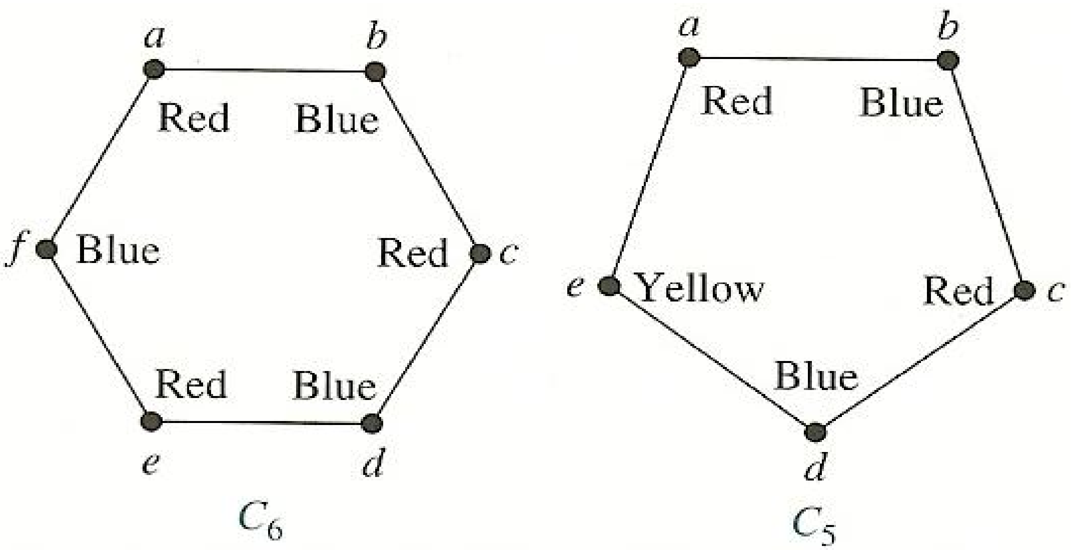
- **The chromatic number of K_n is n**
- **The chromatic number of $k_{m,n}$ is 2**





Chromatic Numbers for C_n

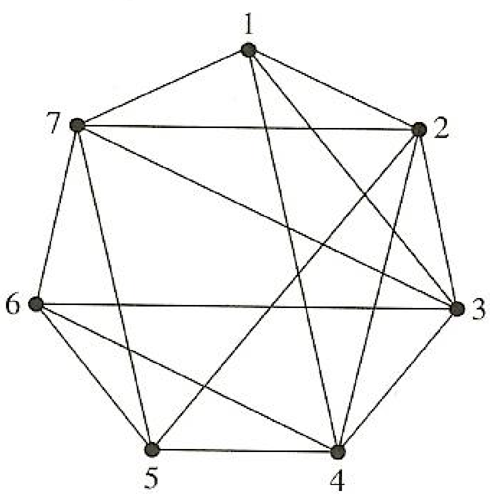
- **The chromatic number of C_n is**
  - 2 if n is even
  - 3 if n is odd

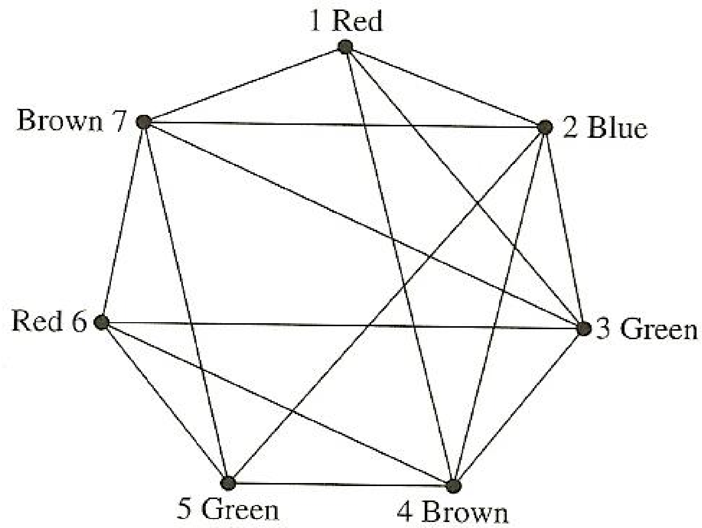


- **However, it's hard to find the chromatic number for an arbitrary graph**

Applications of Graph Colorings

- **Scheduling final exams**





Time Period	Courses
I	1, 6
II	2
III	3, 5
IV	4, 7

63 Two More Proofs

Existence of Euler Circuits (1/2)

- **Theorem: A connected multigraph has an Euler circuit if and only if each of its vertices has even degree.**
- **Pf:**

($\Rightarrow$) Suppose there is an Euler circuit. Think of following the edges on the circuit. Each vertex is traversed by different edges (possibly many times). That is, the circuit enters and leaves a vetex by different edges. The degree of each vertex on the circuit must be even. Since the graph is connected, it follows that all vertices are on the circuit. □

Existence of Euler Circuits (2/2)

- **Pf:**

($\Leftarrow$) Suppose each vertice has even degree. We will construct an Euler circuit. Starting from any vertex, we build a path consisting of different edges as long as possible. Since each vertex has even degree, the path can enter and then leave any vertex other than the initial vertex safely. Hence the path must be a circuit.
If all edges are contained in the circuit, we have found an Euler circuit. Otherwise, we consider the subgraph by removing all edges in the circuit. In the connected components of the subgraph, all vertices of the subgraph still have even degree. Hence we construct more circuits from each connected components. An Euler circuit can be constructed by merging the circuits. $\square$

Existence of Euler Paths

- **A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree.**
- **Pf:**
  - $\Rightarrow$
  - Similar to the previous Theorem, we observe that all but the starting and ending vertices on the path have even degree.
  - Since the graph is connected, the result follows.
  - $\Leftarrow$
  - Let the vertices with odd degree be u and v.
  - Consider $G' = (V, E \cup \{\{u, v\}\})$. Then all vertices in G' have even degree.
  - Applying the previous Theorem, we have an Euler circuit passing the edge $\{u, v\}$.
  - An Euler path in the original graph is found by removing the edge.

Euler's Formula (1/2)

- **Let $G = (V, E)$ be a connected planar graph. Then there are $|E| - |V| + 2$ regions.**
- **Pf:**

We prove by constructing a sequence $G_0 = (V_0, E_0), G_1 = (V_1, E_1), \ldots, G_{|E|-1} = G$ of subgraphs of G and show that there are $|E_i| - |V_i| + 2$ regions for all G_i's.
Initially, we choose any edge in E and call it G_0. G_i is built by adding a new edge incident to G_{i-1}. Since G is connected, we obtain the original graph G after all edges are added.
To show there are $|E_i| - |V_i| + 2$ regions for all $G_i = (V_i, E_i)$, we proceed by induction.
BASIS STEP. $i = 0$. Clearly. Since $|E_i| = 1$ and $|V_i| = 2$ and there is only but one region, we have $1 = 1 - 2 + 2$. □

Euler's Formula (2/2)

- **Pf:**

INDUCTIVE STEP. Assume there are $|E_k| - |V_k| + 2$ regions determined by G_k. After the edge $\{u, v\}$ is added, there are two cases:

- $u \in V_k$ but $v \notin V_k$. Then the number of regions remains the same. We have $|E_{k+1}| = |E_k| + 1$ and $|V_{k+1}| = |V_k| + 1$. Hence $|E_{k+1}| - |V_{k+1}| + 2 = (|E_k| + 1) - (|V_k| + 1) + 2 = |E_k| - |V_k| + 2$.
- $u, v \in V$. Then we have a new region. Moreover, $|E_{k+1}| = |E_k| + 1$ and $|V_{k+1}| = |V_k|$. Hence $|E_{k+1}| - |V_{k+1}| + 2 = (|E_k| - |V_k| + 2) + 1$.