Discrete Mathematics
Chapter 3 : Graph Theory

GATE CS Lectures
by Monalisa

Section1: Engineering Mathematics

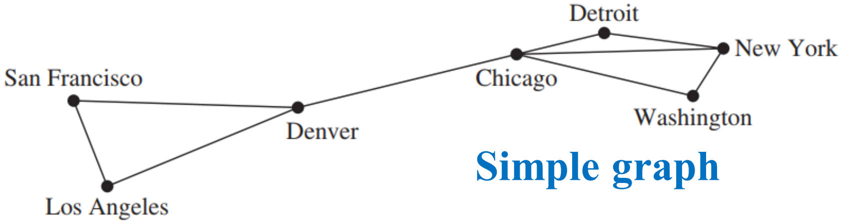
- **Discrete Mathematics**: Propositional and first order logic. Sets, relations, functions, partial orders and lattices. Monoids, Groups. Graphs: connectivity, matching, coloring. Combinatorics: counting, recurrence relations , generating functions.
- **Linear Algebra**: Matrices, determinants, system of linear equations, eigenvalues and eigenvectors, LU decomposition.
- **Calculus**: Limits, continuity and differentiability. Maxima and minima. Mean value theorem. Integration.
- **Probability and Statistics**: Random variables. Uniform, normal, exponential, poisson and binomial distributions. Mean, median, mode and standard deviation. Conditional probability and Bayes theorem.

- **Discrete Mathematics**: Propositional and first order logic. Sets, relations, functions, partial orders and lattices.Monoids, Groups.Graphs: connectivity, matching, coloring.Combinatorics : counting, recurrence relations , generating functions.
- Chapter 1: Logic
- Propositional Logic, Propositional Equivalences , Predicates and Quantifiers , Nested Quantifiers , Rules of Inference , Introduction to Proofs.
- Chapter 2 : Set Theory
- Sets, relations, functions, partial orders and lattices.Monoids, Groups.
- Chapter 3 : Graph Theory
- Graphs: connectivity, matching, coloring.
- Chapter 4 : Combinatorics

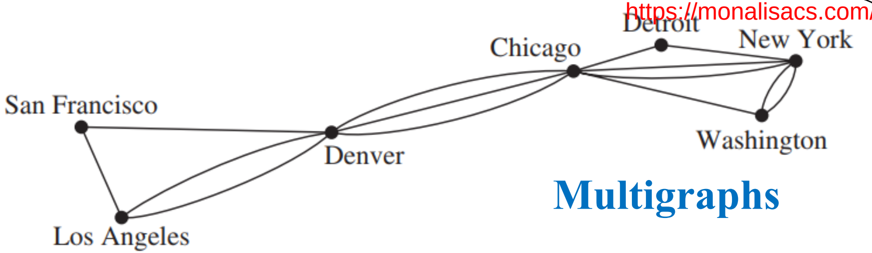
- **Chapter 3 : Graph Theory**
- 3.1 Graph Terminology and Special Types of Graphs
- 3.2 Representing Graphs and Graph Isomorphism
- 3.3 Connectivity
- 3.4 Euler and Hamilton Paths
- 3.5 Planar Graphs
- 3.6 Graph Coloring

3.1 Graph Terminology and Special Types of Graphs

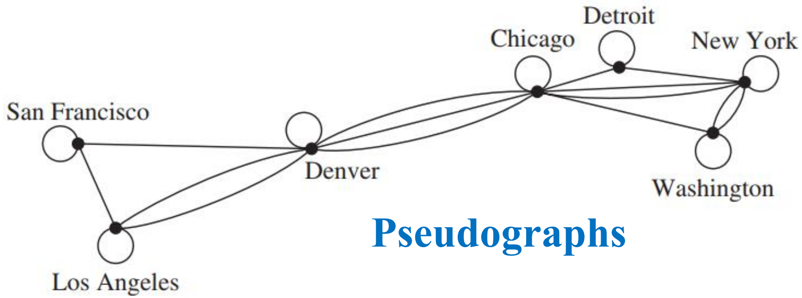
- **DEFINITION 1** A *graph G = (V, E)* consists of *V* , a nonempty set of *vertices* (or *nodes*) and *E*, a set of *edges*. Each edge has either one or two vertices associated with it, called its *endpoints*. An edge is said to *connect* its endpoints.
- |V|=Set of vertices ,|E|=Set of edges
- Two types of graph : Directed & Undirected graphs
- A graph with an infinite vertex set or an infinite number of edges is called an **infinite graph**, and, a graph with a finite vertex set and a finite edge set is called a **finite graph**.
- A graph in which each edge connects two different vertices and where no two edges connect the same pair of vertices is called a **simple graph**.
- Graphs that may have **multiple edges** connecting the same vertices are called **multigraphs.**
- When a node connects to itself called **loop**.
- Graphs that include loops, and multiple edges connecting the same pair of vertices or a vertex to itself, are sometimes called **pseudographs**.
- A graph without loop and multiple edges called Simple 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*.



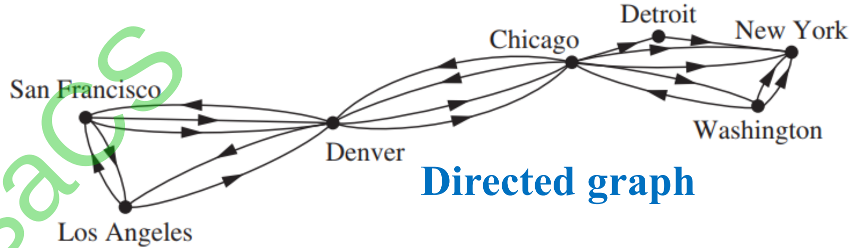
Simple graph



Multigraphs



Pseudographs



Directed graph

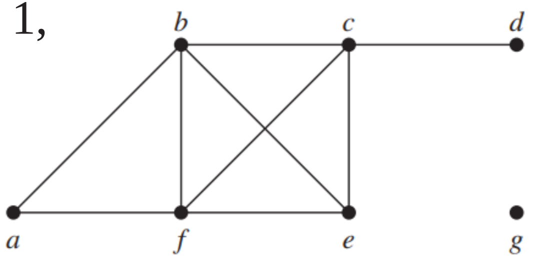
- Directed graphs that may have **multiple directed edges**. We called such graphs **directed multigraphs**.
- When there are *m* directed edges, each associated to an ordered pair of vertices *(u, v)*, we say that *(u, v)* is an edge of **multiplicity** *m*.
- For some models we may need a graph where some edges are undirected, while others are directed.A graph with both directed and undirected edges is called a **mixed graph**.

TABLE 1 Graph Terminology.

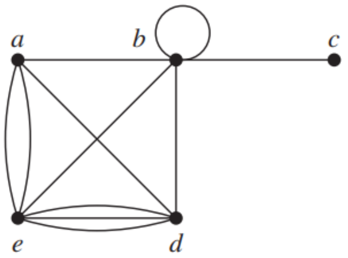
Type	*Edges*	*Multiple Edges Allowed?*	*Loops Allowed?*
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed and undirected	Yes	Yes

- **DEFINITION 3** 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.
- **DEFINITION 4** 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).
- **DEFINITION 5** 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, $N(A) = \cup_{v \in A} N(v)$

- **EXAMPLE 1** What are the degrees and neighborhoods of the vertices in the graphs G and H
- *Solution:* In G, $\deg(a) = 2$, $\deg(b) = \deg(c) = \deg(f) = 4$, $\deg(d) = 1$, $\deg(e) = 3$, and $\deg(g) = 0$.
- $N(a) = \{b, f\}$, $N(b) = \{a, c, e, f\}$, $N(c) = \{b, d, e, f\}$, $N(d) = \{c\}$,
- $N(e) = \{b, c, f\}$, $N(f) = \{a, b, c, e\}$, and $N(g) = \emptyset$.
- In H, $\deg(a) = 4$, $\deg(b) = \deg(e) = 6$, $\deg(c) = 1$, and $\deg(d) = 5$.
- $N(a) = \{b, d, e\}$, $N(b) = \{a, b, c, d, e\}$, $N(c) = \{b\}$, $N(d) = \{a, b, e\}$, and $N(e) = \{a, b, d\}$.
- A vertex of degree zero is called **isolated**. It follows that an isolated vertex is not adjacent to any vertex.
- A vertex is **pendant** if and only if it has degree one.
- **THEOREM 1 THE HANDSHAKING THEOREM**
- Let $G = (V, E)$ be an undirected graph with m edges. Then $2m = \sum_{v \in V} \deg(v)$.
- (Note that this applies even if multiple edges and loops are present.)



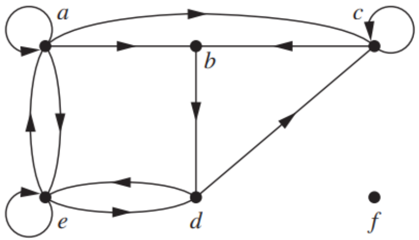
G



H

- **EXAMPLE 2** How many edges are there in a graph with 10 vertices each of degree six?
- *Solution:* $2m = 10*6 = 60$, where m is the number of edges. Therefore, $m = 30$.

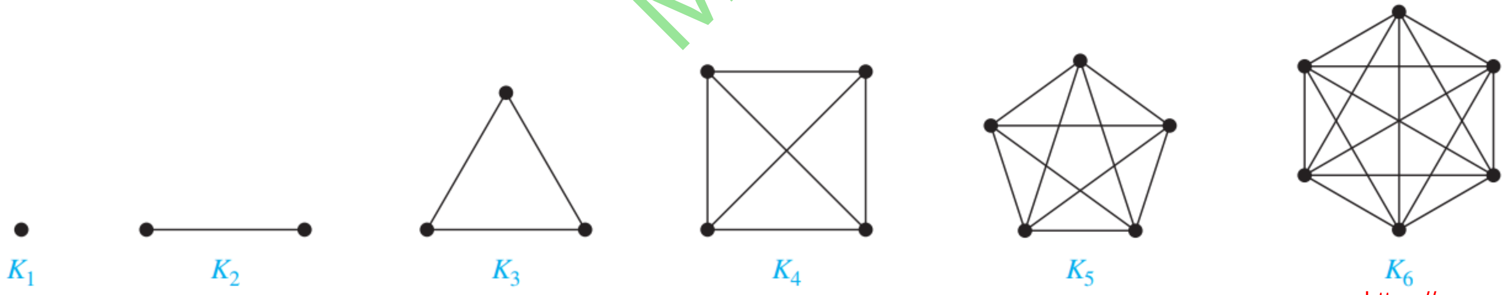
- **THEOREM 2** An undirected graph has an even number of vertices of odd degree .
- **DEFINITION 6** When *(u, v)* is an edge of the graph *G* with directed edges, *u* is said to be *adjacent to v* and *v* is said to be *adjacent from u*. The vertex *u* is called the *initial vertex* of *(u, v)*, and *v* is called the *terminal* or *end vertex* of *(u, v)*. The initial vertex and terminal vertex of a loop are the same.
- **DEFINITION 7** 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 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)
- **EXAMPLE 3** Find the in-degree and out-degree of each vertex in the graph *G* with directed edges shown in Figure .



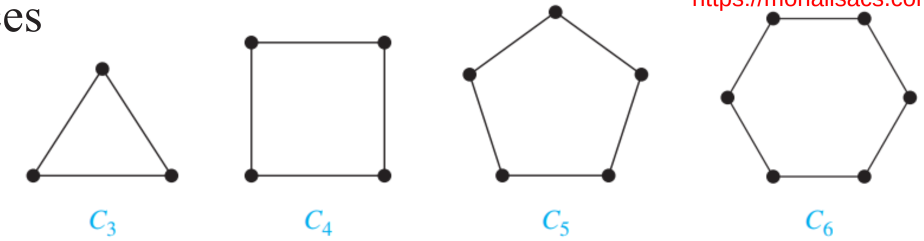
G

- *Solution:* $\deg^-(a) = 2$, $\deg^-(b) = 2$, $\deg^-(c) = 3$, $\deg^-(d) = 2$, $\deg^-(e) = 3$, and $\deg^-(f) = 0$.
- $\deg^+(a) = 4$, $\deg^+(b) = 1$, $\deg^+(c) = 2$,$\deg^+(d) = 2$, $\deg^+(e) = 3$, and $\deg^+(f) = 0$.
- **THEOREM 3** Let *G = (V, E)* be a graph with directed edges. Then
- $\sum_{v\in V} \deg^-(v) = \sum_{v\in V} \deg^+(v) = |E|$.

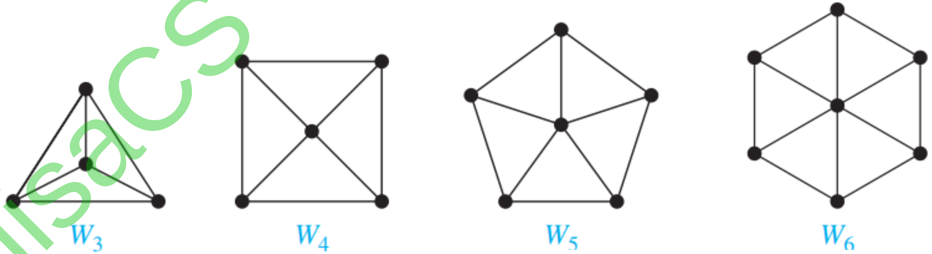
- **Some Special Simple Graphs**
- Maximum numbers of edge possible= $^nC_2 = \frac{n(n-1)}{2}$
- Number of Simple graph possible with n vertices $=2^{\text{Maximum number of edges}}$
- Number of Simple graph possible with n vertices and m edges $=C(\frac{n(n-1)}{2}, m)$
- Several classes of simple graphs
- **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. A simple graph for which there is at least one pair of distinct vertex not connected by an edge is called **noncomplete**.
- In complete graph degree of each vertex is n-1.
- Number of edges in $K_n = {}^nC_2 = \frac{n(n-1)}{2}$
- **The Graphs K_n for $1 \le n \le 6$.**



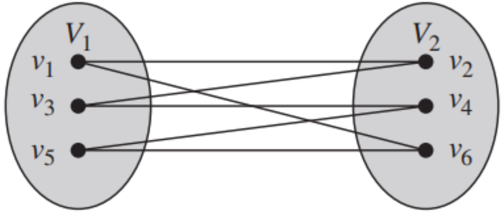
- **Cycles:** A **cycle C_n**, $n \geq 3$, consists of n vertices $v_1, v_2, \ldots, v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}$, and $\{v_n, v_1\}$.
- **The Cycles C_3, C_4, C_5, and C_6.**



- **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.**

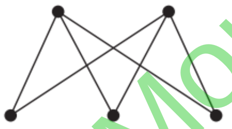


- **Bipartite Graphs:** A simple graph G is called *bipartite* if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2. When this condition holds, we call the pair (V_1, V_2) a *bipartition* of the vertex set V of G.

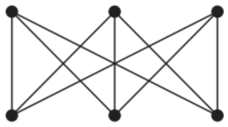


- **EXAMPLE 4** C_6 is bipartite, as shown in Figure, because its vertex set can be partitioned into the two sets $V_1 = \{v_1, v_3, v_5\}$ and $V_2 = \{v_2, v_4, v_6\}$, and every edge of C_6 connects a vertex in V_1 and a vertex in V_2.

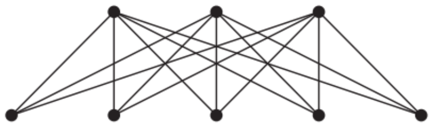
- **THEOREM 4** A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.
- **Complete Bipartite Graphs** A **complete bipartite graph** $K_{m,n}$ is a graph that has its vertex 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.
- Complete bipartite graph ≠Complete graph .$K_{m,n}$ is a complete graph if m=n=1 ,$K_{1,1}=K_2$
- $K_{m,n}$ has m+n vertices and mn edges .



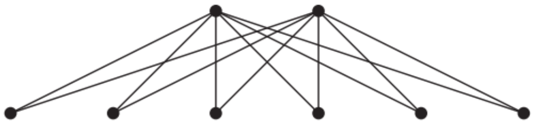
$K_{2,3}$



$K_{3,3}$



$K_{3,5}$



$K_{2,6}$

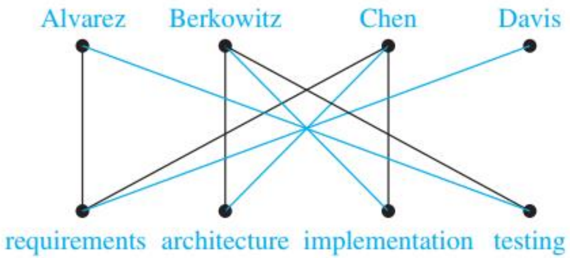
- **THEOREM 5** Bipartite graph have no cycles of odd length .
- Maximum number of edges possible in a bipartite graph with n vertices= $\left\lfloor \frac{n^2}{4} \right\rfloor$
- **EXAMPLE 5** In a graph G ,n=10 then maximum number of edges =$10^2/4$ =25
- $K_{5,5}$=5*5=25 , $K_{4,6}$=4*6=24 , $K_{3,7}$=3*7=21 , $K_{2,8}$=2*8=16
- **Cyclic Graph:** A graph with at least one cycle is called cyclic graph.
- **ACyclic Graph:** A graph with no cycle is called acyclic graph.
- **Tree:** A connected acyclic graph is called a tree .A tree with n vertices has n-1 edges
- **Complement of graph:** Let G be a simple graph with n vertices .The complement of G denoted by $\bar{G}$ is a simple graph with same vertices as that of G but two vertices u and v are adjacent in $\bar{G}$ if and only if u & v are not adjacent in G.
- For G and $\bar{G}$ no edges are common . $|E(G)|+|E(\bar{G})|=|E(K_n)|$
- **EXAMPLE 6** Let G is a simple graph with 10 vertices and 18 edges , Find number of edges in $\bar{G}$?
- $18+ |E(\bar{G})|={}^{10}C_2=45$
- $\Rightarrow|E(\bar{G})|=45-18=27$
- **EXAMPLE 7** Let G is a simple graph with 24 edges , $\bar{G}$ has 54 edges. Find number of vertices?
- $24 + 54={}^{n}C_2=78$
- $n(n-1)=156 \Rightarrow n=13$

- **EXAMPLE 8** 24 edges . Degree of each vertex=k .Which of the following is possible numbers of vertices in a simple graph. (a)10 (b)15 (c)16 (d)6
- n*k=2*|E|=2*24=48
- k=1,n=48 k=2,n=24 k=3,n=16 k=4,n=12 k=6,n=8
- Ans : (c)16
- **EXAMPLE 9** Maximum number of vertices possible in a simple graph with 35 edges & degree of each vertex at least 3 is _________ ?
- 3|v|≤2*35
- |v| ≤ 70/3=23
- **EXAMPLE 10** Which of the fallowing degree sequence represent a simple undirected graph?
- A){2,3,3,4,4,5} 3 vertices can't have odd degree ,so it's not a sequence of simple graph.
- B){2,3,4,4,5} A simple graph with 5 vertices has maximum degree 5-1=4 .Not a simple graph.
- C){1,3,3,4,5,6,6}In a simple graph with 7 vertices if 2 vertices have degree 6 then these 2 vertices are adjacent to all other vertices so vertices with degree 1 is not possible .
- D){0,1,2,3,…..n-1} in a simple graph with n vertices if one vertex have degree n-1 then a vertex with degree 0 is not possible . Not a simple graph.

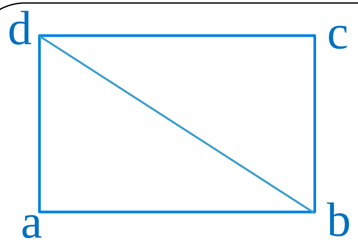
- Havel-Hakimi Algorithm:
- The nonincreasing sequence $(d_1,d_2,\ldots,d_n)$ is simple graph if and only if the sequence
- $(d_{2-1},d_{3-1},\ldots,d_{d1+1}-1,d_{d1+2},d_{d1+3},\ldots,d_n)$ is also simple graph.
- **Note:** The sequence obtained after applying theorem might not be nonincreasing. In such a case, you will have to rearrange it in nonincreasing order before re-applying the theorem.
- **EXAMPLE 11**
- {6,6,6,6,4,3,3,0}
- {5,5,5,3,2,2,0}
- {4,4,2,1,1,0}
- {3,1,0,0,0}
- The last result can't represent by a simple graph ,hence the sequence is not a simple graph.
- {6,5,5,4,3,3,2,2,2}
- {4,4,3,2,2,1,2,2}⇒{4,4,3,2,2,2,2,1}
- {3,2,1,1,2,2,1}⇒{3,2,2,2,1,1,1}
- {1,1,1,1,1,1}
- It's not connected but a simple graph .

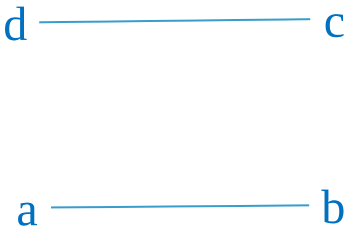
Bipartite Graphs and Matchings

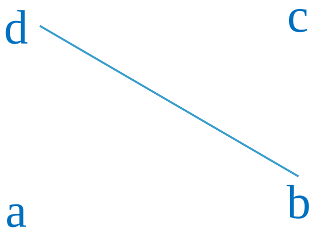
- **Job Assignments** Suppose that there are m employees in a group and n different jobs that need to be done, where $m \geq n$.
- Each employee is trained to do one or more of these n jobs.
- Suppose that a group has four employees: Alvarez, Berkowitz, Chen, and Davis;
- Project : requirements, architecture, implementation, and testing.
- Alvarez has been trained to do requirements and testing;
- Berkowitz has been trained to do architecture, implementation, and testing;
- Chen has been trained to do requirements, architecture, and implementation;
- Davis has only been trained to do requirements.
- We model these employee capabilities using the bipartite graph.



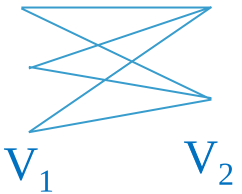
- Finding an assignment of jobs to employees can be thought of as finding a matching in the graph model,
- Where a **matching** M in a simple graph $G = (V, E)$ is a subset of the set E of edges of the graph such that no two edges are incident with the same vertex.
- In other words, a matching is a subset of edges such that if $\{s, t\}$ and $\{u, v\}$ are distinct edges of the matching, then s, t, u, and v are distinct.

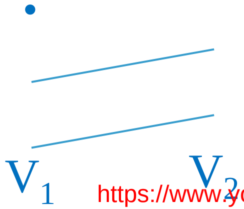




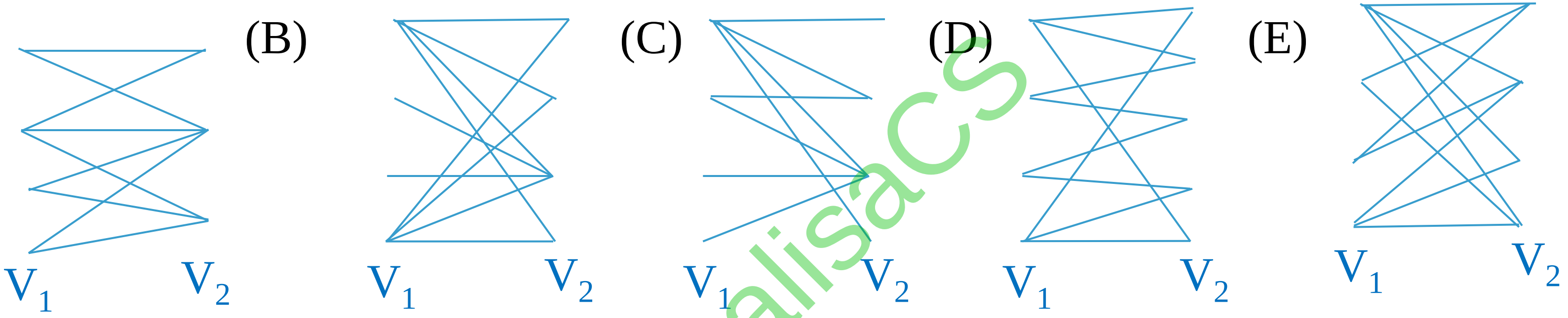


- Graph G M_1 M_2 M_3 M_4
- A vertex that is the endpoint of an edge of a matching *M* is said to be **matched** in *M*; otherwise, it is said to be **unmatched**.
- If deg(v)=1 **matched** , If deg(v)=0 **unmatched.**
- A **maximum matching** is a matching with the largest number of edges(Ex:M_1,M_2)
- The number of edges in Maximum matching called matching numbers.
- We say that a matching *M* in a bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) is a **complete matching from** V_1 **to** V_2 if every vertex in V_1 is the endpoint of an edge in the matching.
- A matching from V_1 **to** V_2 is said to be complete if every vertex in V_1 is matched.
- Every complete matching in a bipartite graph is a maximum matching but reverse need not be true.
- If $|V_1| \leq |V_2|$ then complete matching exists.
- **EXAMPLE 12** Maximum matching from V_1 to V_2 exist but complete matching doesn't exist.



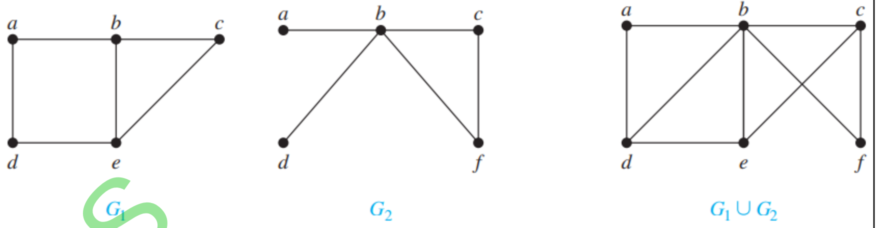


- **THEOREM 6 HALL'S MARRIAGE THEOREM** The bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a complete matching from V_1 to V_2 if and only if $|N(A)| \geq |A|$ for all subsets A of V_1.
- **EXAMPLE 13 F**or which of fallowing graph complete matching exists.
- (A) (B) (C) (D) (E)

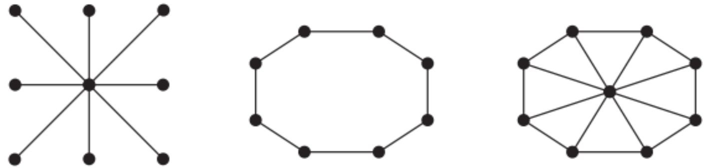


- A)$|V_1| > |V_2|$ so complete matching doesn't exist.
- B)V_1 two vertices are adjacent to V_2 single vertex so complete matching doesn't exist.
- C) V_1 last two vertices are adjacent to V_2 single vertex so complete matching doesn't exist.
- (D) Complete matching possible
- (E) Complete matching possible
- **DEFINITION 8** 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 $H \neq G$.

- **DEFINITION 9** 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 $G_1 \cup G_2$.
- **Some Applications of Special Types of Graphs**



- **EXAMPLE 14 Local Area Networks**
- *star topology*, where all devices are connected to a central control device.
- A local area network can be represented using a complete bipartite graph $K_{1,n}$, as shown in Figure (a). Messages are sent from device to device through the central control device.
- *ring topology*, where each device is connected to exactly two others. Local area networks with a ring topology are modeled using n-cycles, C_n, as shown in Figure (b).
- Finally, some local area networks use a hybrid of these two topologies. Messages may be sent around the ring, or through a central device.
- This redundancy makes the network more reliable. Local area networks with this redundancy can be modeled using wheels W_n, as shown in Figure (c).



(a) (b) (c)

3.2 Representing Graphs and Graph Isomorphism

- **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.
- **EXAMPLE 1** Use adjacency lists to describe the simple graph given in Figure .
- **EXAMPLE 2** Represent the directed graph shown in Figure by listing all the vertices that are the terminal vertices of edges starting at each vertex of the graph

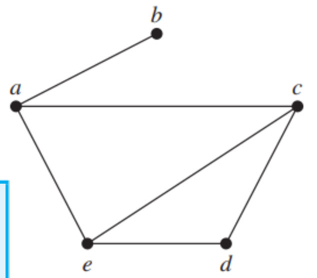


TABLE 1 An Adjacency List for a Simple Graph.

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

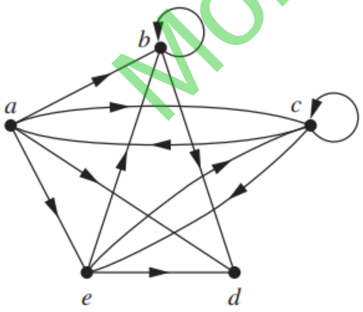
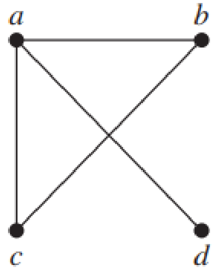


TABLE 2 An Adjacency List for a Directed Graph.

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

Adjacency Matrices

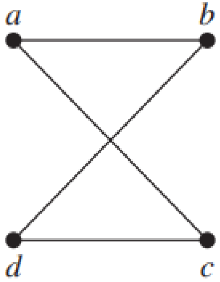
- Suppose that $G = (V, E)$ is a simple graph where $|V| = n$. Suppose that the vertices of G are listed arbitrarily as $v_1, v_2, \ldots, v_n$.
- The **adjacency matrix A** (or $\mathbf{A}_G$) of G, with respect to this listing of the vertices, is the $n \times n$ zero–one matrix with 1 as its (i, j)th entry when v_i and v_j are adjacent, and 0 as its (i, j)th entry when they are not adjacent.
- $a_{ij} = 1$ if $\{v_i, v_j\}$ is an edge of G, 0 otherwise
- **EXAMPLE 3** Use an adjacency matrix to represent the graph shown in Figure



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

- **EXAMPLE 4** Draw a graph with the adjacency matrix

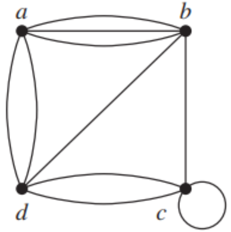
$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$



- The adjacency matrix of a simple graph is symmetric, that is, $a_{ij} = a_{ji}$, because both of these entries are 1 when v_i and v_j are adjacent, and both are 0 otherwise.
- When multiple edges connecting the same pair of vertices v_i and v_j, or multiple loops at the same vertex, are present, the adjacency matrix is no longer a zero–one matrix,
- because the (i, j)th entry of this matrix equals the number of edges that are associated to $\{v_i, v_j\}$.

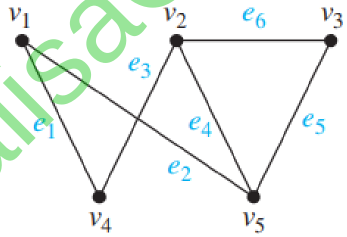
- All undirected graphs, including multigraphs and pseudographs, have symmetric adjacency matrices.
- **EXAMPLE 5** Use an adjacency matrix to represent the pseudograph shown in Figure.

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



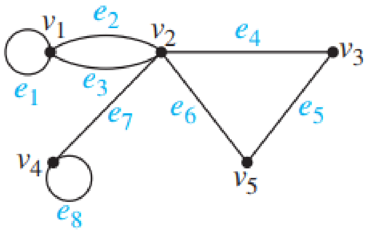
- The matrix for a directed graph $G = (V, E)$ has a 1 in its (i, j)th position if there is an edge from v_i to v_j
- $a_{ij} = 1$ if $\{v_i, v_j\}$ is an edge of G, 0 otherwise
- When a simple graph contains relatively few edges, that is, when it is **sparse**, it is usually preferable to use adjacency lists rather than an adjacency matrix to represent the graph.
- The adjacency matrix of a sparse graph is a **sparse matrix**, that is, a matrix with few nonzero entries.
- Now suppose that a simple graph is **dense**, 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 is usually preferable over using adjacency lists.

- **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 $\mathbf{M} = [m_{ij}]$, where
- $m_{ij} = 1$ when edge e_j is incident with v_i, 0 otherwise.
- **EXAMPLE 6** Represent the graph shown in Figure with an incidence matrix.



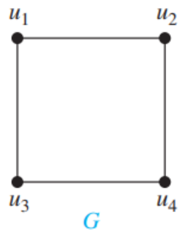
$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}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

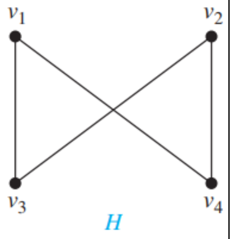
- **EXAMPLE 7** Represent the pseudograph shown in Figure using an incidence matrix.



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

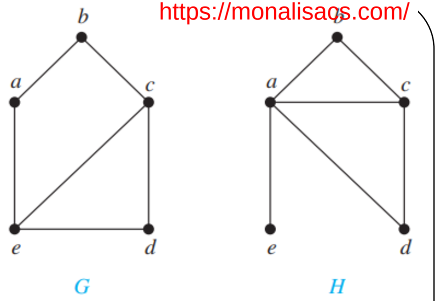
- **Isomorphism of Graphs**
- **DEFINITION 1** The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are *isomorphic* if there exists a one-to-one and onto function f from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if $f(a)$ and $f(b)$ are adjacent in G_2, for all a and b in V_1. Such a function f is called an *isomorphism*. Two simple graphs that are not isomorphic are called *nonisomorphic*.
- When two simple graphs are isomorphic, there is a one-to-one correspondence between vertices of the two graphs that preserves the adjacency relationship.
- Isomorphism of simple graphs is an equivalence relation.
- **EXAMPLE 8** Show that the graphs $G = (V, E)$ and $H = (W, F)$, displayed in Figure , are isomorphic.



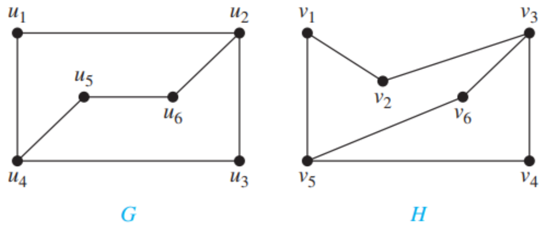


- *Solution:* The function f with $f(u_1) = v_1$, $f(u_2) = v_4$, $f(u_3) = v_3$, and $f(u_4) = v_2$ is a oneto-one correspondence between V and W.
- **Determining whether Two Simple Graphs are Isomorphic**
- It is often difficult to determine whether two simple graphs are isomorphic.
- There are $n!$ possible one-to-one correspondences between the vertex sets of two simple graphs with n vertices.
- A property preserved by isomorphism of graphs is called a **graph invariant**.

- Isomorphic simple graphs must have the same number of vertices ,Same number of edges, same degree sequence
- If ($G \cong H$) then $|V(G)|=|V(H)|, |E(G)|=|E(H)|$
- If the vertices $(v_1, v_2 \ldots . v_k)$ form a cycle of length k in graph G, then $(f(v_1), f(v_2) \ldots\ldots f(v_k))$ form a cycle of length k in graph H.
- G and H are isomorphic if $(\bar{G} \cong \bar{H})$
- G and H are isomorphic if there corresponding subgraphs are also isomorphic



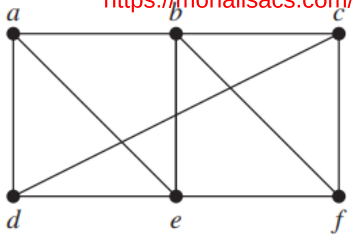
- **EXAMPLE 9** Show that the graphs in Figure are not isomorphic.
- *Solution:* Both *G* and *H* have five vertices and six edges.
- However, *H* has a vertex of degree one, namely, *e*, whereas *G* has no vertices of degree one. It follows that *G* and *H* are not isomorphic.



- **EXAMPLE 10** Determine whether the graphs shown in Figure are isomorphic.
- *Solution:* Both *G* and *H* have six vertices and seven edges. Both have four vertices of degree two and two vertices of degree three.
- $f(u_1) = v_6, f(u_2) = v_3, f(u_3) = v_4, f(u_4) = v_5, f(u_5) = v_1, f(u_6) = v_2.$
- Its isomorphic

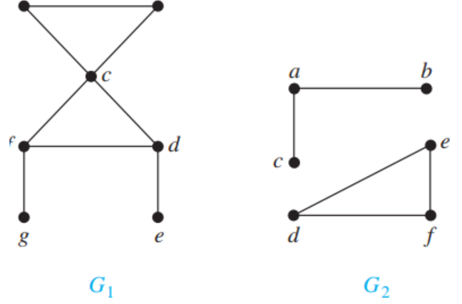
3.3 Connectivity

- **Paths**
- 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 $x_0, x_1, \ldots, x_n$ .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 $x_1, x_2, \ldots, x_{n-1}$ 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.
- In some books, the term **walk** is used instead of *path*, **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.
- When this terminology is used, the terminology **path** is often used for a trail with no repeated vertices.

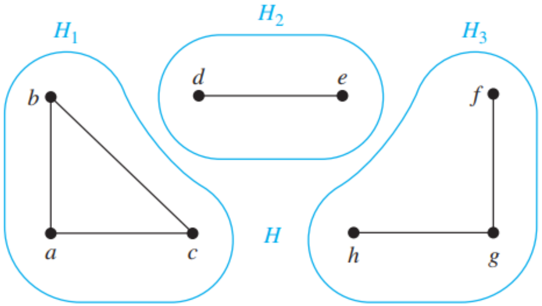


- **EXAMPLE 1** In the simple graph shown in Figure *a*, *d*, *c*, *f* , *e* is a simple path of length 4, because {*a*, *d*},{*d*, *c*}, {*c*, *f* }, and {*f*, *e*} are all edges.
- However, *d*, *e*, *c*, *a* is not a path, because {*e*, *c*} is not an edge.
- *b*, *c*, *f* , *e*, *b* is a circuit 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.
- **DEFINITION 2** Let n be a nonnegative integer and G a directed graph. A *path* of length n from u to v in G is a sequence of edges $e_1, e_2, \ldots, e_n$ of G such that e_1 is associated with (x_0, x_1), e_2 is associated with (x_1, x_2), and so on, with e_n associated with (x_{n-1}, x_n), where $x_0 = u$ and $x_n = v$. When there are no multiple edges in the directed graph, this path is denoted by its vertex sequence $x_0, x_1, x_2, \ldots, x_n$. A path of length greater than zero that begins and ends at the same vertex is called a *circuit* or *cycle*. A path or circuit is called *simple* if it does not contain the same edge more than once.
- **DEFINITION 3** 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.

- **EXAMPLE 2** The graph G_1 in Figure is connected, because for every pair of distinct vertices there is a path between them.
- However, the graph G_2 in Figure is not connected. For instance, there is no path in G_2 between vertices a and d.

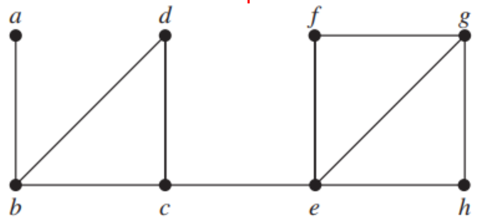


- **THEOREM 1** There is a simple path between every pair of distinct vertices of a connected undirected graph.
- CONNECTED COMPONENTS A connected component of a graph G is a maximal connected subgraph of G.A graph G that is not connected has two or more connected components that are disjoint and have G as their union.
- **EXAMPLE 3** What are the connected components of the graph H shown in Figure
- *Solution:* The graph H is the union of three disjoint connected subgraphs H_1, H_2, and H_3, shown in Figure.
- These three subgraphs are the connected components of H.



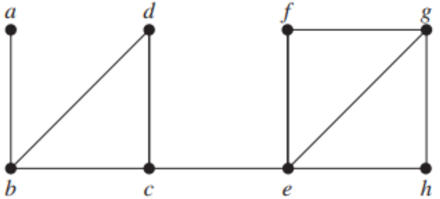
- Sometimes the removal from a graph of a vertex and all incident edges produces a subgraph with more connected components.
- Such vertices are called **cut vertices** (or **articulation points**).

- An edge whose removal produces a graph with more connected components than in the original graph is called a **cut edge** or **bridge**.
- EXAMPLE 4 Find the cut vertices and cut edges in the graph G_1 shown in Figure

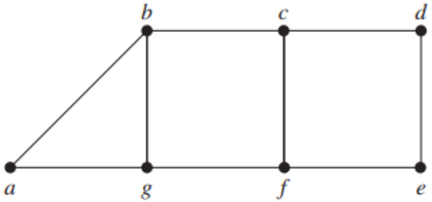


G_1

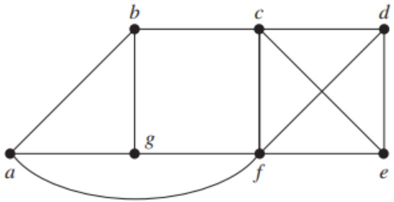
- *Solution:* The cut vertices of G_1 are b, c, and e.
- The removal of one of these vertices (and its adjacent edges) disconnects the graph.
- The cut edges are $\{a, b\}$ and $\{c, e\}$. Removing either one of these edges disconnects G_1.
- VERTEX CONNECTIVITY Not all graphs have cut vertices. The complete graph K_n, where $n \geq 3$, has no cut vertices.
- When you remove a vertex from K_n and all edges incident to it, the resulting subgraph is the complete graph K_{n-1}, a connected graph.
- Connected graphs without cut vertices are called **nonseparable graphs**
- A subset V ′ of the vertex set V of G = (V, E) is a **vertex cut**, or separating set, if G − V ′ is disconnected.
- The **vertex connectivity** of a noncomplete graph G, denoted by κ(G), as the minimum number of vertices in a vertex cut.
- A graph is *k*-**connected** (or *k*-**vertex-connected**), if $\kappa(G) \geq k$.



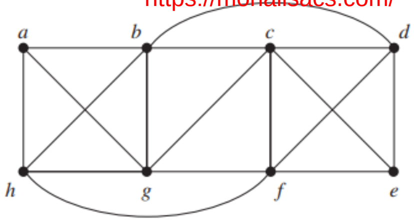
G_1



G_3

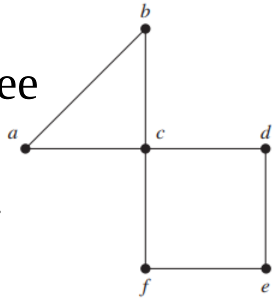


G_4



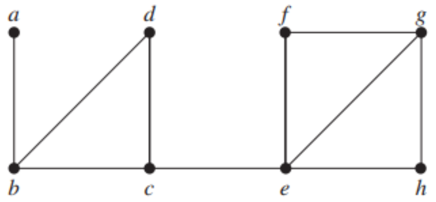
G_5

- A graph *G* is 1- connected if it is connected and not a graph containing a single vertex;
- A graph is 2-connected, or **biconnected**, if it is nonseparable and has at least three vertices.
- If *G* is a *k*-connected graph, then *G* is a *j*-connected graph for all *j* with $0 \leq j \leq k$.
- **EXAMPLE 5** Find the vertex connectivity for each of the graphs in Figure.
- *Solution:* Each of the five graphs are connected and has more than one vertex, so each of these graphs has positive vertex connectivity.
- Because G_1 is a connected graph with a cut vertex, we know that $\kappa(G_1) = 1$.
- Similarly, $\kappa(G_2) = 1$, because *c* is a cut vertex of G_2.
- G_3 has no cut vertices. but that $\{b, g\}$ is a vertex cut, $\kappa(G_3) = 2$.
- G_4 has a vertex cut of size two, $\{c, f\}$, but no cut vertices. $\kappa(G_4) = 2$.
- G_5 has no vertex cut of size two, but $\{b, c, f\}$ is a vertex cut of G_5, $\kappa(G_5) = 3$.

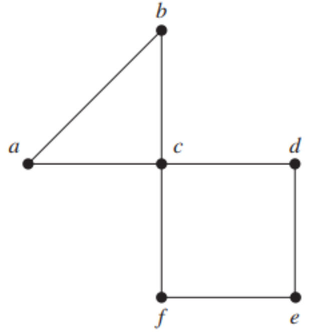


G_2

- EDGE CONNECTIVITY
- We can also measure the connectivity of a connected graph $G = (V, E)$ in terms of the minimum number of edges that we can remove to disconnect it.
- If a graph has a cut edge, then we need only remove it to disconnect G.
- If G does not have a cut edge, we look for the smallest set of edges that can be removed to disconnect it.
- A set of edges E' is called an **edge cut** of G if the subgraph $G - E'$ is disconnected.
- The **edge connectivity** of a graph G, denoted by $\lambda(G)$, is the minimum number of edges in an edge cut of G.
- $\lambda(G) = 0$ if G is not connected. We also specify that $\lambda(G) = 0$ if G is a graph consisting of a single vertex.
- It follows that if G is a graph with n vertices, then $0 \leq \lambda(G) \leq n - 1$.
- **EXAMPLE 6** Find the edge connectivity of each of the graphs in Figure
- *Solution:* Each of the five graphs in Figure is connected and has more than one vertex, so we know that all of them have positive edge connectivity.
- $G1$ has a cut edge, so $\lambda(G1) = 1$.

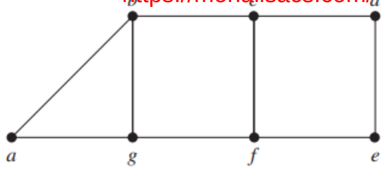


G_1

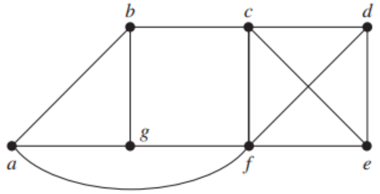


G_2

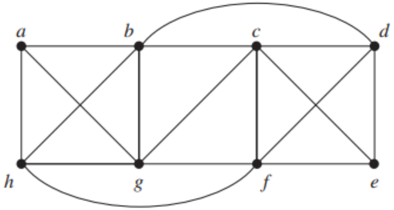
- The graph $G2$ has no cut edges, but the removal of the two edges $\{a,b\}$ and $\{a, c\}$ disconnects it. Hence, $\lambda(G2) = 2$.
- Similarly, $\lambda(G3) = 2$, because $G3$ has no cut edges, but the removal of the two edges $\{b, c\}$ and $\{f, g\}$ disconnects it.
- The removal of no two edges disconnects $G4$, but the removal of the three edges $\{b, c\}$, $\{a, f\}$, and $\{f, g\}$ disconnects it. Hence, $\lambda(G4) = 3$.
- Finally, $\lambda(G5) = 3$, because the removal of any two of its edges does not disconnect it, but the removal of $\{a, b\}$, $\{a, g\}$, and $\{a, h\}$ does.
- When $G = (V, E)$ is a noncomplete connected graph with at least three vertices, the minimum degree of a vertex of G is an upper bound for both the vertex connectivity of G and the edge connectivity of G.
- That is, $\kappa(G) \leq \min_{v \in V} \deg(v)$ and $\lambda(G) \leq \min_{v \in V} \deg(v)$.
- $\kappa(G) \leq \lambda(G) \leq \min_{v \in V} \deg(v)$.
- APPLICATIONS OF VERTEX AND EDGE CONNECTIVITY
- We can model a data network using vertices to represent routers and edges to represent links between them.
- The vertex connectivity of the resulting graph equals the minimum number of routers that disconnect the network when they are out of service.
- If fewer routers are down, data transmission between every pair of routers is still possible.
- The edge connectivity represents the minimum number of fiber optic links that can be down to disconnect the network.



G_3



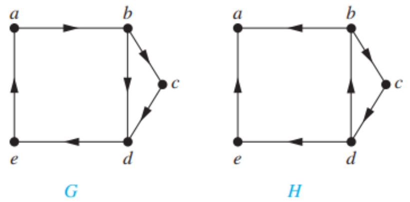
G_4



G_5

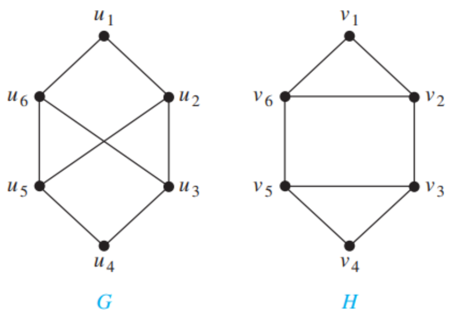
- If fewer links are down, it will still be possible for data to be transmitted between every pair of routers.
- We can model a highway network, using vertices to represent highway intersections and edges to represent sections of roads running between intersections.
- The vertex connectivity of the resulting graph represents the minimum number of intersections that can be closed at a particular time that makes it impossible to travel between every two intersections.
- If fewer intersections are closed, travel between every pair of intersections is still possible.
- The edge connectivity represents the minimum number of roads that can be closed to disconnect the highway network.
- If fewer highways are closed, it will still be possible to travel between any two intersections.
- It would be useful for the highway department to take this information into account when planning road repairs.

- **Connectedness in Directed Graphs**
- **DEFINITION 4** A directed graph is *strongly connected* if there is a path from *a* to *b* and from *b* to *a* whenever *a* and *b* are vertices in the graph.
- For a directed graph to be strongly connected there must be a sequence of directed edges from any vertex in the graph to any other vertex
- **DEFINITION 5** A directed graph is *weakly connected* if there is a path between every two vertices in the underlying undirected graph.
- That is, a directed graph is weakly connected if and only if there is always a path between two
- vertices when the directions of the edges are disregarded.
- Clearly, any strongly connected directed graph is also weakly connected.
- **EXAMPLE 7** Are the directed graphs *G* and *H* shown in Figure strongly connected? Are they weakly connected?

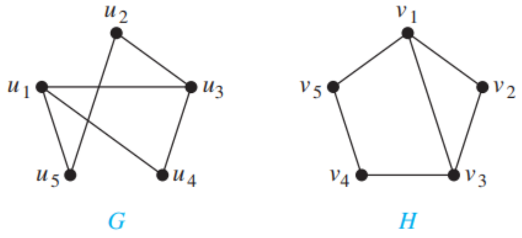


- *Solution:* *G* is strongly connected because there is a path between any two vertices in this directed graph. Hence, *G* is also weakly connected.
- The graph *H* is not strongly connected. There is no directed path from *a* to *b* in this graph. However, *H* is weakly connected, because there is a path between any two vertices in the underlying undirected graph of *H*.

- STRONG COMPONENTS OF A DIRECTED GRAPH
- The maximal strongly connected subgraphs, are called the strongly connected components or strong components of G.
- **EXAMPLE 8** The graph *H* in Figure has subgraph consisting of the vertices *b*, *c*, and *d* and edges *(b, c)*, *(c, d)*, and *(d, b)*.
- **Paths and Isomorphism**
- There are several ways that paths and circuits can help determine whether two graphs are isomorphic.
- 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.
- Paths can be used to construct mappings that may be isomorphisms.
- **EXAMPLE 9** Determine whether the graphs *G* and *H* shown in
- Figure are isomorphic.
- *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.
- Because the existence of a simple circuit of length three is an isomorphic invariant, *G* and *H* are not isomorphic.
- **EXAMPLE 10** Determine whether the graphs *G* and *H* shown in Figure 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.
- $f(u_1) = v_3$, $f(u_4) = v_2$, $f(u_3) = v_1$, $f(u_2) = v_5$, and $f(u_5) = v_4$.
- So, *G* and *H* are isomorphic,

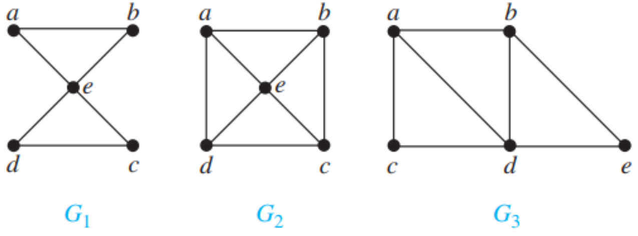
- **Theorem 2** A simple graph G with n vertices is necessarily connected if $|E(G)| > \frac{(n-1)(n-2)}{2}$ and necessarily disconnected if |(E)|< n-1
- **EXAMPLE 11** Which of the following simple graph is connected
- (a) A graph with 10 vertices and 30 edge
- 30 > (10-1)(10-2)/2
- 30 >36 not connected
- (b) A graph with 9 vertices and 28 edge
- 28> 8*7/2 =28 May or may not connected
- (c) A graph with 8 vertices and 22 edges
- 22> 7*6/2 =21 Connected
- **EXAMPLE 12** The minimum number of edge required to ensure connectivity with 7 vertices
- |E| > 6*5/2 =15
- Minimum 16 edge require.
- A simple graph with n vertices and k component has at least n-k edges

- A simple graph with n vertices & k components has at most $\frac{(n-k)(n-k+1)}{2}$ edge.
- n-k ≤ |E| ≤ $\frac{(n-k)(n-k+1)}{2}$

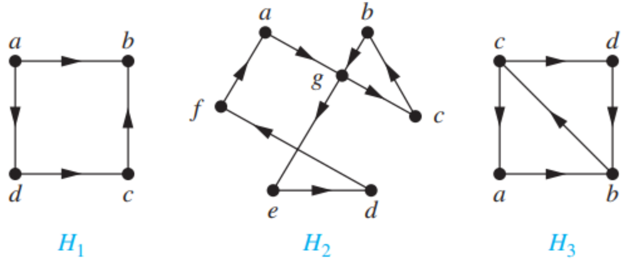
- **EXAMPLE 13** Minimum number of edges in a simple graph with 10 vertices & 3 component is 10-3=7
- Maximum number of edge possible in a simple graph with 10 vertices & 3 component ≤ $\frac{(10-3)(10-3+1)}{2}$ =28
- **EXAMPLE 14** Let G be a Simple graph with n vertices and k components if we delate an edge in G then number of component in G is k or k+1
- If that edge is a cut edge, then k+1 component .else k component .
- **EXAMPLE 15** Let G be a Simple graph with n vertices and k components if we delate a vertices in G then number of component in G should be between k & n-1.
- If the vertex is a cut vertex of star graph, then number of component become n-1.

3.4 Euler and Hamilton Paths

- **DEFINITION 1** 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*.
- It should cover each edge exactly once and each vertex at least once .
- **EXAMPLE 1** Which of the undirected graphs in Figure have an Euler circuit? Of those that do not, which have an Euler path?

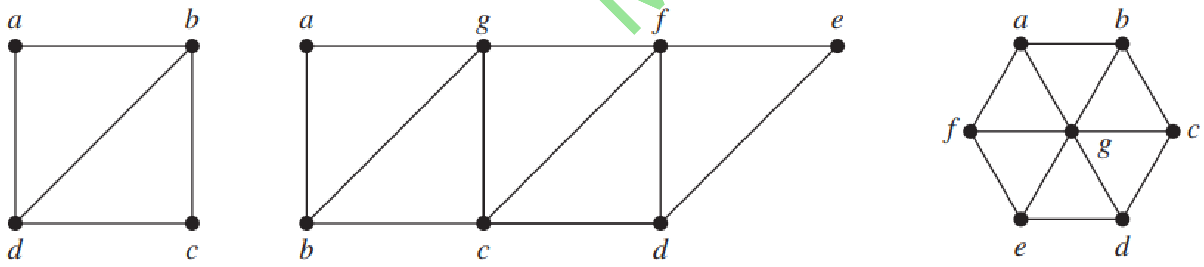


- *Solution:* The graph G_1 has an Euler circuit, for example, *a, e, c, d, e, b, a*.
- Neither of the graphs G_2 or G_3 has an Euler circuit.
- However, G_3 has an Euler path, namely, *a, c, d, e, b, d, a, b*.
- G_2 does not have an Euler path.
- **EXAMPLE 2** Which of the directed graphs in Figure have an Euler circuit? Of those that do not, which have an Euler path?



- *Solution:* The graph H_2 has an Euler circuit, for example, *a, g, c, b, g, e, d, f, a*.
- Neither H_1 nor H_3 has an Euler circuit.
- H_3 has an Euler path, namely,*c, a, b, c, d, b*, but H_1 does not

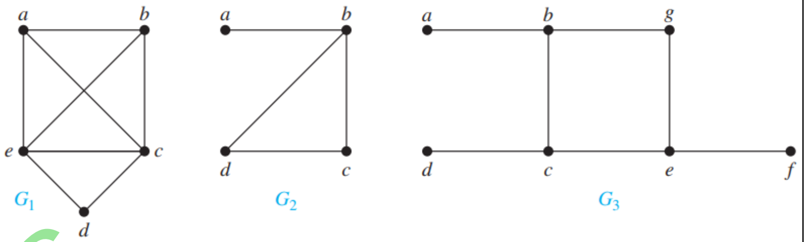
- **THEOREM 1** A connected multigraph with at least two vertices has an Euler circuit if and only if each of its vertices has even degree.
- **THEOREM 2** A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree.
- **EXAMPLE 3** Which graphs shown in Figure have an Euler path?
- *Solution:* G_1 contains exactly two vertices of odd degree, namely, b and d.
- Hence, it has an Euler path that must have b and d as its endpoints.
- One such Euler path is d, a, b, c, d, b.
- Similarly, G_2 has exactly two vertices of odd degree, namely, b and d.
- So it has an Euler path that must have b and d as endpoints.
- One such Euler path is $b, a, g, f, e, d, c, g, b, c, f, d$.
- G_3 has no Euler path because it has six vertices of odd degree.



G_1 G_2 G_3

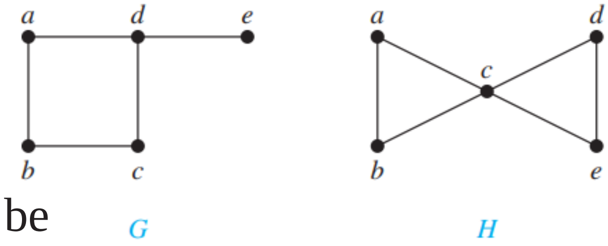
- APPLICATIONS OF EULER PATHS AND CIRCUITS
- Many applications ask for a path or circuit that traverses each street in a neighborhood, each road in a transportation network, each connection in a utility grid, or each link in a communications network exactly once.
- Finding an Euler path or circuit in the appropriate graph model can solve such problems.
- For example, if a postman can find an Euler path in the graph that represents the streets the postman needs to cover, this path produces a route that traverses each street of the route exactly once.
- If no Euler path exists, some streets will have to be traversed more than once.
- Euler circuits and paths are applied is in the layout of circuits, in network multicasting, and in molecular biology, where Euler paths are used in the sequencing of DNA.
- **Hamilton Paths and Circuits**
- **DEFINITION 2** 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 $x_0, x_1, \ldots, x_{n-1}, x_n$ in the graph $G = (V, E)$ is a Hamilton path if $V = \{x_0, x_1, \ldots, x_{n-1}, x_n\}$ and $x_i \neq x_j$ for $0 \leq i < j \leq n$, and the simple circuit $x_0, x_1, \ldots, x_{n-1}, x_n, x_0$ is a Hamilton circuit if $x_0, x_1, \ldots, x_{n-1}, x_n$ is a Hamilton path

- **EXAMPLE 4** Which of the simple graphs in Figure have a Hamilton circuit or, if not, a Hamilton path?



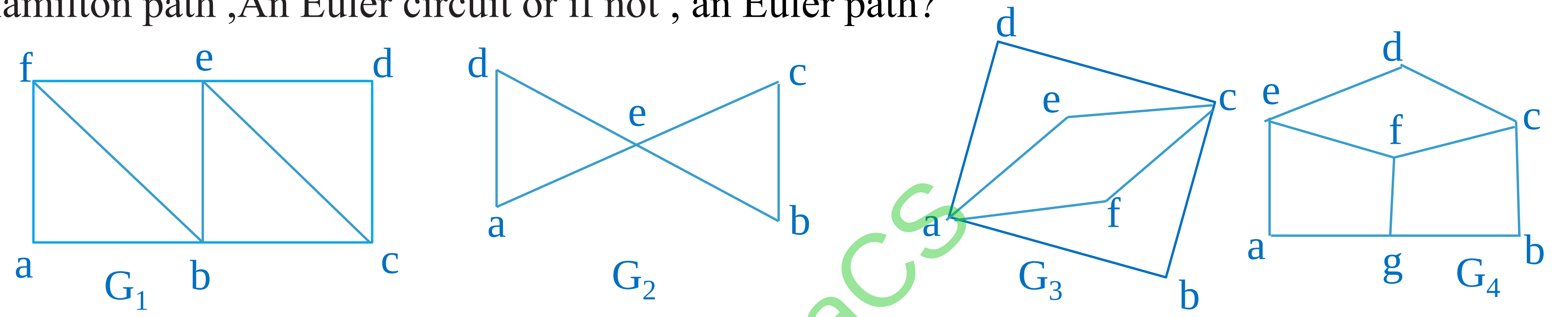
- *Solution:* G_1 has a Hamilton circuit: a, b, c, d, e, a.
- There is no Hamilton circuit in G_2
- But G_2 does have a Hamilton path, namely, a, b, c, d.
- G_3 has neither a Hamilton circuit nor a Hamilton path, because any path containing all vertices must contain one of the edges $\{a, b\}$, $\{e, f\}$, and $\{c, d\}$ more than once.
- Conditions for the existence of Hamilton circuits
- A graph with a vertex of degree one cannot have a Hamilton circuit, because in a Hamilton circuit, each vertex is incident with two edges in the circuit.
- Moreover, if a vertex in the graph has degree two, then both edges that are incident with this vertex must be part of any Hamilton circuit.
- When a Hamilton circuit is being constructed and this circuit has passed through a vertex, then all remaining edges incident with this vertex, other than the two used in the circuit, can be removed from consideration.
- A Hamilton circuit cannot contain a smaller circuit within it.

- **EXAMPLE 5** Show that neither graph displayed in Figure has a Hamilton circuit.



- *Solution:* There is no Hamilton circuit in *G* because *G* has a vertex of degree one, namely, *e*.
- Now consider *H*. Because the degrees of the vertices *a*, *b*, *d*, and *e* are all two, every edge incident with these vertices must be part of any Hamilton circuit.
- No Hamilton circuit can exist in *H*, for any Hamilton circuit would have to contain four edges incident with *c*, which is impossible
- **Applications of Hamilton Circuits**
- Many applications ask for a path or circuit that visits each road intersection in a city, each place pipelines intersect in a utility grid, or each node in a communications network exactly once.
- Finding a Hamilton path or circuit in the appropriate graph model can solve such problems.
- The famous **traveling salesperson problem** or **TSP** asks for the shortest route a traveling salesperson should take to visit a set of cities.

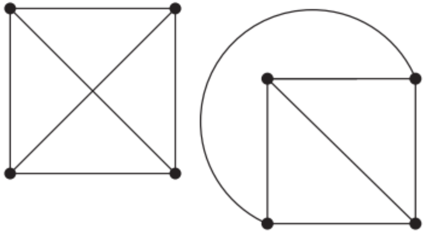
EXAMPLE 6 Which of the simple graphs in Figure have a Hamilton circuit or, if not, a Hamilton path ,An Euler circuit or if not , an Euler path?



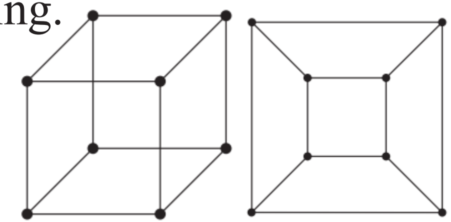
Graph	Euler Path	Euler Circuit	Hamilton path	Hamilton Circuit
G_1	Yes	No	Yes	Yes
G_2	Yes	Yes	Yes	No
G_3	Yes	Yes	No	No
G_4	No	No	Yes	No

3.5 Planar Graphs

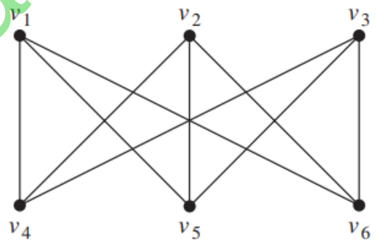
- **DEFINITION 1** A graph is called *planar* if it can be drawn in the plane without any edges crossing .Such a drawing is called a *planar representation* of the graph.



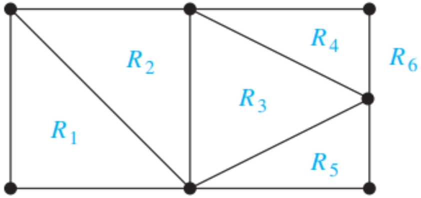
- **EXAMPLE 1** Is K_4 planar?
- *Solution:* K_4 is planar because it can be drawn without crossings.
- **EXAMPLE 2** Is Q_3 planar?
- *Solution:* Q_3 is planar, because it can be drawn without any edges crossing.



- **EXAMPLE 3** Is $K_{3,3}$, shown in Figure, planar?



- *Solution:* Nonplanar
- **Euler's Formula**
- A planar representation of a graph splits the plane into **regions**, including an unbounded region.
- For instance, the planar representation of the graph shown in Figure splits the plane into six regions.

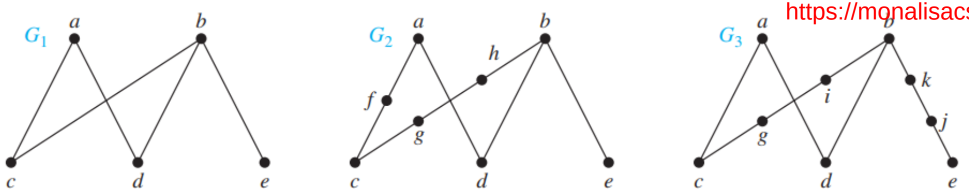


- **THEOREM 1 EULER'S FORMULA** Let G be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then $r = e - v + 2$.

- **EXAMPLE 4** Suppose that a connected planar simple graph has 20 vertices, each of degree 3. Into how many regions does a representation of this planar graph split the plane?
- *Solution:* $v = 20$. $3v = 3 * 20 = 60$, $2e = 60$, or $e = 30$.
- The number of regions is
- $r = e - v + 2 = 30 - 20 + 2 = 12$.
- **COROLLARY 1** If G is a connected planar simple graph with e edges and v vertices, where $v \geq 3$, then $e \leq 3v - 6$
- ***Proof:*** $2e = \Sigma \deg(R) \geq 3r$
- $(2/3)e \geq r$.
- Using $r = e - v + 2$ (Euler's formula),
- $e - v + 2 \leq (2/3)e$.
- $e/3 \leq v - 2$.
- This shows that $e \leq 3v - 6$.
- **EXAMPLE 5** Show that K_5 is nonplanar using Corollary 1.
- *Solution:* The graph K_5 has five vertices and 10 edges.
- $e \leq 3v - 6$ is not satisfied for this graph
- $e = 10$ and $3v - 6 = 9$. Therefore, K_5 is not planar.

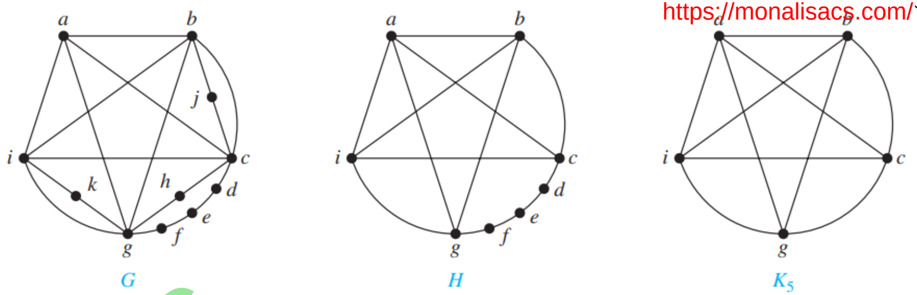
- **COROLLARY 2** If a connected planar simple graph has e edges and v vertices with $v \geq 3$ and no circuits of length three, then $e \leq 2v - 4$
- **EXAMPLE 6** Use Corollary 2 to show that $K_{3,3}$ is nonplanar.
- *Solution:* $K_{3,3}$ has six vertices and nine edges. Because $e = 9$ and $2v - 4 = 8$,
- Corollary 2 shows that $K_{3,3}$ is nonplanar.
- **Kuratowski's Theorem**
- We have seen that $K_{3,3}$ and K_5 are not planar.
- Clearly, a graph is not planar if it contains either of these two graphs as a subgraph.
- All nonplanar graphs must contain a subgraph that can be obtained from $K_{3,3}$ or K_5 using certain permitted operations.
- If a graph is planar, so will be any graph obtained by removing an edge $\{u, v\}$ and adding a new vertex w together with edges $\{u, w\}$ and $\{w, v\}$. Such an operation is called an **elementary subdivision**.
- The graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are called **homeomorphic** if they can be obtained from the same graph by a sequence of elementary subdivisions.

- **EXAMPLE 7** Show that the graphs G_1, G_2, and G_3 displayed in Figure are all homeomorphic.



- *Solution:* These three graphs are homeomorphic because all three can be obtained from G_1 by elementary subdivisions.
- G_1 can be obtained from itself by an empty sequence of elementary subdivisions.
- To obtain G_2 from G_1 we can use this sequence of elementary subdivisions:
- (*i*) remove the edge $\{a, c\}$, add the vertex f , and add the edges $\{a, f\}$ and $\{f, c\}$;
- (*ii*) remove the edge $\{b, c\}$, add the vertex g, and add the edges $\{b, g\}$ and $\{g, c\}$; and
- (*iii*) remove the edge $\{b, g\}$, add the vertex h, and add the edges $\{g, h\}$ and $\{b, h\}$.
- Same way G_3 can obtain from G_1 by elementary subdivisions
- **THEOREM 2** A graph is nonplanar if and only if it contains a subgraph homeomorphic to $K_{3,3}$ or K_5.
- **EXAMPLE 8** Determine whether the graph G shown in Figure is planar.
- *Solution:* G has a subgraph H homeomorphic to K_5.

- *H* is obtained by deleting *h, j*, and *k* and all edges incident with these vertices.
- *H* is homeomorphic to K_5 because it can be obtained from K_5 by a sequence of elementary subdivisions, adding the vertices *d*, *e*, and *f* .

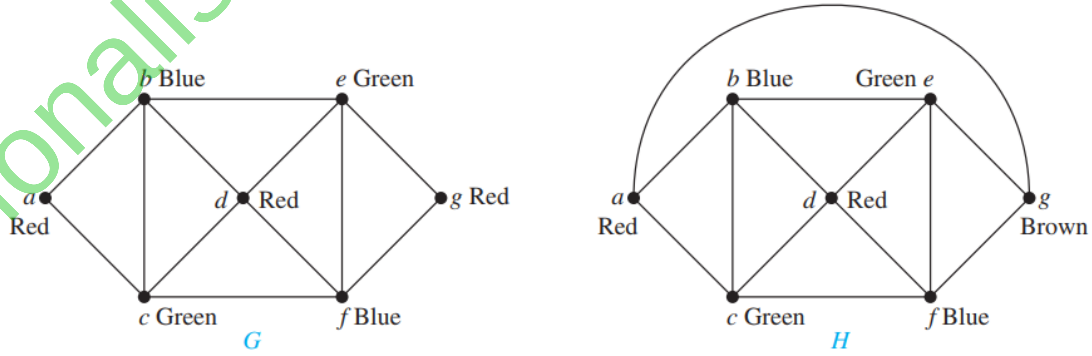


- Hence, *G* is nonplanar.
- **EXAMPLE 9** Let G is a connected planar graph with 25 vertices and 60 edges then number of bounded region in G is ____ ?
- *Solution:* $|V|+|R|=|E|+2$
- $|R|=60+2-25=37$
- 36 Bounded region ,1 unbounded region .
- **EXAMPLE 10** If G is a disconnected planar graph with 10 vertices ,15 edges & 3 component then R=?
- *Solution:* |V|=10 ,|E|=15 ,|k|=3 ,R=?
- 10+ R=15+4 ,R=9

- **EXAMPLE 11** If G is a connected planar graph with 35 region ,degree of each region 6 ,V=?
- *Solution:* 35 *6=2*E ,E=105
- |V|=|E|-|R|+2=105-35+2=72
- **EXAMPLE 12** In a connected planar graph |V|=20 ,|E|=30 ,k=?
- *Solution:* |V|+|R|=|E|+2 => |R|=30+2-20 =12
- k*|R|=2*|E| => k=2*30/12 =5
- **EXAMPLE 13** Maximum number of edges possible in a simple connected planar graph with 8 vertices is ?
- *Solution:* $|E| \leq 3|V| - 6$
- $|E| \leq 3*8-6=18$
- **EXAMPLE 14** Minimum number of vertices necessary in a simple connected planar graph with 14 edges is ?
- *Solution:* $14 \leq 3*|V|-6 \Rightarrow 20 \leq 3*|V|$
- |V|=7
- **EXAMPLE 15** |V|=10 , No tringle ,number of edges in G can’t exceed ?
- *Solution:* $|E| \leq 2*10-4=16$

3.6 Graph Coloring

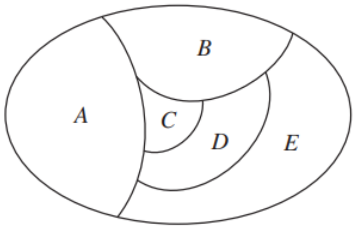
- **DEFINITION 1** A *coloring* of a simple graph is the assignment of a color to each vertex of the graph so that no two adjacent vertices are assigned the same color.
- **DEFINITION 2** The *chromatic number* of a graph is the least number of colors needed for a coloring of this graph. The chromatic number of a graph G is denoted by $\chi(G)$.
- (Here χ is the Greek letter *chi*.)
- **THEOREM 1 THE FOUR-COLOR THEOREM** The chromatic number of a planar graph is no greater than four.
- **EXAMPLE 1** What are the chromatic numbers of the graphs G and H shown in Figure

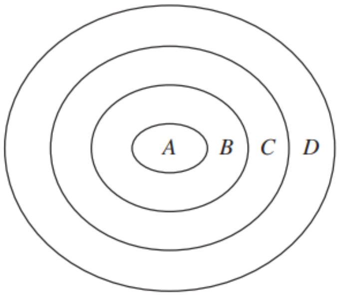


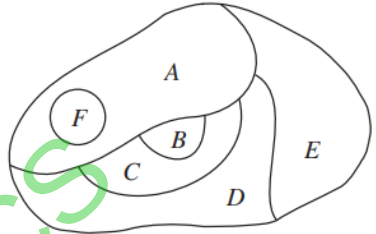
- *Solution:* The chromatic number of G is at least three, because the vertices a, b, and c must be assigned different colors.
- H has a chromatic number equal to 4.

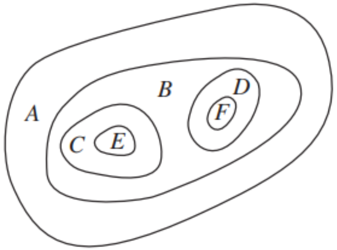
- **EXAMPLE 2** What is the chromatic number of K_n?
- *Solution:* A coloring of K_n can be constructed using n colors by assigning a different color to each vertex.
- **EXAMPLE 3** What is the chromatic number of the complete bipartite graph $K_{m,n}$, where m and n are positive integers?
- *Solution:* $\chi(K_{m,n}) = 2$. This means that we can color the set of m vertices with one color and the set of n vertices with a second color.
- **EXAMPLE 4** What is the chromatic number of the graph C_n, where $n \geq 3$.
- $\chi(C_n) = 2$ if n is an even positive integer with $n \geq 4$ and $\chi(C_n) = 3$ if n is an odd positive integer with $n \geq 3$.
- **Applications of Graph Colorings**
- Graph coloring has a variety of applications to problems involving scheduling and assignments.
- **Scheduling Final Exams** How can the final exams at a university be scheduled so that no student has two exams at the same time?
- **Frequency Assignments** Television channels are assigned to stations in North America so that no two stations within 150 miles can operate on the same channel.

- **EXAMPLE 5** Find the number of colors needed to color the map so that no two adjacent regions have the same color.

1.


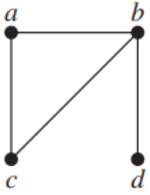
2.


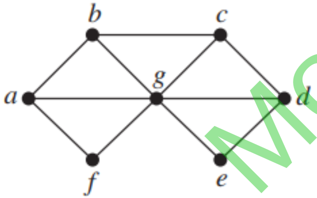
3.


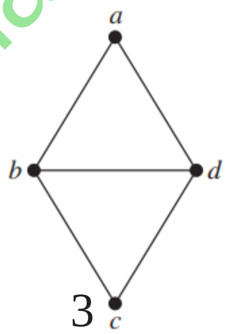
4.


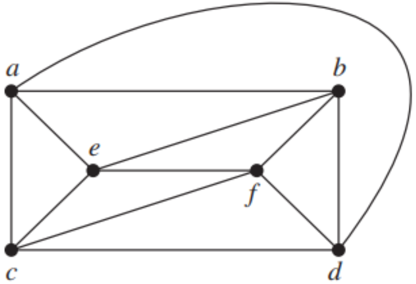
- 4 2 3 2

- **EXAMPLE 6** find the chromatic number of the given graph.

5.


6.


7.


8.


- 3 3 3 3