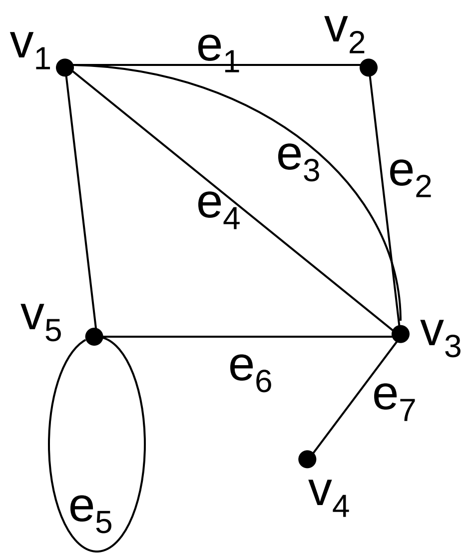
Graph Theory and Combinatorics

Tohida Rehman
Assistant professor
Department of Information Technology

Definitions

- A graph $G=(V,E,\Phi)$ consists of
  - A set of objects $V=\{v_1, v_2, v_3,\ldots\}$ called vertices
  - Another set $E=\{e_1, e_2, e_3,\ldots\}$, whose elements are called edges.
    - An edge is associated with a unordered pair of vertices, $e_k=(v_i, v_j)$, undirected edge
      - Φ is edge map: $\Phi(e_k)=(v_i, v_j)$.
    - Vertices v_i, v_j are the end vertices of the edge e_k.
    - In other way, e_k is incident on v_i and v_j.

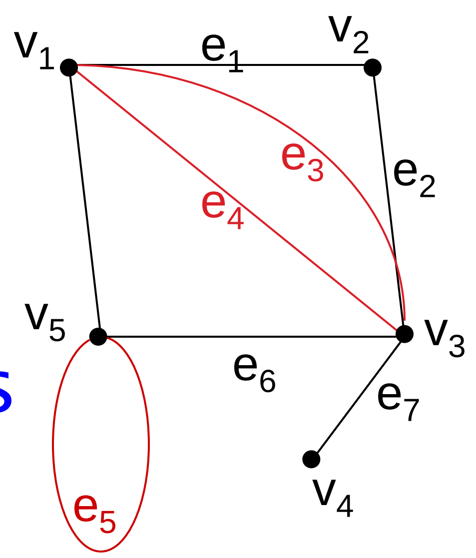
It is obvious that $E \subseteq V \times V$



Vertices are $V=\{v_1, v_2, v_3, v_4, v_5\}$ and edge are $E\ \{e_1,e_2,e_3,\ldots,e_7\}$

Definitions

- Self-loop (loop): an edge $e=(v_i,v_i)$
  - $\Phi(e_5) = (v_5, v_5)$.
- Parallel edges: two edges $e_q = (v_r,v_s)$ and $e_p = (v_r,v_s)$ $[p \neq q]$
- Simple graph: no parallel edges & no self loop
  - Non-simple graph is *multigraph [possibly no self-loop]*
- Directed edge: edge is defined as ordered pair of vertices
  - Initial vertex and terminal vertex



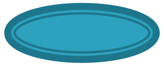
In the diagram edge is connection between two vertices:
may be joined by straight line or curved line
may be short or long

Definitions

- Respect to source vertex edge is outgoing edge and respect to terminal vertex edge is incoming vertex
- A graph is called undirected if every edge is undirected
- A graph is said to be directed if every edge is directed
- Otherwise, graph is mixed graph
  - A city road map, one-way or two-way direction of traffic

Definitions

- *Order* of a graph is the number of vertices
- *Size* is the number of edges
  - A graph with single vertex is called trivial graph
  - A non-trivial graph, order is at least two



- Trivial Graph

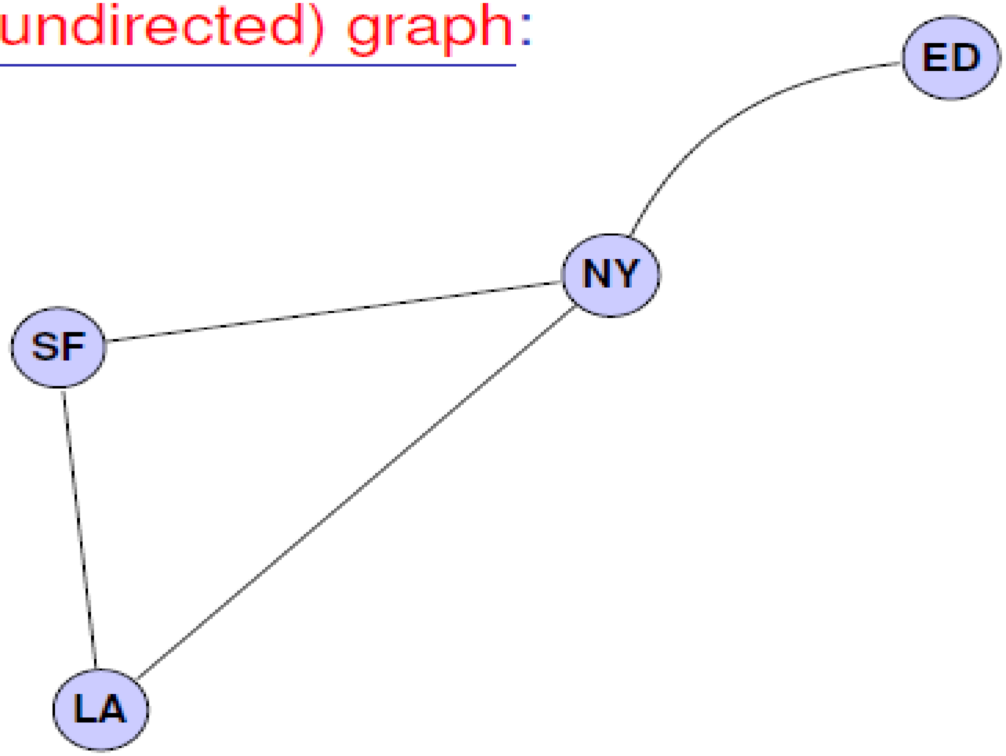
What is a Graph?

- Informally, a **graph** consists of a non-empty set of **vertices** (or **nodes**),and a set E of **edges** that connect (pairs of) nodes.
- But different types of graphs (undirected, directed, simple, multigraph,. . .) have different formal definitions, depending on what kinds of edgesare allowed.

- Before formalizing, let's see some examples....

Simple Undirected Graph

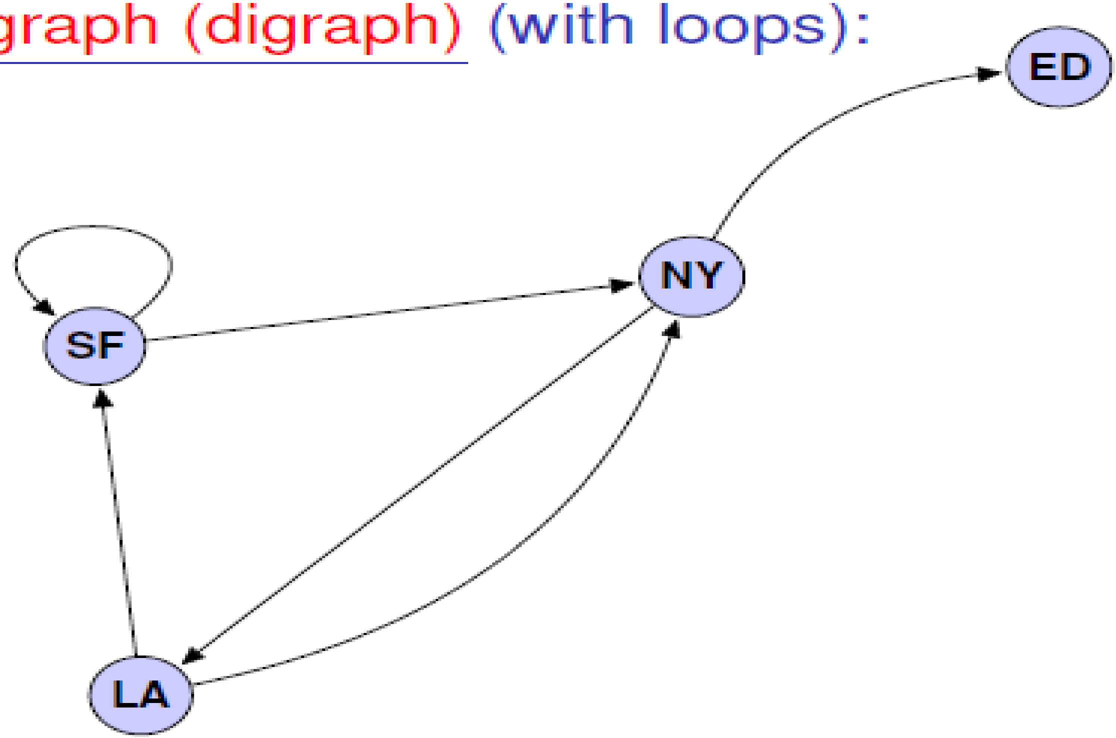
A (simple undirected) graph:



Only undirected edges; at most one edge between any pair of distinct nodes; and no loops (edges between a node and itself).

A directed graph (digraph) (with loops)

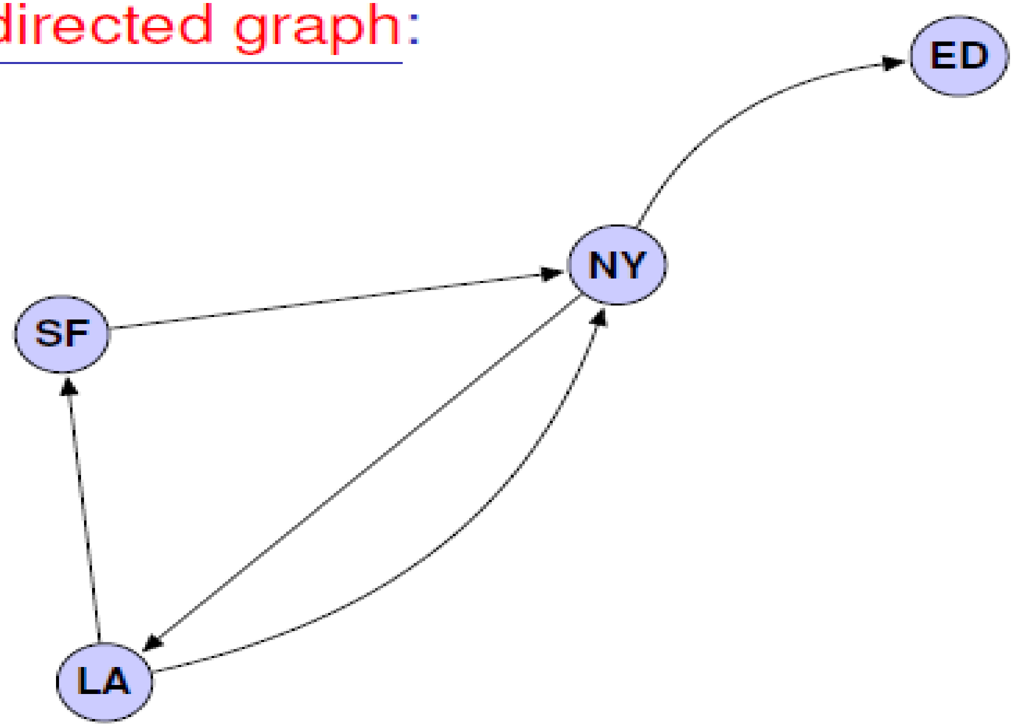
A directed graph (digraph) (with loops):



Only directed edges; at most one directed edge from any node to any node; and loops are allowed.

Simple Directed graph

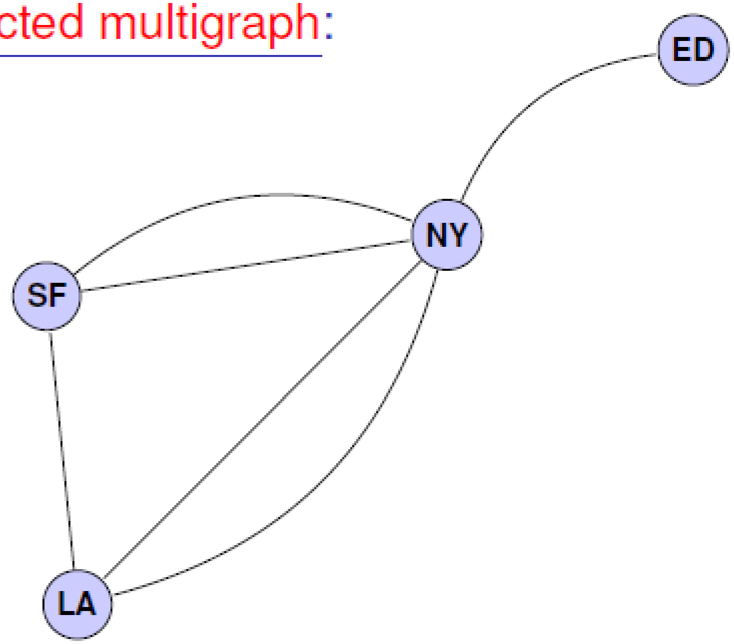
A simple directed graph:



Only directed edges; at most one directed edge from any node to any other node; and no loops allowed.

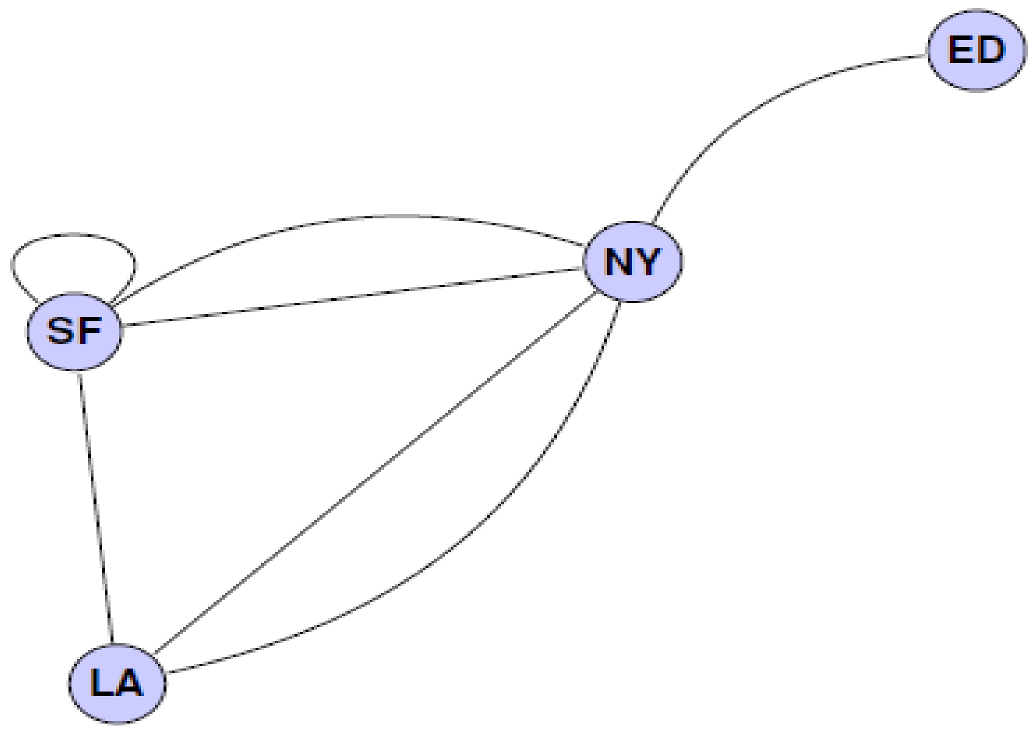
Undirected Multigraph

An undirected multigraph:



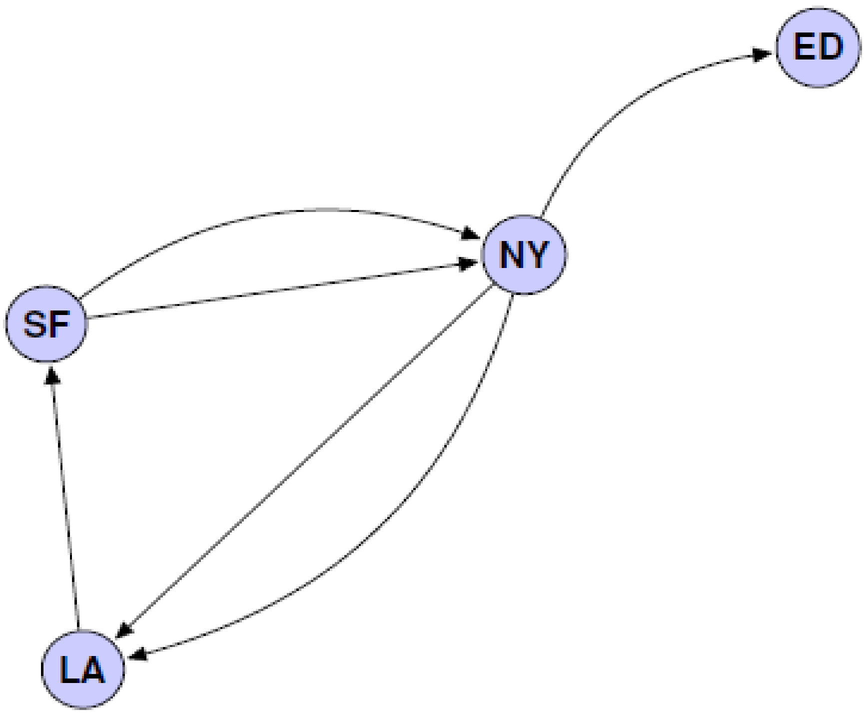
Only undirected edges; may contain multiple edges between a pair of nodes; but no loops.

Undirected Pseudograph



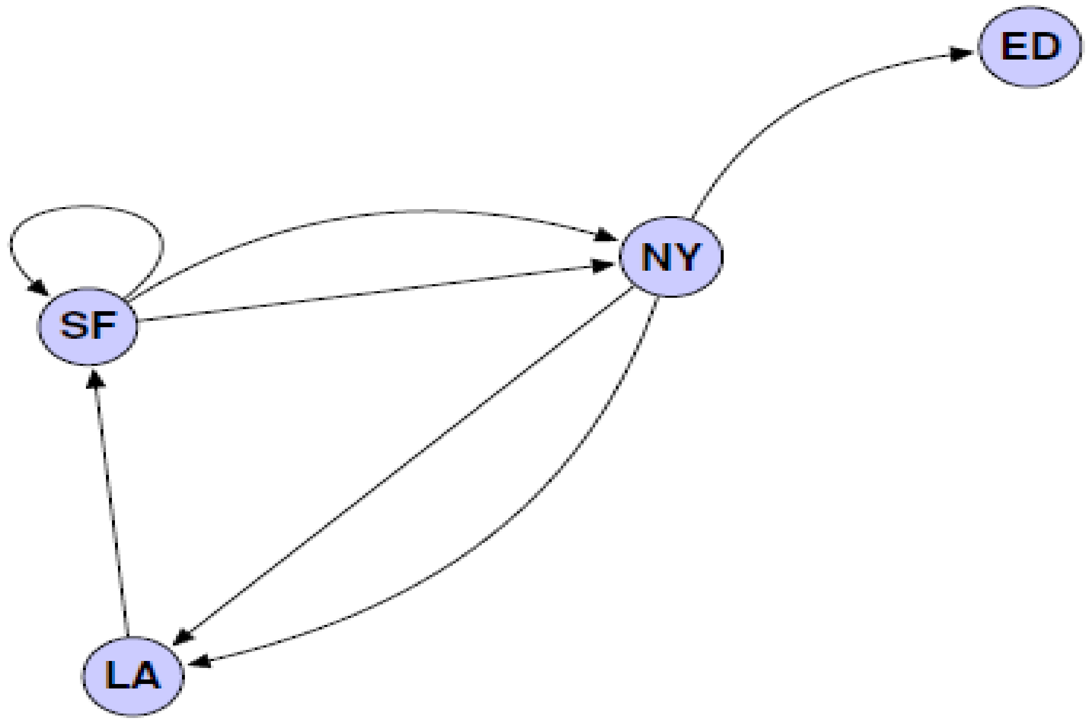
Only undirected edges; may contain multiple edges between a pair of nodes; and may contain loops (even multiple loops on the same node).

Directed Multigraph



Only directed edges; may contain multiple edges from one node to another; but no loops allowed.
Warning: this differs slightly from the Rosen book terminology. The book's notion of "directed multigraph" would allow loops.

Directed Pseudograph



Only directed edges; may contain multiple edges from one node to another; and may contain loops (even multiple loops on one node).

Graph Terminology

	Type	Edges	Multi-Edges?	Loops?
1.	(simple undirected) graph	Undirected	No	No
2.	(undirected) multigraph	Undirected	Yes	No
3.	(undirected) pseudograph	Undirected	Yes	Yes
4.	directed graph	Directed	No	Yes
5.	simple directed graph	Directed	No	No
6.	directed multigraph	Directed	Yes	No[1]
7.	directed pseudograph	Directed	Yes	Yes
8.	mixed graph	Both	Yes	Yes

We will focus on the two most standard types:
(1.) graphs (simple undirected), and
(4.) directed graphs (also known as digraphs).

[1] differs from book. Better to use possibly

Different applications

- Time table problem
- Class room assignment problem
- Map coloring problem
- Graph theory has its applications in diverse fields of engineering −
- **Electrical Engineering** − in designing circuit connections
- Com**puter Science** − Graph theory is used for the study of algorithms. For example,
  - Kruskal's Algorithm, Prim's Algorithm, Dijkstra's Algorithm
- **Computer Network** − The relationships among interconnected computers in the network follows the principles of graph theory.
- **Science** − The molecular structure and chemical structure of a substance, etc.
- **General** − Routes between the cities can be represented using graphs

Adjacency

- Vertices u and v are adjacent or neighbors, if they are endvertices of an edge $e \in E$, i.e., $e=(u,v)$

- Two distinct edges e and f are adjacent, if they have a common endvertex, i.e., $|\Phi(e)\cap\Phi(f)| = 1$

- Two distinct edges e and f are parallel, if $\Phi(e) = \Phi(f)$

Adjacency

- Incidence & degree
  - When a vertex ***v*** is an end vertex of the edge ***e***, then ***v*** & ***e*** are said to be *incident* with (on or to) each other.
  - Number of edges incident on a vertex is called as the degree of the vertex.
    - For self-loop counted twice.
- A simple graph with n vertices and degree of each vertex is n-1 is known as complete graph.

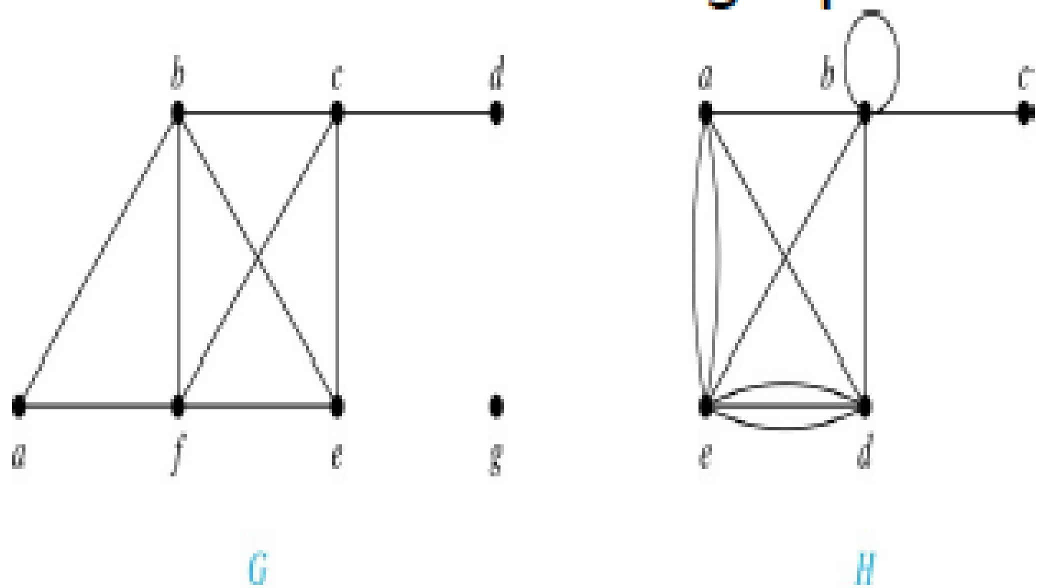
Degree and neighborhood of a vertex

- The degree of a vertex v in a undirected graph is the number of edges incident with it. The degree of the vertex v is denoted by deg(v).

- The neighborhood (neighbor set) of a vertex v in a undirected graph, denoted N(v) is the set of vertices adjacent to v.

Degree and neighborhood of a vertex

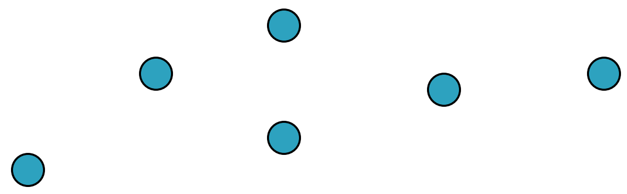
Example: What are the degrees and neighborhoods of the vertices in the graphs *G* and *H*?



Solution: deg(a) = 2, deg(b) = 4, deg(d) = 1,
$N(a) = \{b, f\}$, $N(b) = \{a, c, e, f\}$, $N(d) = \{c\}$.

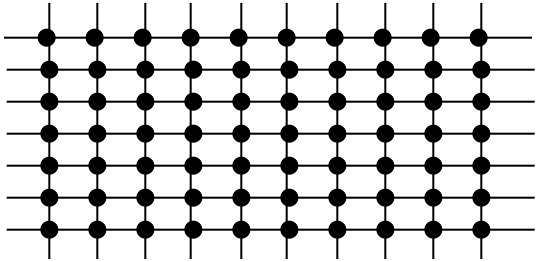
Isolated vertex, pendant vertex, null graph

- A vertex with degree 0 [no edge is incident on it], is called an isolated vertex
- A vertex with degree 1, is called pendant vertex and the corresponding edge is called pendant edge.
- A graph without any edges is called null graph.
  - Assumed some vertices are present in the graph
  - All the vertices are isolated



Finite & infinite graph

- A graph with finite number of vertices and finite number of edges is called a finite graph.
- Otherwise, it is an infinite graph.



Infinite graph

Finite & infinite graph

- Prove that an infinite graph with a finite number of edges must have an infinite number of isolated vertices.
- Show that an infinite graph with finite vertices will have at least one pair of vertices joined by an infinite number of parallel lines [one vertex in case on infinite parallel self-loops].
  - For multigraph at least one pair of vertices joined by an infinite number of parallel lines.

Handshaking Theorem

THEOREM 1 (Handshaking Lemma): If G=(V,E) is a undirected graph with **m** edges, then:

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

Proof:

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

Number of odd degree vertex

- Sum of the degree of the vertices is 2.edges. [$\Sigma d(v) = 2e$]
- int d[max]=0, count=0;
- for each edge $e_i=(v_r, v_s)$ do

  {
  - d[v_{r},]++;
  - d[v_{s},]++;
  - count=count+2;

  }

$$\boxed{\sum_{i=1}^{n} d(v_i) \equiv 0 \pmod 2}$$

Number of odd degree vertex

- Number of vertices of odd degree is always even.
  - We have $\sum_{i=1}^{n} d(v_i) = 2e$

$$\sum_{i=1}^{n} d(v_i) = \sum_{odd} d(v_j) + \sum_{even} d(v_k)$$

$$even = \sum_{odd} d(v_j) + even \Rightarrow \sum_{odd} d(v_j) = even$$

Handshaking Theorem:Examples

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

Solution: the sum of the degrees of the vertices is $6 \cdot 10 = 60$. The handshaking theorem says $2m = 60$.

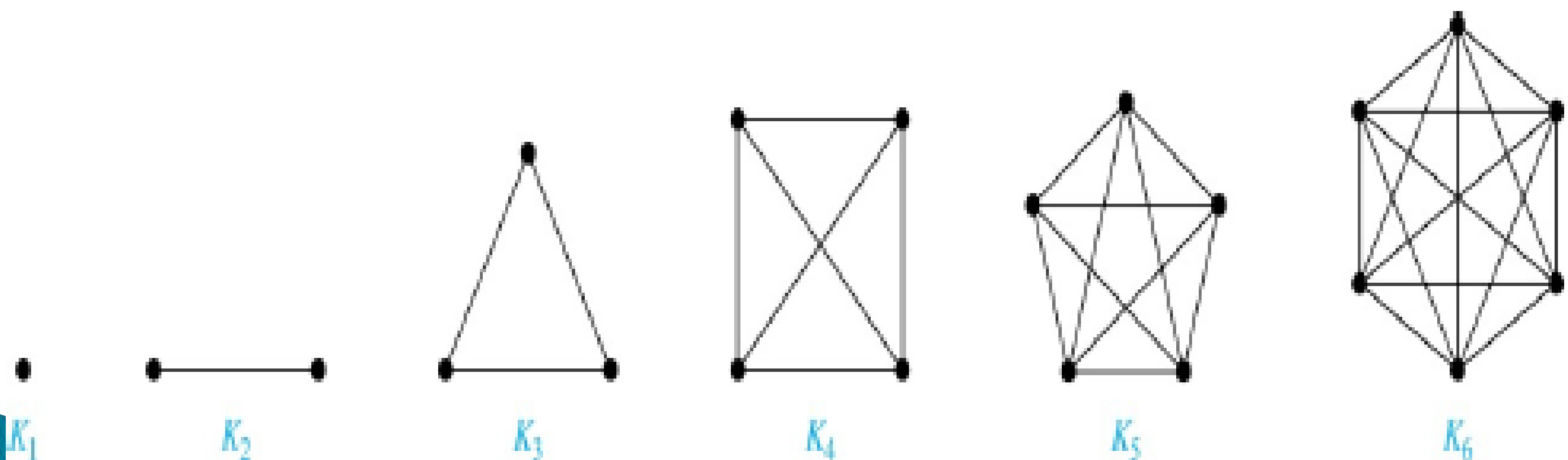
So the number of edges is $m = 30$.

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

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

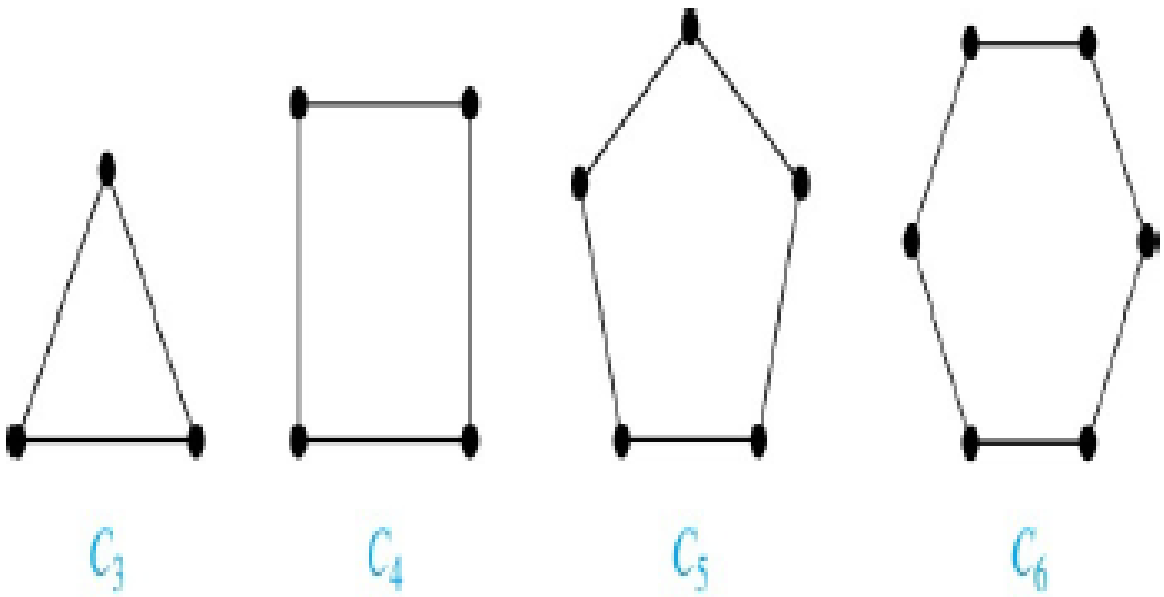
Special Types of Graphs: Complete Graphs

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



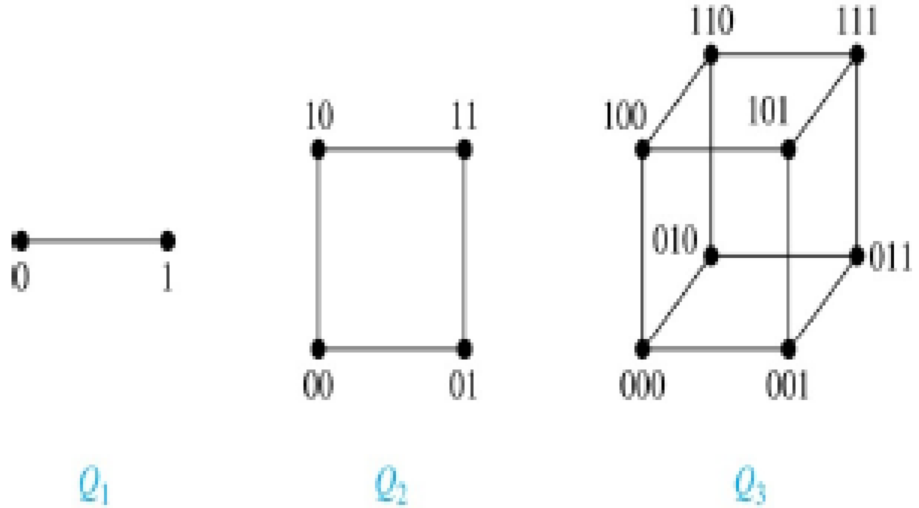
Special Types of Graphs: Cycle

A *cycle* C_n for $n \geq 3$ consists of n vertices $v1$, $v2$,⋯ , vn, and edges {$v1$, $v2$}, {$v2$, $v3$} ,⋯ , {vn-1, vn}, {vn, $v1$}.



Special Types of Simple Graphs: n-Cubes

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



Walks, paths & circuits

- A walk in a graph $G=(V,E,\Phi)$ is a finite alternating sequence $(v_0, e_1, v_1, e_2, v_2,\ldots e_k, v_k)$ of vertices and edges that begins and ends with a vertex. For each i, end vertices of e_i are v_{i-1} and v_i.
  - A vertex may appear more than once.
  - v_0, is the initial vertex of the walk
  - v_k, is the final vertex of the walk
  - k is the length of the walk
  - If both initial and final vertices are same, then it is a closed walk;
  - otherwise, open walk.
- A walk in which no edge appears more than once is called trail.

Walks, paths & circuits

- An open walk is a path, if no vertex is repeated more than once.
  - Respect to the path, degree of end vertices is one and all other vertices having degree two.
- A closed walk in which no vertex (except the initial and final vertex) appears more than once is called a circuit.
  - A circuit is also known as cycle or polygon, etc.

Walks, paths & circuits

- If a graph G contains a u-v walks of length l, then G contains a u-v path of length at most l.
  - If the walk is a path, nothing to prove.
  - If some vertices occurs more than once
    - Ignore the loop part and repeat the process, finally we will have a path with length less than l.

Paths (in undirected graphs)

Definition: For an undirected graph $G = (V, E)$, an integer $n \geq 0$, and vertices $u, v \in V$, a path (or walk) of length n from u to v in G is a sequence:

$$x_0, e_1, x_1, e_2, \ldots, x_{n-1}, e_n, x_n$$

of interleaved vertices $x_j \in V$ and edges $e_i \in E$,
such that $x_0 = u$ and $x_n = v$, and such that $e_i = \{x_{i-1}, x_i\} \in E$ for all $i \in \{1, \ldots, n\}$.

Such a path starts at u and ends at v. A path of length $n \geq 1$ is called a circuit (or cycle) if $n \geq 1$ and the path starts and ends at the same vertex, i.e., $u = v$.

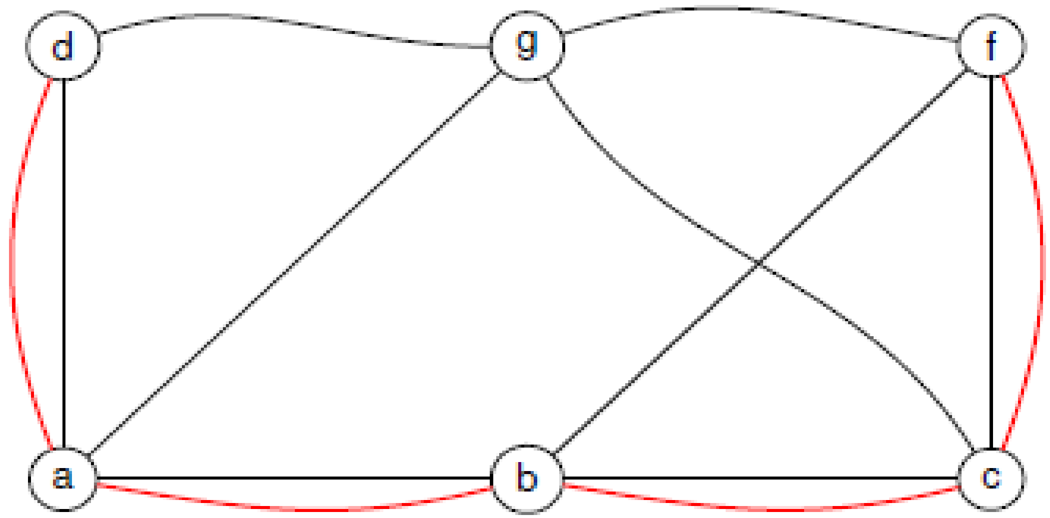
A path or circuit is called simple if it does not contain the same edge more than once.

When G = (V,E) is a simple undirected graph a path $x0, e1, \ldots, en, xn$ is determined uniquely by the sequence of vertices $x0, x1, \ldots, xn$. So, for simple undirected graphs we can denote a path by its sequence of vertices $x0, x1, \ldots, xn$.

Note 1: The word "simple" is overloaded. Don't confuse a simple undirected graph with a simple path. There can be a simple path in a non-simple graph, and a non-simple path in a simple graph.

Note 2: The terms "path" and "simple path" used in Rosen's book are not entirely standard. Other books use the terms "walk" and "trail" to denote "path" and "simple path", respectively. Furthermore, others use "path" itself to mean a walk that doesn't re-visit any vertex, except possibly the first and last in case it is a circuit. To stick to Rosen's terminology, we shall use the non-standard term tidy path to refer to such a walk

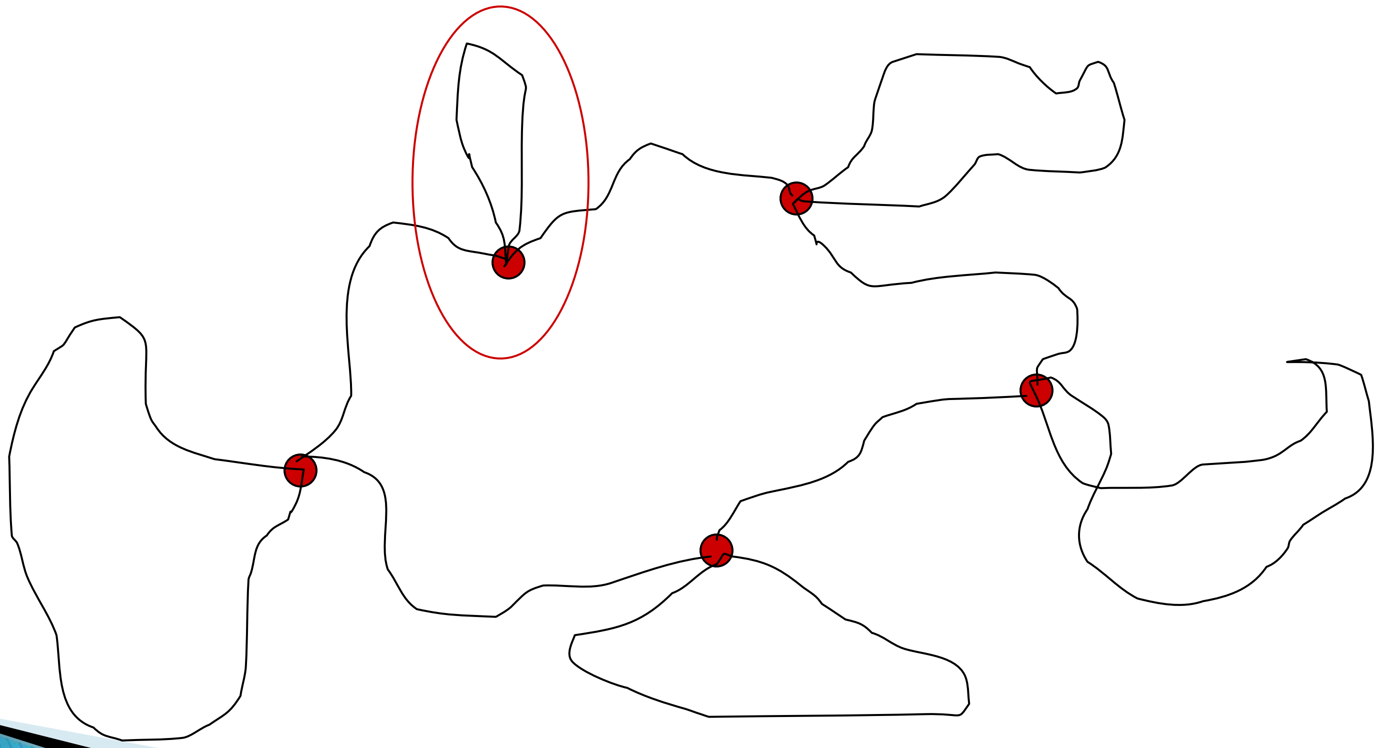
Example:



- d, a, b, c, f is a simple (and tidy) path of length 4.
- d, g, c, b, a, d is a simple (and tidy) circuit of length 5.
- a, b, g, f **is not** a path, because $\{b, g\}$ is not an edge.
- d, a, b, c, f, b, a, g **is** a path, but it **is not** a simple path, because the edge $\{a, b\}$ occurs twice in it.
- c, g, a, d, g, f is a simple path, but it **is not** a tidy path, because vertex g occurs twice in it.

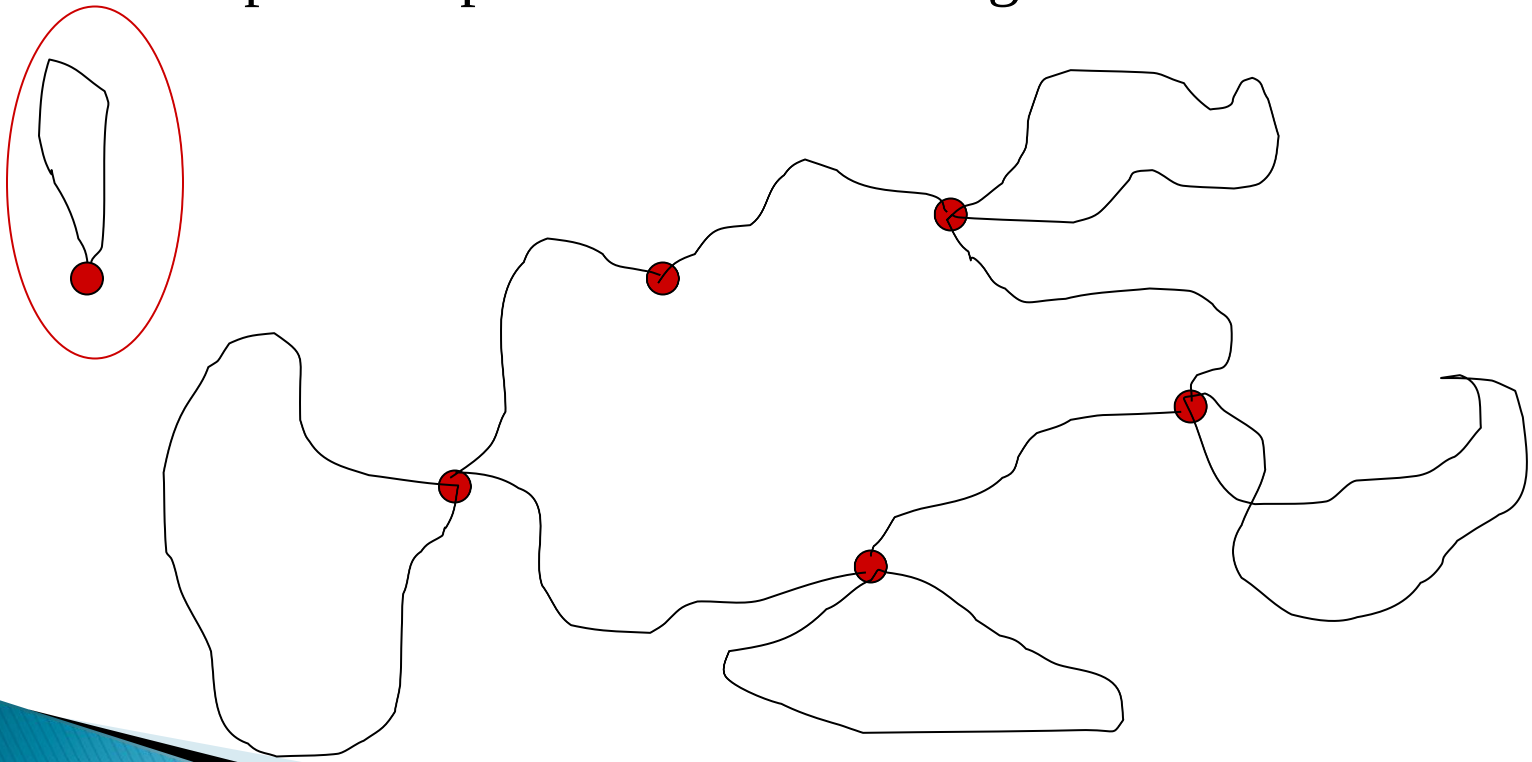
Walks, paths & circuits

- If G contains a closed walk of odd length, then it contains a circuit of odd length.



Walks, paths & circuits

- Now we have two closed walks
  - Length of one odd and other one is even.
  - Repeat the process with odd length closed walk.



Existence of circuit

- *Let G be a simple graph in which all vertices have degree at least two, then G contains a cycle.*

  - *Let P= $v_0v_1v_2 \ldots v_k$ be a longest path in G*
  - *Since degree of v_k is at least two so v_k has neighbor other than v_{k-1} (since simple graph)*
  - *If the new adjacent of v_k is 'v' and other than v_0 to v_k*
    - *Then we have a path $v_0v_1v_2 \ldots v_k\, v$, longer than the longest path, a contradiction*
    - *So, $v_i = v$ for $0 \le i \le k\text{-}2$. Therefore $v_i\, v_{i+1} \ldots . v_k v_i$ is a circuit*

Paths in directed graphs (same definitions)

- **Definition:** For an directed graph G = (V,E), an integer n ≥ 0,and vertices u, v 2 V, a path (or walk) of length n from u to v in G is a sequence of vertices and edges x0, e1, x1, e2, . . . , xn, en, such that x0 = u and xn = v, and such that ei = (xi−1, xi) 2 E for all i 2 {1, . . . , n}.
- When there are no multi-edges in the directed graph G, the path can be denoted (uniquely) by its vertex sequence x0, x1, . . . , xn.
- A path of length n ≥ 1 is called a circuit (or cycle) if the path starts and ends at the same vertex, i.e., u = v.
- A path or circuit is called simple if it does not contain the same edge more than once. (And we call it tidy if it does not contain the same vertex more than once, except possibly the first and last in case u = v and the path is a circuit (cycle)

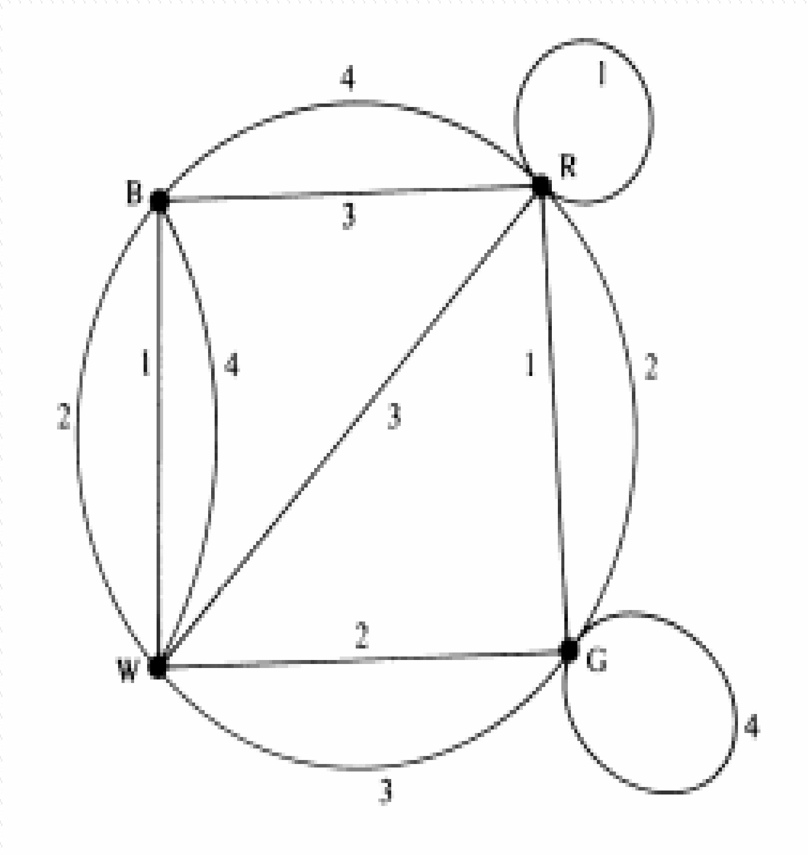
Sub graphs

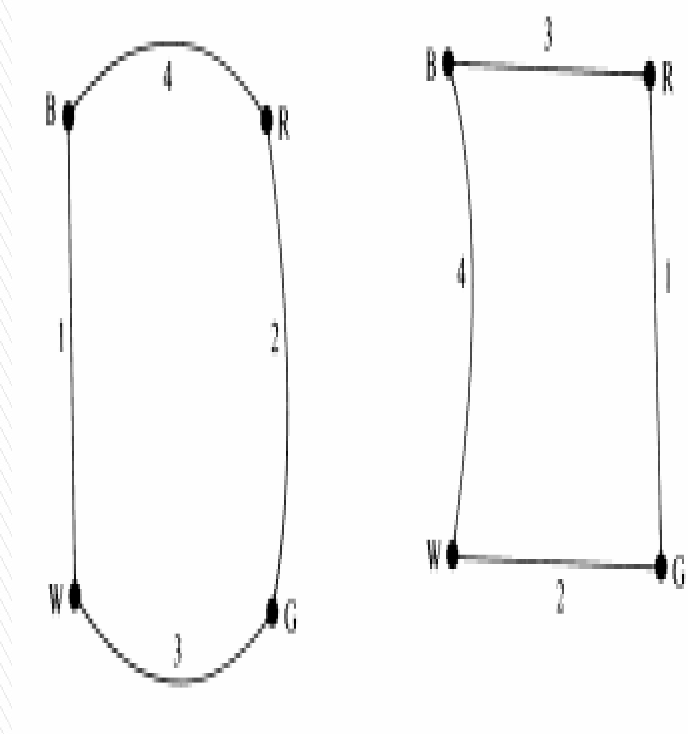
- A graph g is said to be sub-graph of a graph G if
  - All the vertices of g are in G
  - All the edges of g has same end vertices in g as in G.
- Every graph is its own subgraph
- A subgraph of a subgraph of G is a subgraph of G.
- A single vertex in G is a subgraph of G.
- A single edge in G, with the same end vertices, is a subgraph of G.

Sub-graphs

- Edge-disjoint sub-graph: two sub-graphs g_1 and g_2 of a graph G is said to be edge disjoint, if they do not have any edges in common.
- They may have vertices in common
- Similarly, we define vertex disjoint sub-graphs.
  - They have no vertex in common
  - Implies that no edge will be in common
- If two graphs are vertex disjoint, which implies they are edge disjoint also.

Example of Edge-disjoint sub-graph





(a) North-South Subgraph

(b) East-West Subgraph

Fig. 2-7 Two edge-disjoint subgraphs of the graph in Fig. 2-6.

Sub-graphs

- A complete sub-graph in G is known as clique in G.
- Any graph G1 for which given graph G is sub-graph, then G1 is super-graph of G.
- A sub-graph which contains all the vertices of the given graph is called a spanning sub-graph.

Connected, disconnected graph & components

- A graph is connected if we can reach any vertex from any other vertex by traveling along the edges.
- A graph is connected if there is at least one path between any pair of vertices.
  - Otherwise the graph is disconnected graph.
  - A null graph with more than one vertices is always a disconnected graph.

Components

- If a graph is disconnected, then it consists with two or more connected (sub)graphs.
- Each of these connected sub-graphs is called a component.
- A graph G is disconnected, iff the vertex set V can be partitioned into two sub-sets V_1 and V_2 such that there exists no edge in G whose one end vertex is in V_1 and the other in V_2.

Components

- Let G be a graph and g1, g2,..,gk be connected sub-graphs of G such that
  - Vertex sets are pair wise disjoint and covers V.
  - Edge sets are pair wise disjoint and also cover E.
  - Each of these sub-graphs are called component or connected components of G.
  - A graph G has a unique collection of connected components.
    - If connected graph only one component else ….

Components

- A given graph (connected or disconnected) has exactly two odd degree vertices and there must be a path between them.
  - A component can be treated as a graph.
  - there are even number of odd degree vertices in a graph.
  - These two odd degree vertices must be member of a component. Hence, there is a path between them.

Connected, disconnected graph & components

- A simple graph with n vertices and k components can have at most (n-k)(n-k+1)/2 edges.
  - Number of vertices in each component is n_i, $1 \le i \le k$. $n_1 + n_2 + \ldots n_k = n$ and $n_i \ge 1$.
  - Maximum edges in component i is $n_i(n_i-1)/2$.
  - Maximum number of edges in the graph is

$$\frac{1}{2}\sum_{i=1}^{k} n_i(n_i-1) = \frac{1}{2}\sum_{i=1}^{k} n_i^2 - \frac{n}{2}$$

$$\sum_{i=1}^{k}(n_i-1) = n-k \Rightarrow [\sum_{i=1}^{k}(n_i-1)]^2 = n^2-2nk+k^2$$

$$(n_1-1)^2+(n_2-1)^2+\ldots+(n_k-1)^2+2(n_1-1)(n_2-1)+2(n_2-1)(n_3-1)+\ldots$$

Connected, disconnected graph & components

$$(n_1^2 - 2n_1 + 1) + (n_2^2 - 2n_2 + 1) + \ldots . + non - negative\,cross\;\;product\;\;terms$$

$$\sum_{i=1}^{k} n_i^{\;2} - 2n + k + non\text{-}negative\,terms = n^2 + k^2 - 2nk$$

$$\sum_{i=1}^{k} n_i^2 \le n^2 + k^2 - 2nk - k + 2n$$

$$\frac{1}{2}\sum_{i=1}^{k} n_i^2 - \frac{n}{2} \le \frac{1}{2}(n^2 + k^2 - 2nk - k + 2n - n)$$

$$= \frac{1}{2}(n^2 - nk + n + k^2 - nk - k) = \frac{1}{2}(n-k)(n-k+1)$$

Connected, disconnected graph & components

- The complement or **inverse** of a graph *G* is a graph *H* on the same vertices such that two vertices of *H* are adjacent if and only if they are not adjacent in *G*.
- If a graph G is disconnected, the complement G_c is connected.
  - Consider any two vertices u,v.
  - First, say, they from different components
    - In the complement graph they are adjacent [connected]
  - Second, say, they are from same component
    - w is another vertex from another component.
    - There is an edge between u, w and v, w.
    - U and v are connected.