Graphs

Chapter 10

Adapted Version

Chapter Summary

- Graphs and Graph Models
- Graph Terminology and Special Types of Graphs
- Representing Graphs and Graph Isomorphism

Graph Terminology and Special Types of Graphs

Section 10.2

Section Summary

- Basic Terminology
- Some Special Types of Graphs
- Bipartite Graphs
- New Graphs from Old

Basic Terminology

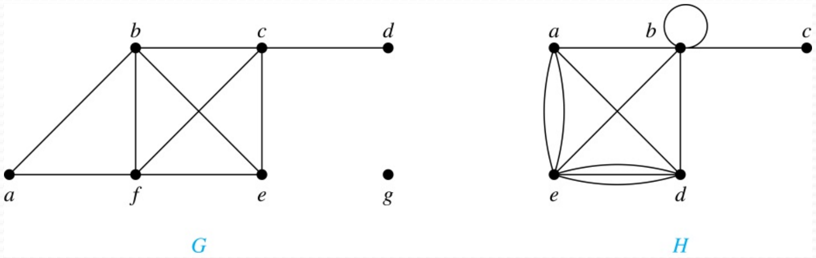
Definition 1. Two vertices u, v in an undirected graph G are called *adjacent* (or *neighbors*) in G if there is an edge e between u and v. Such an edge e is called *incident with* the vertices u and v and e is said to *connect* u and v.

Definition 2. The set of all neighbors of a vertex v of $G = (V, E)$, denoted by $N(v)$, is called the *neighborhood* of v. If A is a subset of V, we denote by $N(A)$ the set of all vertices in G that are adjacent to at least one vertex in A. So, $N(A) = \bigcup_{v \in A} N(v)$.

Definition 3. The *degree of a vertex in a undirected graph* is the number of edges incident with it, except that a loop at a vertex contributes two to the degree of that vertex. The degree of the vertex v is denoted by $\deg(v)$.

Degrees and Neighborhoods of Vertices

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



Solution:

G: $\deg(a) = 2$, $\deg(b) = \deg(c) = \deg(f) = 4$, $\deg(d) = 1$,
$\deg(e) = 3$, $\deg(g) = 0$.
$N(a) = \{b, f\}$, $N(b) = \{a, c, e, f\}$, $N(c) = \{b, d, e, f\}$, $N(d) = \{c\}$,
$N(e) = \{b, c, f\}$, $N(f) = \{a, b, c, e\}$, $N(g) = \varnothing$.

H: $\deg(a) = 4$, $\deg(b) = \deg(e) = 6$, $\deg(c) = 1$, $\deg(d) = 5$.
$N(a) = \{b, d, e\}$, $N(b) = \{a, b, c, d, e\}$, $N(c) = \{b\}$,
$N(d) = \{a, b, e\}$, $N(e) = \{a, b, d\}$.

Degrees of Vertices

- **Theorem 1** (***Handshaking Theorem***): If $G = (V,E)$ is an 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. ◀

Think about the graph where vertices represent the people at a party and an edge connects two people who have shaken hands.

Handshaking Theorem

We now give two examples illustrating the usefulness of the handshaking theorem.

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

Solution: Because the sum of the degrees of the vertices is $6 \cdot 10 = 60$, the handshaking theorem tells us that $2m = 60$. So the number of edges $m = 30$.

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.

Degree of Vertices (*continued*)

Theorem 2: An undirected graph has an even number of vertices of odd degree.

Proof: Let V_1 be the vertices of even degree and V_2 be the vertices of odd degree in an undirected graph $G = (V, E)$ with m edges. Then

even

$$2m = \sum_{v \in V} \deg(v) = \sum_{v \in V_1} \deg(v) + \sum_{v \in V_2} \deg(v).$$

must be even since deg(v) is even for each $v \in V_1$

This sum must be even because $2m$ is even and the sum of the degrees of the vertices of even degrees is also even. Because this is the sum of the degrees of all vertices of odd degree in the graph, there must be an even number of such vertices.

Directed Graphs

Recall the definition of a directed graph.

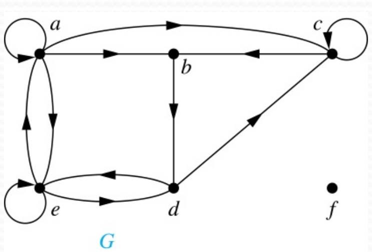
Definition: An *directed graph* $G = (V, E)$ consists of V, a nonempty set of *vertices* (or *nodes*), and E, a set of *directed edges* or *arcs*. Each edge is an ordered pair of vertices. The directed edge (u,v) is said to start at u and end at v.

Definition: Let (u,v) be an edge in G. Then u is the *initial vertex* of this edge and is *adjacent to* v and v is the *terminal* (or *end*) *vertex* of this edge and is *adjacent from* u. The initial and terminal vertices of a loop are the same.

Directed Graphs (*continued*)

Definition: The *in-degree of a vertex v*, denoted $deg^{-}(v)$, is the number of edges which terminate at v. The *out-degree of v*, denoted $deg^{+}(v)$, is the number of edges with v as their initial vertex. Note that a loop at a vertex contributes 1 to both the in-degree and the out-degree of the vertex.

Example: In the graph G we have



$\deg^{-}(a) = 2$, $\deg^{-}(b) = 2$, $\deg^{-}(c) = 3$, $\deg^{-}(d) = 2$, $\deg^{-}(e) = 3$, $\deg^{-}(f) = 0$.

$\deg^{+}(a) = 4$, $\deg^{+}(b) = 1$, $\deg^{+}(c) = 2$, $\deg^{+}(d) = 2$, $\deg^{+}(e) = 3$, $\deg^{+}(f) = 0$.

Directed Graphs (*continued*)

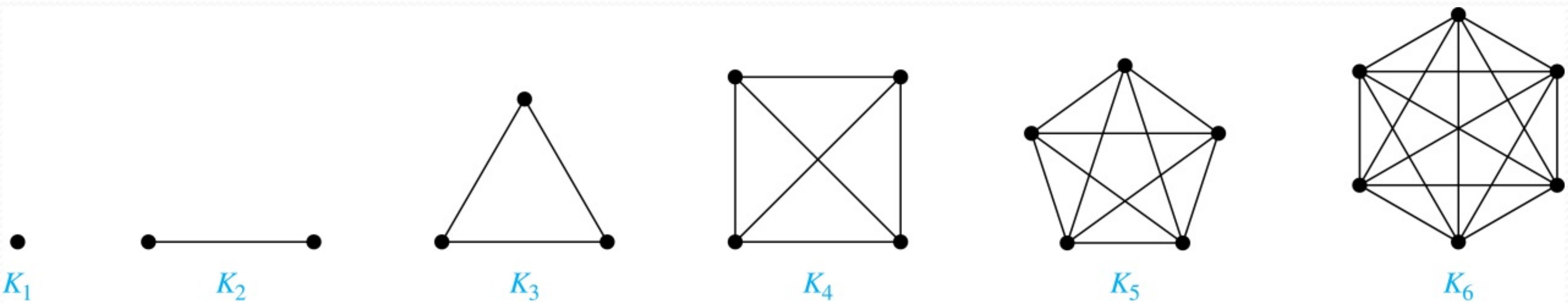
Theorem 3: Let $G = (V, E)$ be a graph with directed edges. Then:

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

Proof: The first sum counts the number of outgoing edges over all vertices and the second sum counts the number of incoming edges over all vertices. It follows that both sums equal the number of edges in the graph. ◀

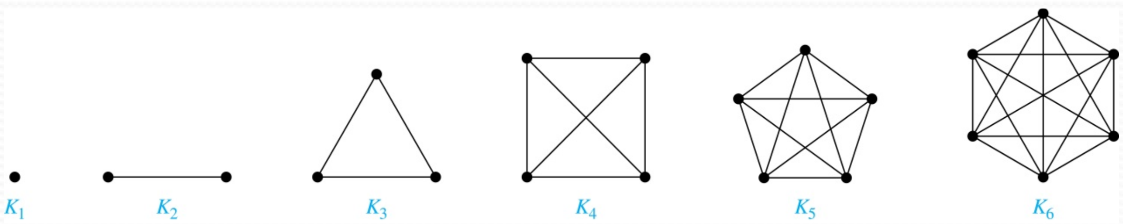
Special Types of Simple Graphs: Complete Graphs

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 Simple Graphs: Complete Graphs

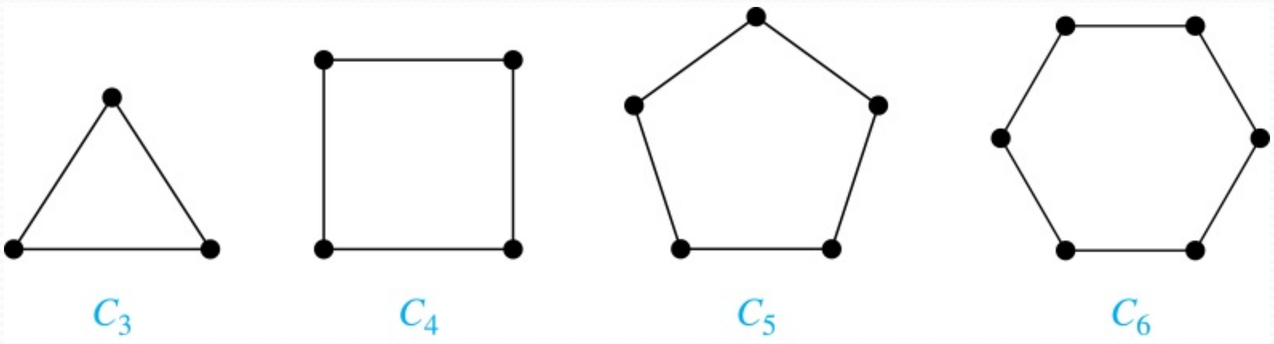
A *complete graph on n vertices*, denoted by K_n, is the simple graph that contains exactly one edge between each pair of distinct vertices.



- How many vertices and how many edges does $\boldsymbol{K_n}$ has?

Special Types of Simple Graphs: Cycles and Wheels

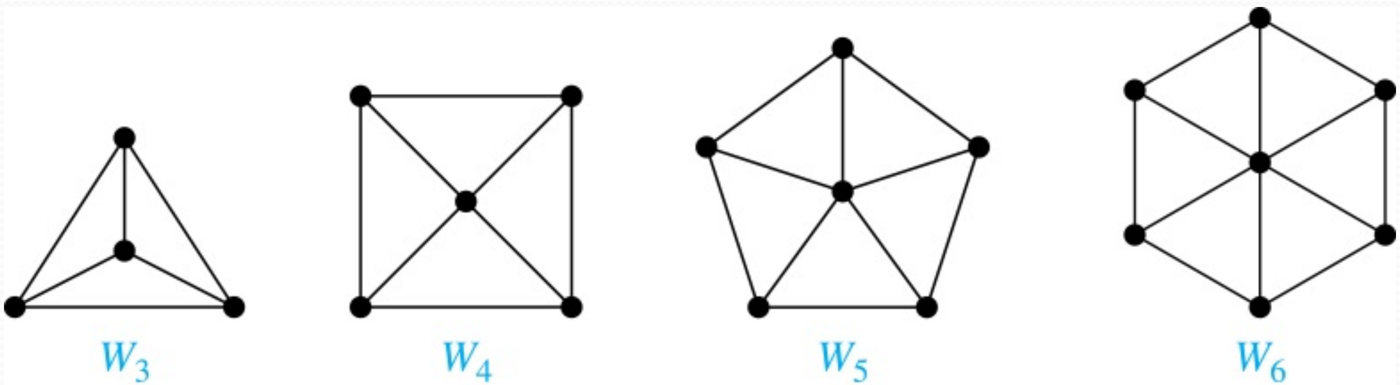
A *cycle* C_n for $n \geq 3$ consists of n vertices $v_1, v_2, \cdots, v_n$, and edges $\{v_1, v_2\}, \{v_2, v_3\}, \cdots, \{v_{n-1}, v_n\}, \{v_n, v_1\}$.



How many vertices and how many edges does $\boldsymbol{C_n}$ has?

Special Types of Simple Graphs: Cycles and Wheels

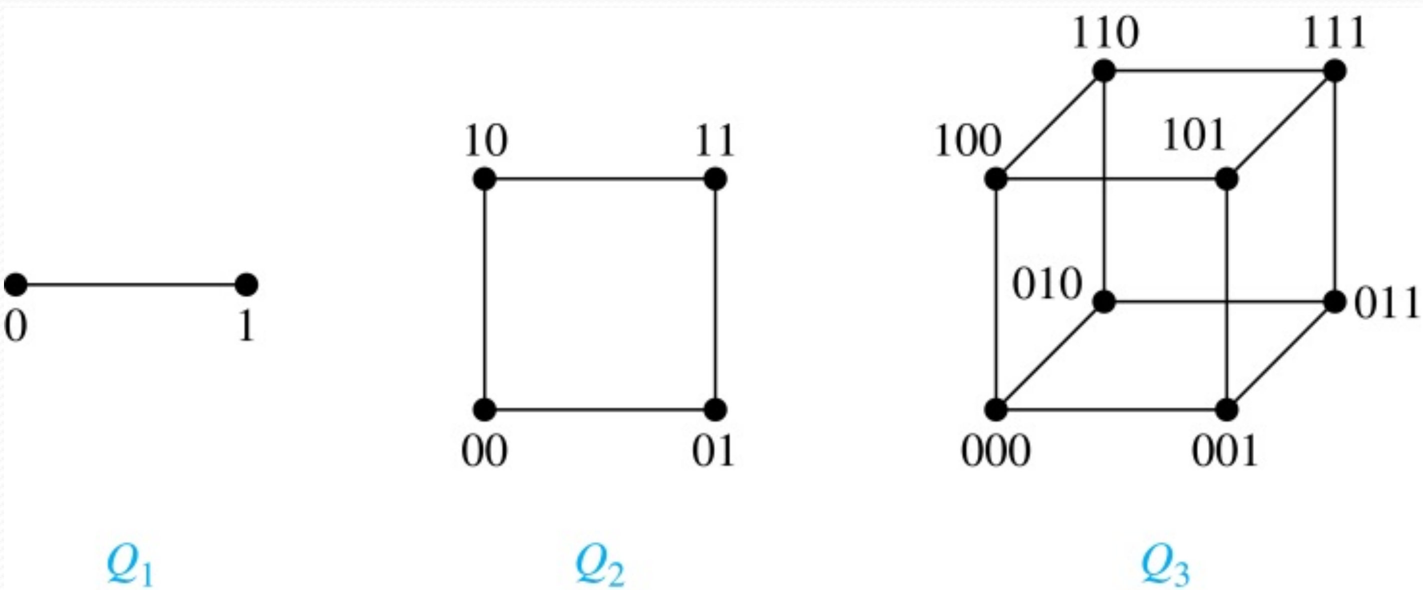
A *wheel* W_n is obtained by adding an additional vertex to a cycle C_n for $n \geq 3$ and connecting this new vertex to each of the n vertices in C_n by new edges.



How many vertices and how many edges does $\boldsymbol{W_n}$ has?

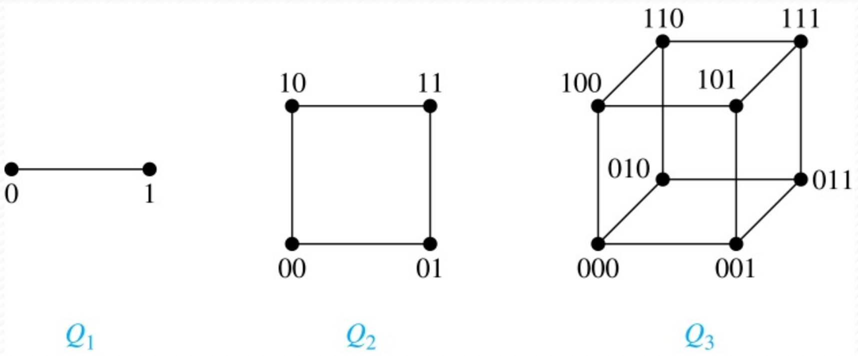
Special Types of Simple Graphs: *n*-Cubes

An *n-dimensional hypercube*, or *n-cube,* $\boldsymbol{Q_n}$, is a graph with 2^n vertices representing all bit strings of length n, where there is an edge between two vertices that differ in exactly one bit position.



Special Types of Simple Graphs: *n*-Cubes

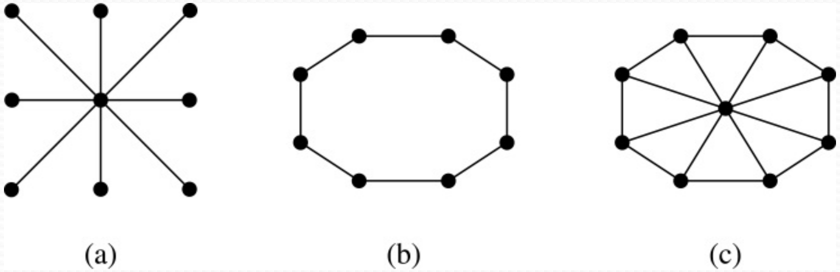
An *n-dimensional hypercube*, or *n-cube,* Q_n, is a graph with 2^n vertices representing all bit strings of length n, where there is an edge between two vertices that differ in exactly one bit position.



How many vertices and how many edges does C_n has?

Special Types of Graphs and Computer Network Architecture

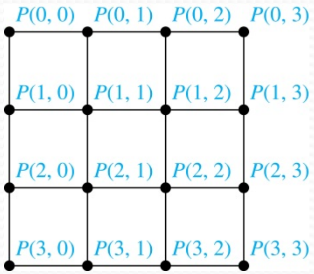
Various special graphs play an important role in the design of computer networks.



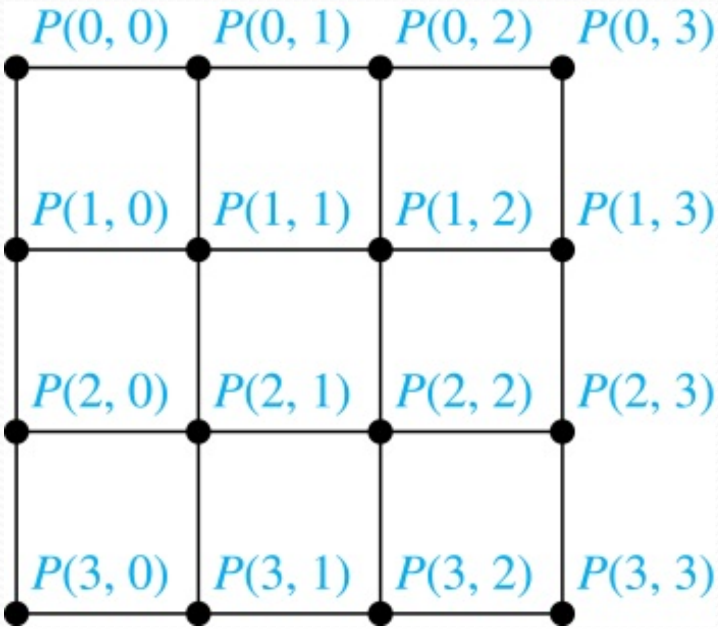
- Some local area networks use a ***star topology***, which is a complete bipartite graph $K_{1,n}$, as shown in (a). All devices are connected to a central control device.
- Other local networks are based on a ***ring topology***, where each device is connected to exactly two others using C_n, as illustrated in (b). Messages may be sent around the ring.
- Others, as illustrated in (c), use a W_n – based topology, combining the features of a star topology and a ring topology.

Special Types of Graphs and Computer Network Architecture

- Various special graphs also play a role in parallel processing where processors need to be interconnected as one processor may need the output generated by another.
  - The *n-dimensional hypercube*, or *n-cube,* $\boldsymbol{Q_n}$, is a common way to connect processors in parallel, e.g., Intel Hypercube.
  - Another common method is the *mesh* network, illustrated here for 16 processors.



Mesh Network

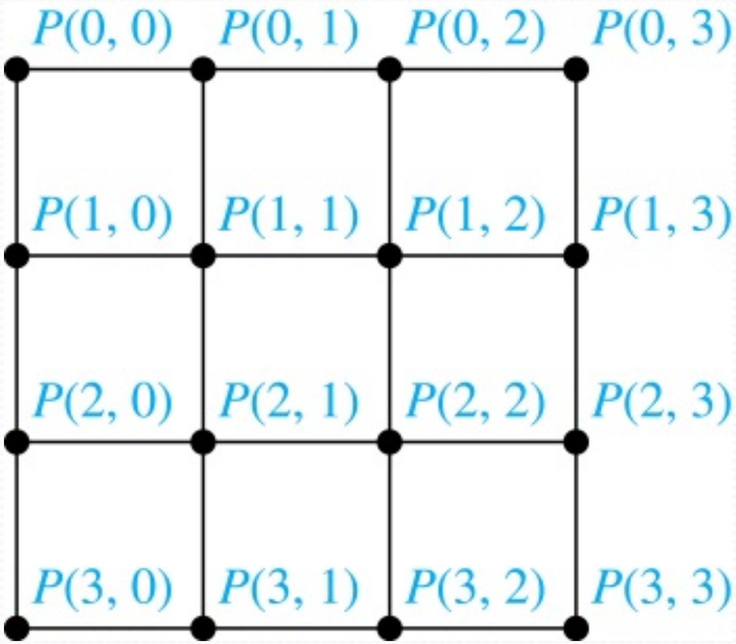


- Draw the mesh network for interconnecting nine parallel processors.

Mesh Network

- In a variant of a mesh network for interconnecting $n=m^2$ processors, processor $P(i, j)$ is connected to the four processors $P((i \pm 1) \textbf{ mod } m, j)$ and $P(i, (j \pm 1) \textbf{ mod } m)$ so that connections wrap around the edges of the mesh.

- **Draw this variant of the mesh network for 16 processors.**



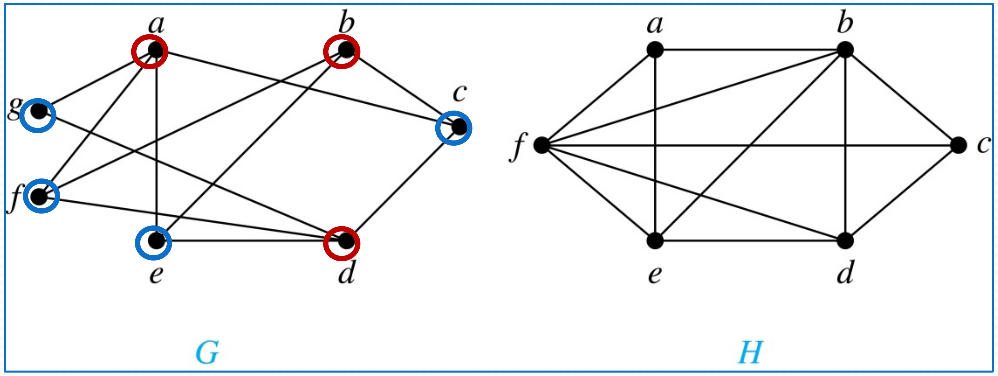
Bipartite Graphs

Definition: A simple graph G is bipartite if V can be partitioned into two disjoint subsets V_1 and V_2 such that every edge connects a vertex in V_1 and a vertex in V_2. In other words, there are no edges which connect two vertices in V_1 or in V_2.

Bipartite Graphs

It is not hard to show that an equivalent definition of a bipartite graph is a graph where it is possible to color the vertices **red** or **blue** so that *no two adjacent vertices are the same color*.

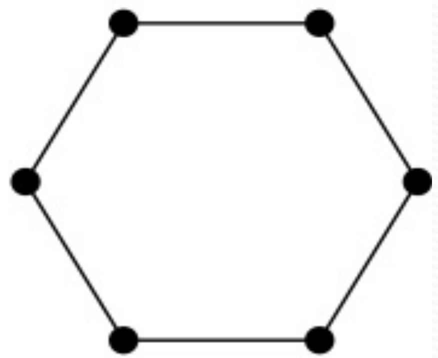
G is bipartite



H is not bipartite since if we color *a* red, then the adjacent vertices *f* and *b* must both be blue.

Bipartite Graphs (*continued*)

Example: Show that C_6 is bipartite.

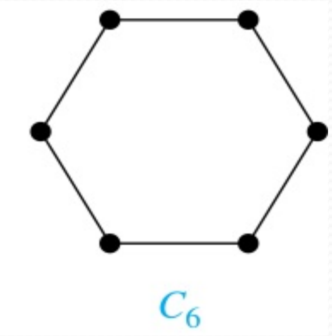


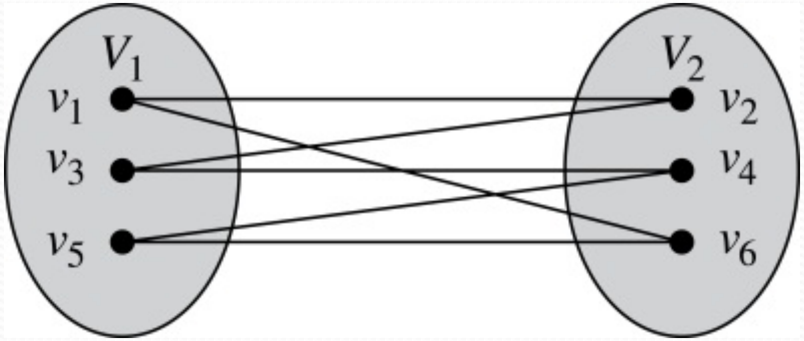
C_6

Bipartite Graphs (*continued*)

Example: Show that C_6 is bipartite.

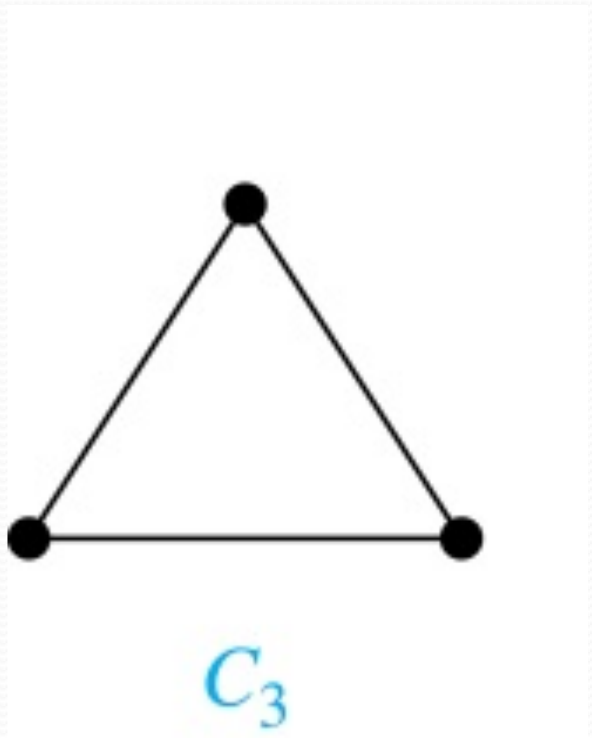
Solution: We can partition the vertex set into $V_1 = \{v_1, v_3, v_5\}$ and $V_2 = \{v_2, v_4, v_6\}$ so that every edge of C_6 connects a vertex in V_1 and V_2 .





Bipartite Graphs (*continued*)

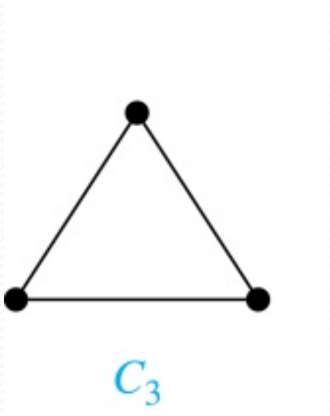
Example: Show that C_3 is not bipartite.



Bipartite Graphs (*continued*)

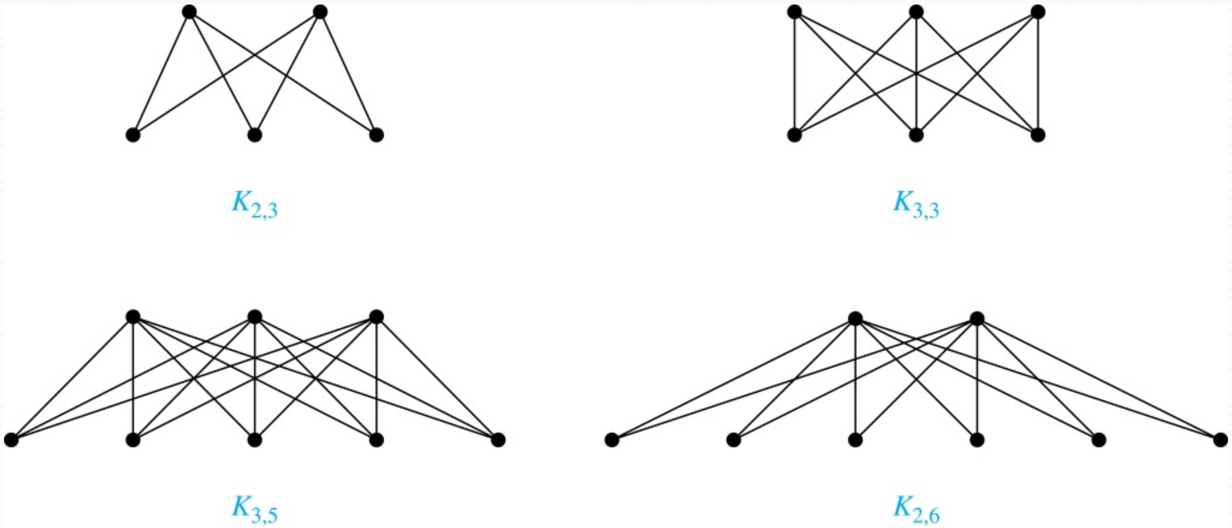
Example: Show that C_3 is not bipartite.

Solution: If we divide the vertex set of C_3 into two nonempty sets, one of the two must contain two vertices. But in C_3 every vertex is connected to every other vertex. Therefore, the two vertices in the same partition are connected. Hence, C_3 is not bipartite.

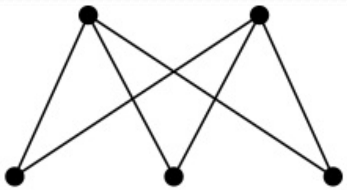


Complete Bipartite Graphs

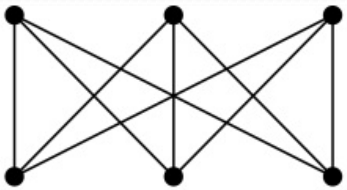
Definition: A *complete bipartite graph* $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets V_1 of size m and V_2 of size n such that there is an edge from every vertex in V_1 to every vertex in V_2.



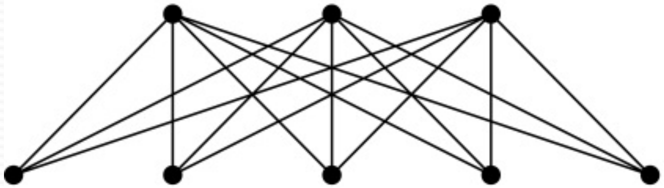
Complete Bipartite Graphs



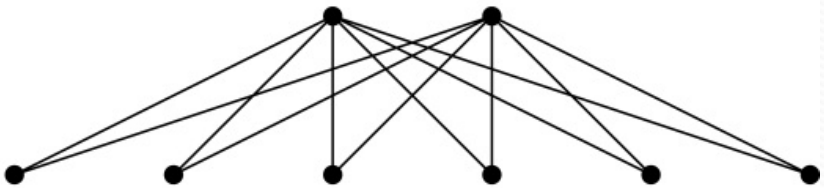
$K_{2,3}$



$K_{3,3}$



$K_{3,5}$

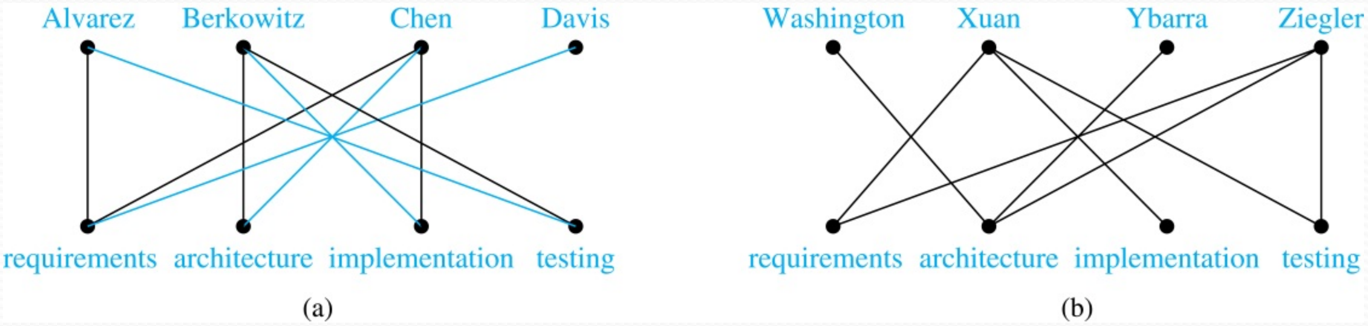


$K_{2,6}$

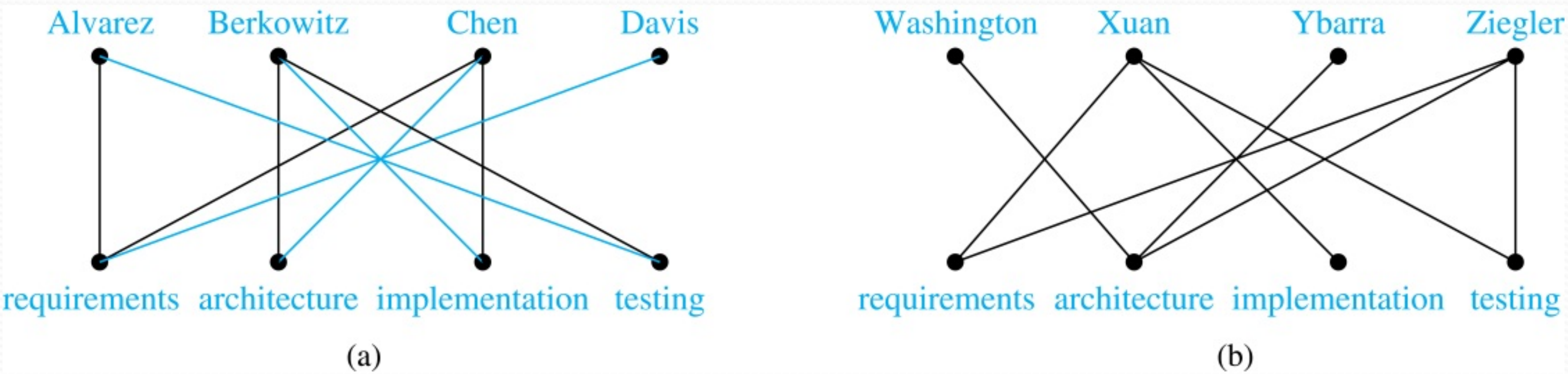
- How many vertices and how many edges does $K_{m,n}$ has?

Bipartite Graphs and Matchings

- Bipartite graphs are used to model applications that involve matching the elements of one set to elements in another, for example:
- *Job assignments* - vertices represent the jobs and the employees, edges link employees with those jobs they have been trained to do. A common goal is to match jobs to employees so that the most jobs are done.



Bipartite Graphs and Matchings



Bipartite Graphs and Matchings

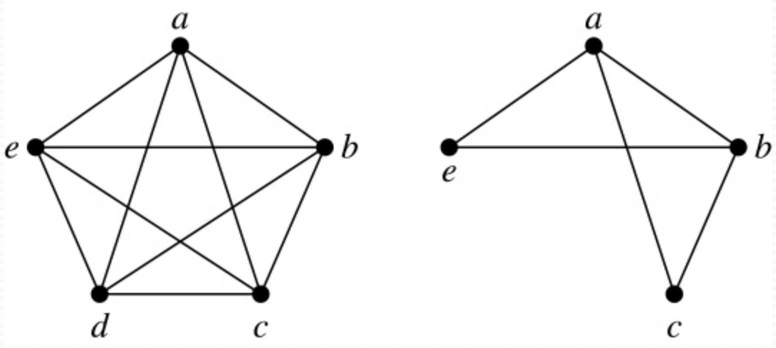
- Bipartite graphs are used to model applications that involve matching the elements of one set to elements in another, for example:

- *Marriage* - vertices represent the men and the women and edges link a a man and a woman if they are an acceptable spouse. We may wish to find the largest number of possible marriages.

New Graphs from Old

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

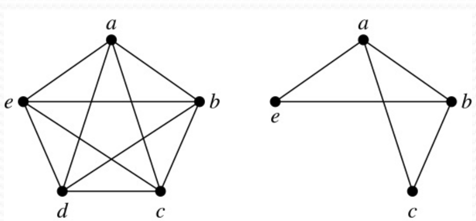
Example: Here we show K_5 and one of its subgraphs.

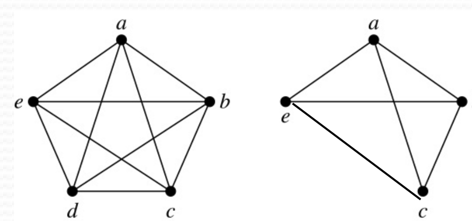


New Graphs from Old

Definition: Let $G = (V, E)$ be a simple graph. The *subgraph* ***induced*** by a subset W of the vertex set V is the graph (W,F), where the edge set F contains an edge in E if and only if both endpoints are in W.

Example: Here we show K_5 and the subgraph induced by $W = \{a,b,c,e\}$.

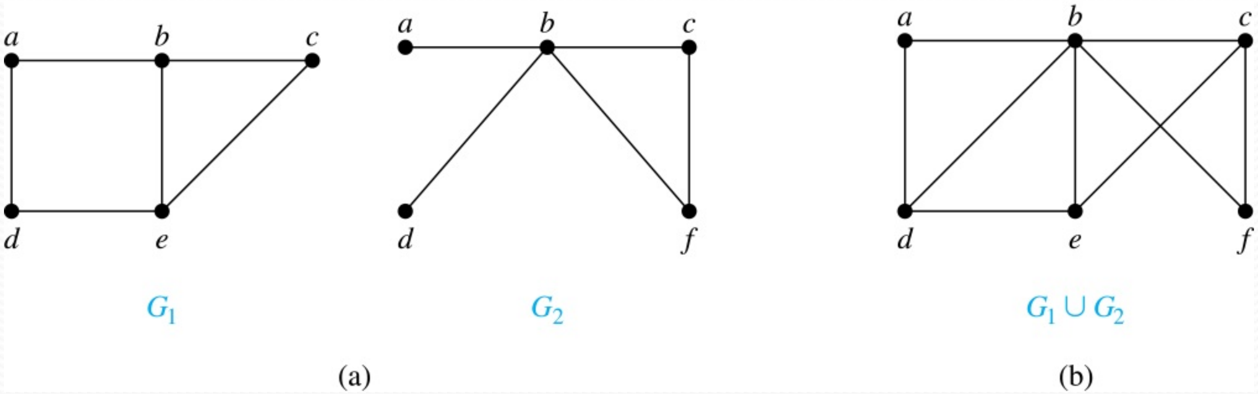




New Graphs from Old (*continued*)

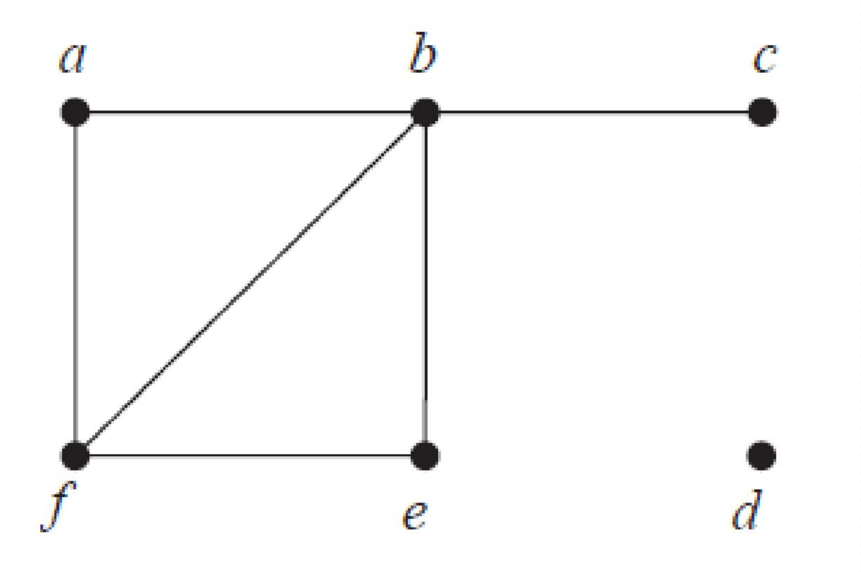
Definition: The *union* of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$. The union of G_1 and G_2 is denoted by $G_1 \cup G_2$.

Example:



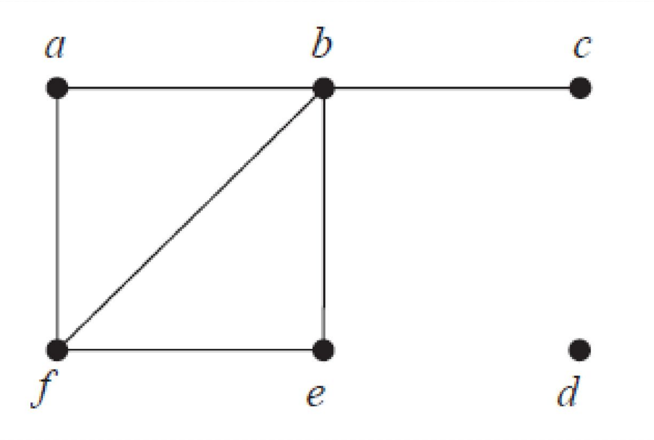
New Graphs from Old (*continued*)

- For the following graph find the subgraph **induced by** the vertices a, b, c, and f .



New Graphs from Old (*continued*)

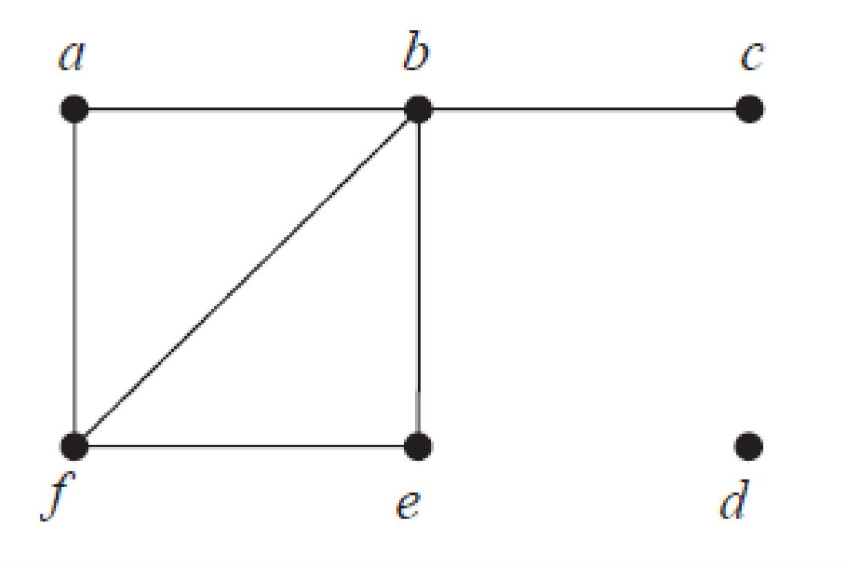
- For the following graph find the subgraph **induced by** the vertices a, b, c, and f.



- $(\{a, b, c, f\}, \{\{a, b\}, \{a, f\}, \{b, c\}, \{b, f\}\})$

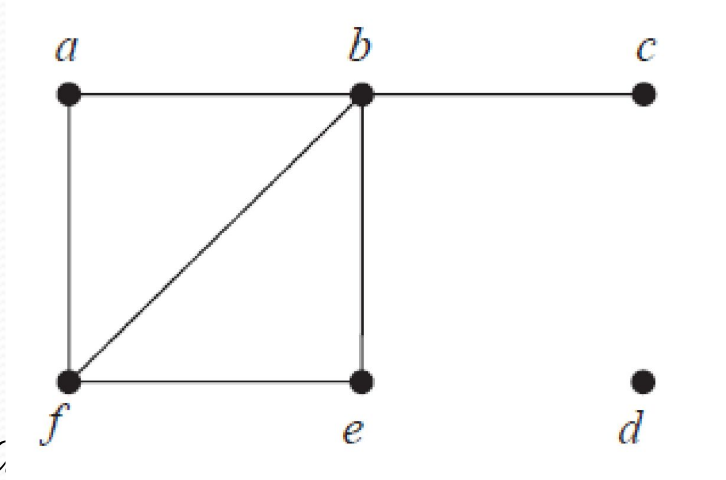
New Graphs from Old (*continued*)

- For the following graph find the new graph obtained by **contracting** the edge connecting b and f.



New Graphs from Old (*continued*)

- For the following graph find the new graph obtained by **contracting** the edge connecting b and f.



- $(\{a, \mathbf{x}, c, e,$ … $\}\})$