Section 8.2
Graph Terminology

Undirected Graphs

Definition: Two vertices *u, v* in *V* are *adjacent* or *neighbors* if there is an edge e between *u* and *v*.

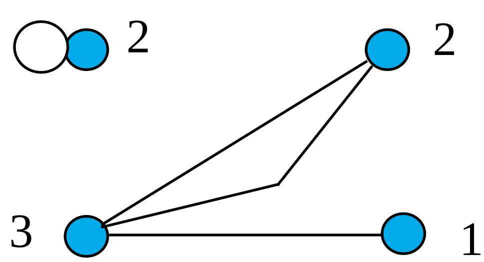
The edge e *connects* *u* and *v*.

The vertices *u* and *v* are endpoints of e.

Definition: The *degree* of a vertex *v*, denoted deg(*v*), is the number of edges for which it is an endpoint.

A loop contributes twice in an undirected graph.

Example:



• If deg(*v*) = 0, v is called *isolated.*

• If deg(v) = 1, v is called *pendant.*

The Handshaking Theorem:

Let $G = (V, E)$. Then

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

Proof:

Each edge contributes twice to the degree count of all vertices.

Q. E. D.

Example:

If a graph has 5 vertices, can each vertex have degree 3? 4?

• The sum is 3•5 = 15 which is an odd number. Not possible.

• The sum is 20 = 2 | E | and 20/2 = 10. May be possible.

Theorem: A graph has an even number of vertices of odd degree.

Proof:

Let V1 = vertices of odd degree

V2= vertices of even degree

The sum must be even. But

- odd times odd = odd
- odd times even = even
- even times even = even
- even plus odd = odd

It doesn't matter whether V2 has odd or even cardinality.

V1 cannot have odd cardinality.

Q. E. D.

Example:

It is not possible to have a graph with 3 vertices each of which has degree 1.

Directed Graphs

Definition: Let $<u, v>$ be an edge in G. Then u is an *initial vertex* and is *adjacent to* v and v is a *terminal vertex* and is *adjacent from u*.

Definition: The *in degree* of a vertex v, denoted $\deg^{-}(v)$ is the number of edges which terminate at v.

Similarly, the *out degree* of v, denoted $\deg^{+}(v)$, is the number of edges which initiate at v.

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

Special Simple Graphs

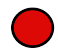
- Complete graphs - K_n: the simple graph with

  - n vertices

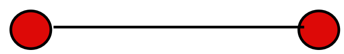
  - exactly one edge between every pair of distinct vertices.

Maximum redundancy in local area networks and processor connection in parallel machines.

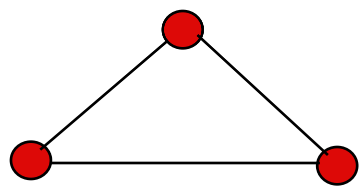
Examples:



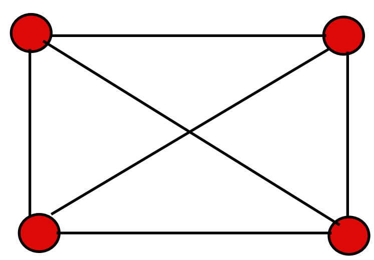
K_1



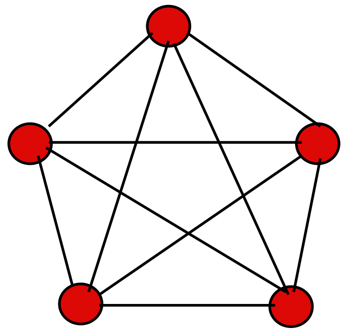
K_2



K_3



K_4

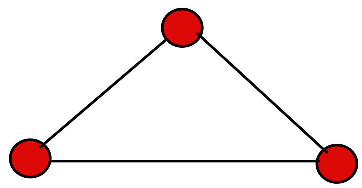


K_5

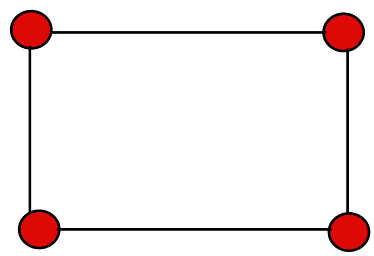
Note: K5 is important because it is the simplest nonplanar graph: It cannot be drawn in a plane with nonintersecting edges.

- Cycles:

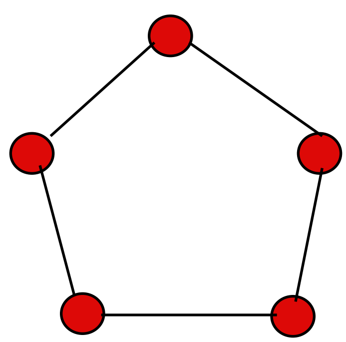
C_n is an n vertex graph which is a cycle. Local area networks are sometimes configured this way called *Ring* networks.



C_3



C_4

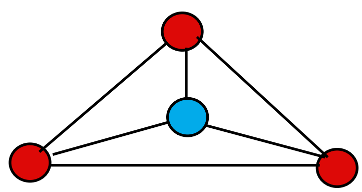


C_5

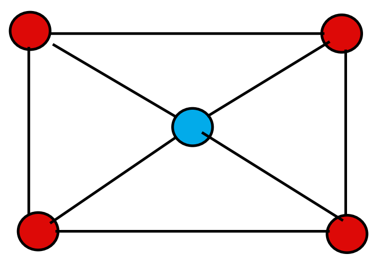
• Wheels:

Add one additional vertex to the cycle C_n and add an edge from each vertex to the new vertex to produce W_n.

Provides redundancy in local area networks.



W_3



W_4

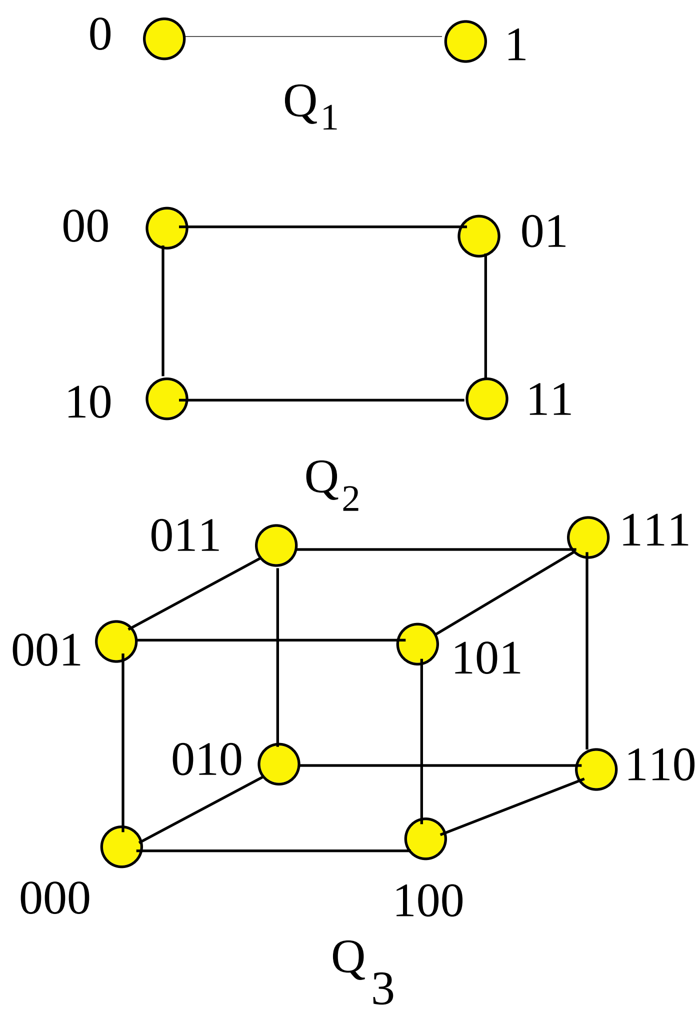
- n-Cubes:

Q_n is the graph with 2^n vertices representing bit strings of length n.

An edge exists between two vertices that differ by one bit position.

A common way to connect processors in parallel machines.

Intel Hypercube.



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.

Note: There are no edges which connect vertices in V_1 or in V_2.

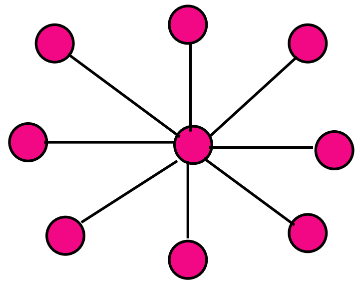
A bipartite graph is *complete* if there is an edge from every vertex in V_1 to every vertex in V_2, denoted $K_{m,n}$ where $m = | V_1 |$ and $n = | V_2 |$.

Examples:

- Suppose bigamy is permitted but not same sex marriages and males are in V1 and females in V2 and an edge represents a marriage. If every male is married to every female then the graph is complete.

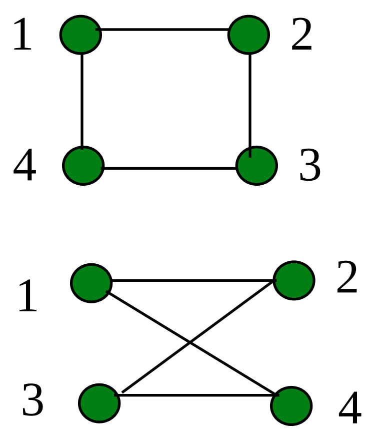
- Supplier, warehouse transportation models are bipartite and an edge indicates that a given supplier sends inventory to a given warehouse.

- A Star network is a $K_{1,n}$ bipartite graph.

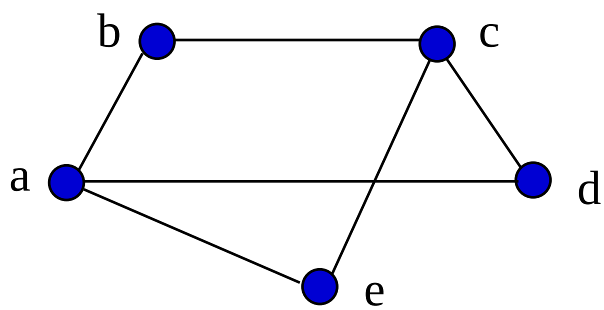


$K_{1,8}$

• C_k for k even is a bipartite graph: even numbered vertices in *V1*, odd numbered in *V2*.



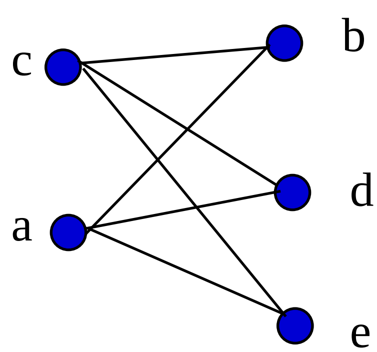
• Is the following graph bipartite?



If *a* is in *V1* then *e*, d and *b* must be in *V2* (why?).

Then *c* is in *V1* and there is no inconsistency.

We rearrange the graph as follows:



New Graphs from Old

Definition: *(W, F)* is a *subgraph* of *G* = *(V, E)* if

W   *V* and *F*   *E*.

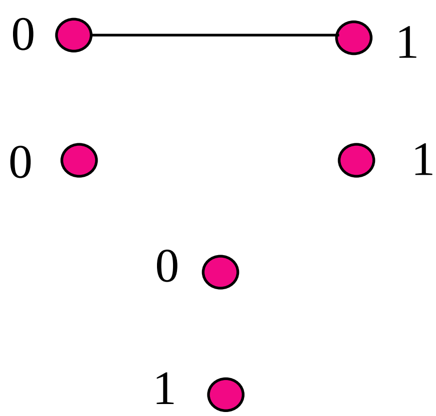
Definition: If *G1* and *G2* are simple then

$$G1 \quad G2 = (V1 \quad V2, E1 \quad E2)$$

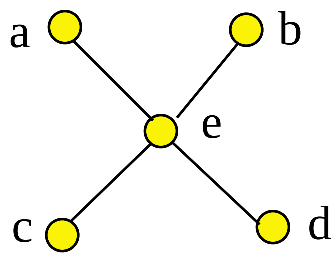
and the graph is simple.

Examples:

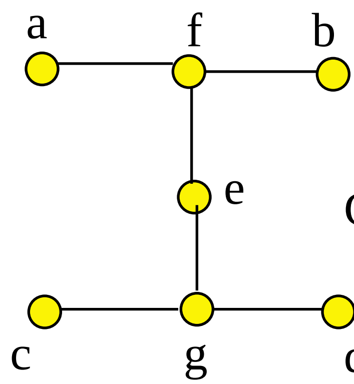
- Find the subgraphs of Q_1:



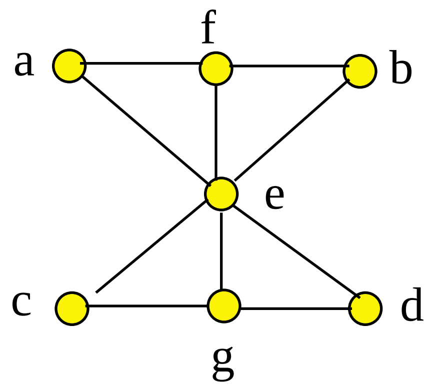
- Count the number of subgraphs of a given graph.
- Find the union of the two graphs G_1 and G_2:



G_1



G_2



G_1 G_2

Note: The important properties of a graph do not depend on how we draw it. We want to be able to identify two graphs that are the same (up to labeling of the vertices).