Unit – VI
Graph theory

Representation of Graphs

Graph *G*

(*V*, *E*, γ)

V *Set of vertices*

E Set of *edges*

γ **Function that assigns vertices {v, w} to each edge**

γ(e) = {v, w}

e *Edge between vertices v and w*

Example 1:

$V = \{1, 2, 3, 4\}$

$E = \{e_1, e_2, e_3, e_4, e_5\}$

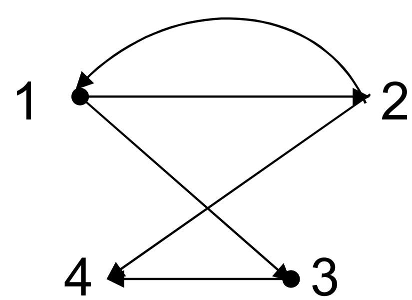
$\gamma(e_1) = \{1, 2\}$

$\gamma(e_2) = \{4, 3\}$

$\gamma(e_3) = \{1, 3\}$

$\gamma(e_4) = \{2, 4\}$

$\gamma(e_5) = \{2, 1\}$



Degree of a vertex

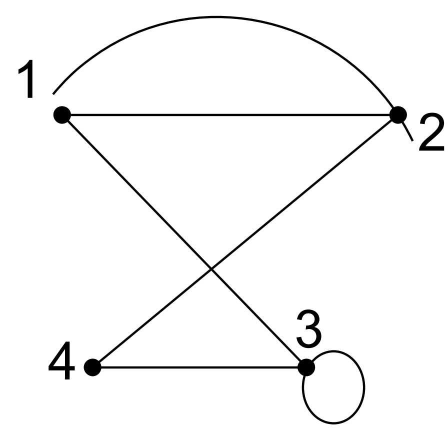
No. of edges at a vertex

Loop

Edge from a vertex to itself

Degree is 2

Example 2:



Vertex	***Degree***
1	**3**
2	**3**
3	**4**
4	**2**

Isolated vertex

Vertex with degree 0

Adjacent vertices

Pair of vertices that determine an edge

Path π in a graph G

Pair (Vπ , Eπ) of sequences

Vπ : v1, v2, … , vk **Vertex sequence**

Eπ: e1, e2, ..., ek – 1 **Edge sequence**

Each successive pair vi, vi+1 of vertices is adjacent in *G*

Edge ei has vi and vi+1 as end points

No edge occurs more than once in the edge sequence

Circuit

Path that begins and ends at the same vertex.

Simple path

No vertex appears more than once in the vertex sequence

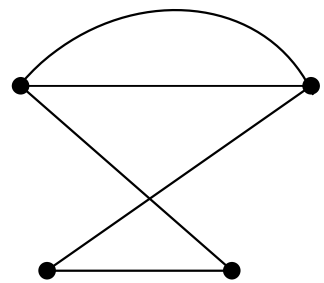
Simple circuit

v1 = vk

***Connected* graph**

Path from any vertex to any other vertex

Example:



Connected Graph

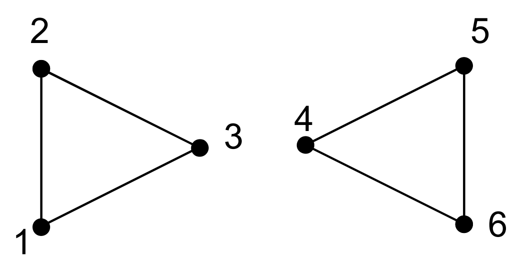
***Disconnected* graph**

Not connected graph

***Components* of the graph**

Various connected pieces in the disconnected graph

Example:

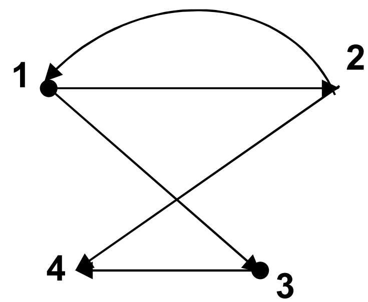


Disconnected Graph having 2 components

Adjacency Matrix representation of graphs

Examples

1. Give adjacency matrix representation for the graph G given below:



Adjacency Matrix representation for the graph G

$$
\begin{array}{c c}
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}
\end{array}
$$

- A tree is a simple graph G such that there is a unique simple path between each pair of vertices.
- A rooted tree is a tree in which there is one designated vertex, called a root.
- A rooted tree is a directed tree if there is a root from which there is a directed path to each vertex.
- The level of a vertex v in a rooted tree is the length of the path to v from the root.
- A tree T with only one vertex is called a trivial tree; otherwise T is nontrivial tree.
- If a graph G is connected and e is an edge such that G-e is not connected , then e is said to be a bridge or a cut edge. If v is a vertex of G such that G-v is not connected ,then v is a cut vertex.

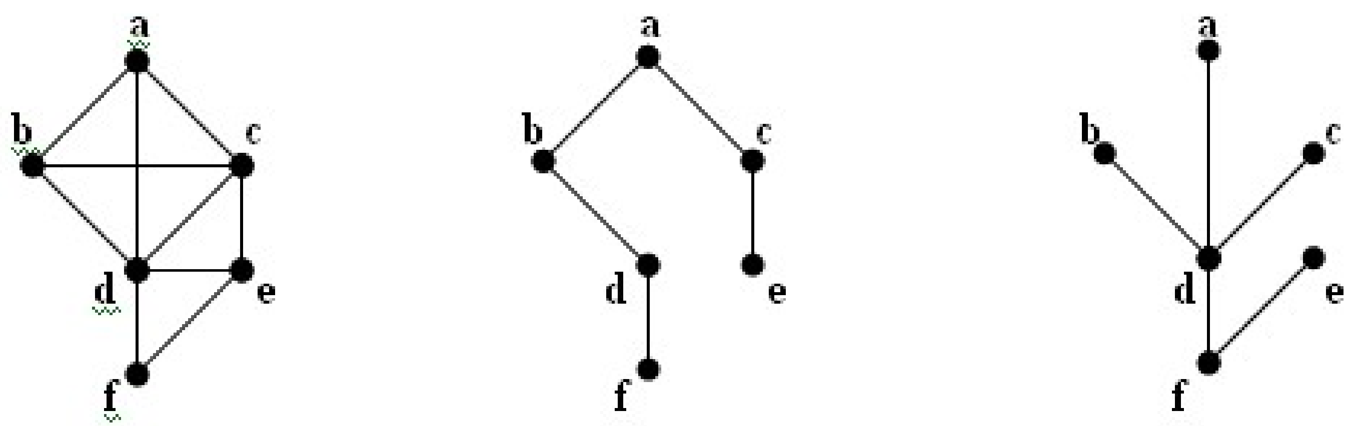
Theorems

- A simple nondirected graph G is a tree iff G is connected and contains no cycles.
- In every nontrivial tree there is at least one vertex of degree 1.
- A tree with n vertices has exactly n-1 edges.
- If G is a nontrivial tree then G contains at least 2 vertices of degree 1.
- If 2 nonadjacent vertices of a tree T are connected by adding an edge , then the resulting graph will contain a cycle.
- A graph G is a tree if and only if G has no cycles and $|E| = |V| - 1$.

Spanning Trees

A ***Spanning tree*** **of a connected graph G is a subgraph of the connected graph, which is a tree and contains all vertices of the connected graph.**

Example:



- In general, if G is a connected graph with n vertices and m edges, a spanning tree of G must have n – 1 edges.
- Hence, the number of edges must be removed before a spanning tree is obtained must be m-(n-1)=m-n+1.
- This number is frequently called the circuit rank of G.
- Theorem: A nondirected graph G s connected if and only if G contains a spanning tree. If we successively delete edges of cycles until no further cycles remain, then the result s a spanning tree of G

<u>Planar Graphs:</u>

A *planar graph* is a graph G that can be drawn on a plane without crossovers. Otherwise G is said to be *non planar*.

(or)

A graph which can be represented by at least on plane drawing in which the edges meet only at the vertices

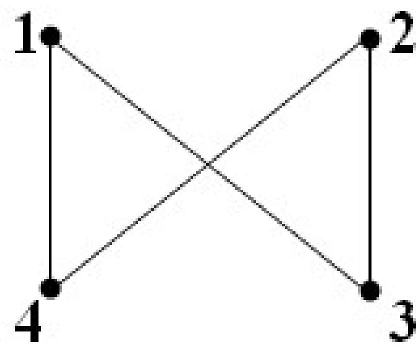
<u>Plane Graph:</u>

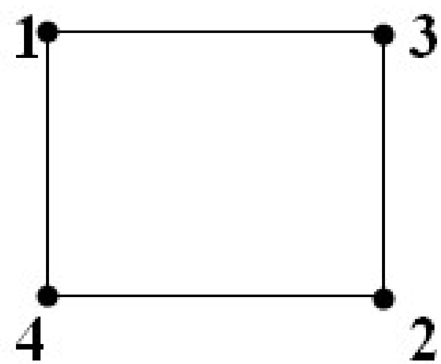
If a planar graph is drawn in the plane so that no two edges cross over.

Note:

1. Order means no. of nodes in the graph.
2. Size   means no. of edges in the graph

Example

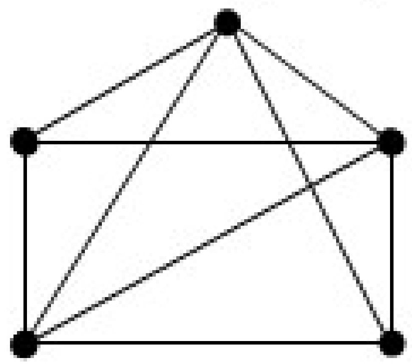




Plane Graph

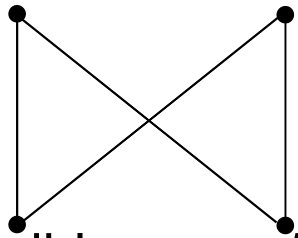
Exercises

1. Show that the graph given below is a *planar*.

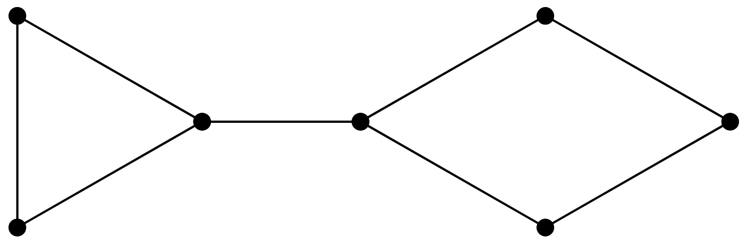


Exercises

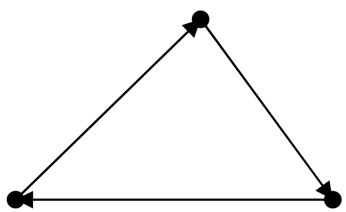
1. Derive all possible *spanning trees* for the *graph* shown below.



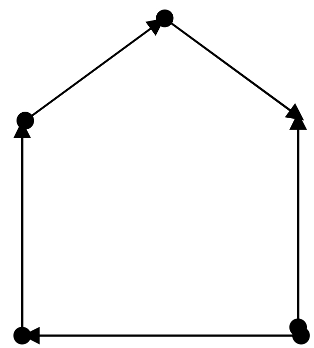
2. Derive all possible *spanning trees* for the *graph* shown below



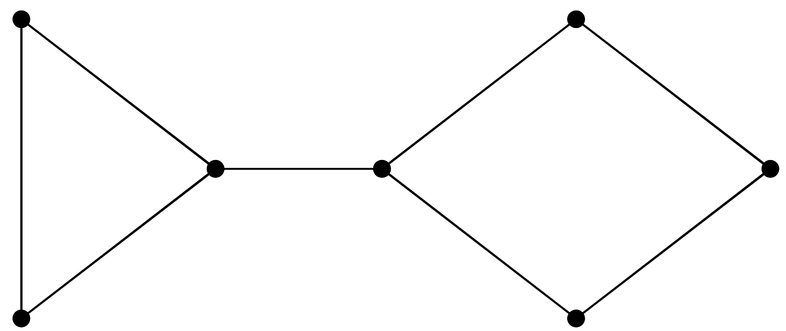
3. For the graph below, give all spanning trees.



4. For the graph below, how many different spanning trees are there?



5. Find DFS and BFS spanning trees for the graph given below.



Isomorphism

Two graphs G and G' are said to be isomorphic (to each other) if there is a one-to-one correspondence between their vertices and between their edges such that the adjacency of vertices is preserved.

Isomorphic graphs will have same structure, differing only in the way their vertices and edges are labeled or only in the way they are represented geometrically.

$G \cong G'$.

Isomorphic graphs must have:

1. same no. of vertices.
2. same no. of edges.
3. equal no. of vertices with a given degree.

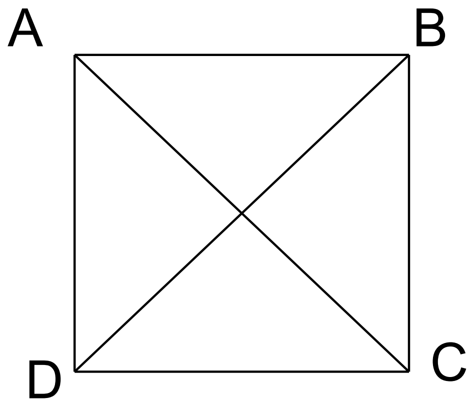


Fig: X

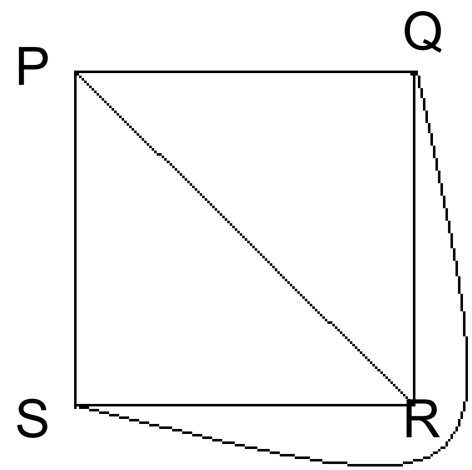


Fig: Y

No. of vertices : In X = 4 and in Y = 4
No. of Edges : In X = 6 and in Y = 6

Degrees of the vertices are as follows:

Degree	X	Y
0	0	0
1	0	0
2	0	0
3	4	4
4	0	0

one-to-one correspondence between the vertices of the graphs X and y are as follows:

$A \leftrightarrow P$, $B \leftrightarrow Q$, $C \leftrightarrow R$, $D \leftrightarrow S$

Similarly one-to-one correspondence between the edges of the graphs X and y are as follows:

$\{A,B\} \leftrightarrow \{P,Q\}$,
$\{B,C\} \leftrightarrow \{Q,R\}$,
$\{C,D\} \leftrightarrow \{R,S\}$,
$\{D,A\} \leftrightarrow \{S,P\}$,
$\{A,C\} \leftrightarrow \{P,R\}$,
$\{B,D\} \leftrightarrow \{Q,S\}$

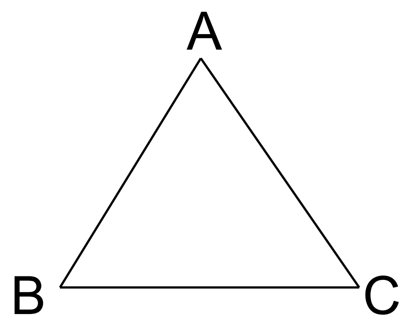


Fig: X

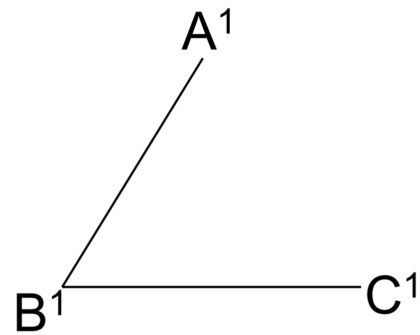


Fig: Y

No. of vertices : In X = 3 and in Y = 3

No. of Edges : In X = 3 and in Y = 2

Degrees of the vertices are as follows:

Degree	X	Y
0	0	0
1	0	2
2	3	1
3	0	0
4	0	0

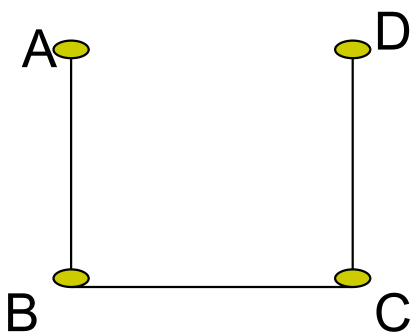


Fig: X

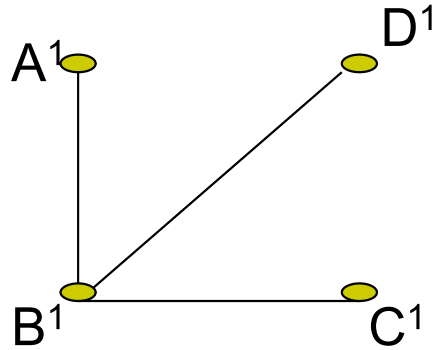


Fig: Y

No. of vertices : In X = 4 and in Y = 4
No. of Edges : In X = 3 and in Y = 3

Degrees of the vertices are as follows:

Degree	X	Y
0	0	0
1	2	3
2	2	0
3	0	1
4	0	0

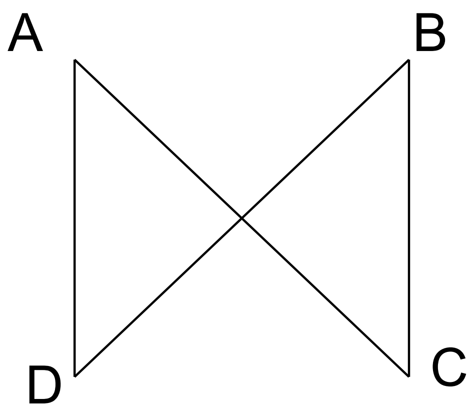


Fig: X

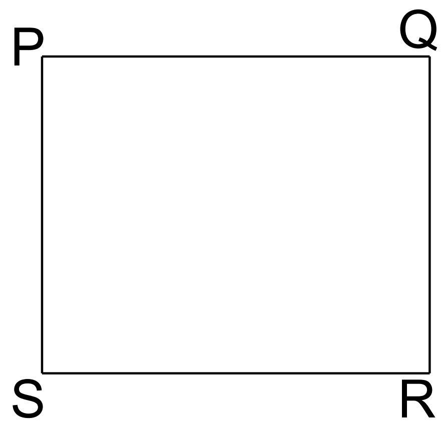


Fig: Y

one-to-one correspondence between the vertices of the graphs X and y are as follows:

$$A \leftrightarrow P,\ B \leftrightarrow R\ ,\ \ C \leftrightarrow Q\ ,\ \ D \leftrightarrow S$$

Similarly one-to-one correspondence between the edges of the graphs X and y are as follows:

$\{A,C\} \leftrightarrow \{P,Q\}$,
$\{B,C\} \leftrightarrow \{Q,R\}$,
$\{B,D\} \leftrightarrow \{R,S\}$,
$\{D,A\} \leftrightarrow \{S,P\}$,

Null / Discrete graph

N_n

Graph with n vertices and no edges

Disconnected

Exactly n components.

Examples:

N_2 N_4

Complete graph

K_n

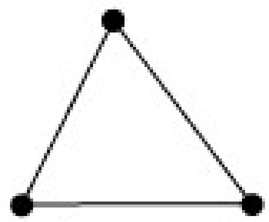
An edge $\{v_i, v_j\}$ for every i and j

Every vertex is *connected* to every other vertex

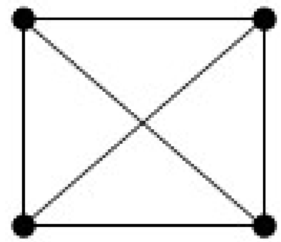
Regular graph

Each vertex of a graph has the same degree as every other vertex

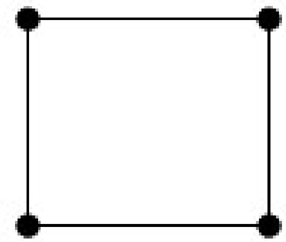
Examples:



K_3



K_4



linear graph

L_n

Edges $\{v_i, v_{i+1}\}$ for $1 \leq i \leq n$

Connected

Examples:

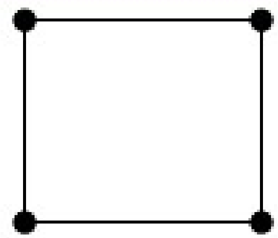
L_2

L_4

Cycle graph

C_n

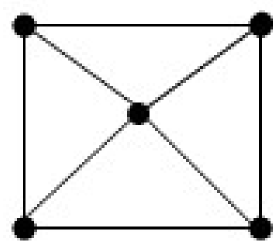
Connected graph whose edges form a cycle.



Wheel graph

W_n

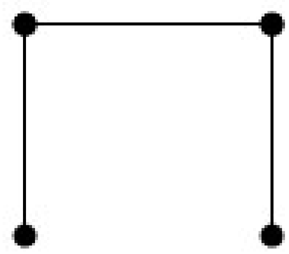
Graph obtained by joining a single new vertex (hub) to each vertex of a cycle graph.



Path graph

P_n

Graph obtained by removing an edge from a cycle graph.



Empty graph

Graph with no vertices and no edges.

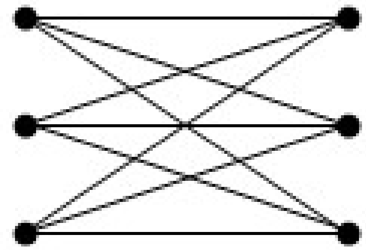
Bipartite graph

Nondirected graph whose set of vertices can be partitioned into two sets M and N in such a way that each edge joins a vertex in M to a vertex in N.

Complete Bipartite graph

$K_{m,n}$

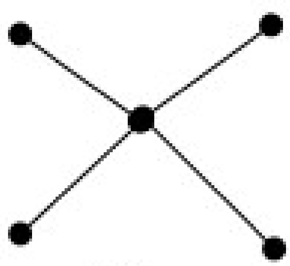
Bipartite graph in which every vertex of M is adjacent to every vertex of N.



$K_{3,3}$

Star graph

$K_{1,n}$ graph



$K_{1,4}$

Subgraphs

Graph $G = (V, E, \gamma)$

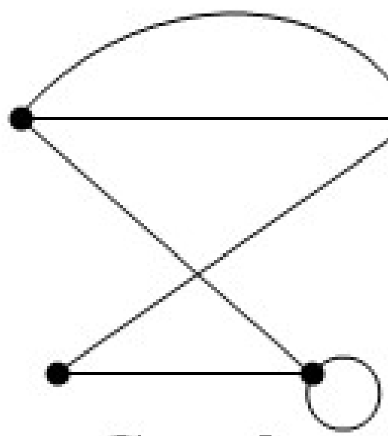
Choose a subset E_1 of the edges in E

and a subset V_1 of the vertices in V

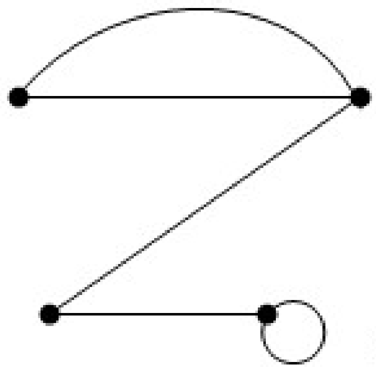
Subgraph $H = (V_1, E_1, \gamma_1)$

γ_1 $\quad$ γ restricted to edges in E_1

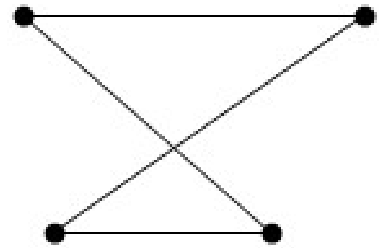
Example:



Graph



Subgraphs



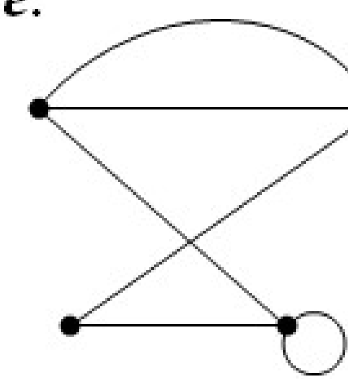
Graph $G = (V, E, \gamma)$

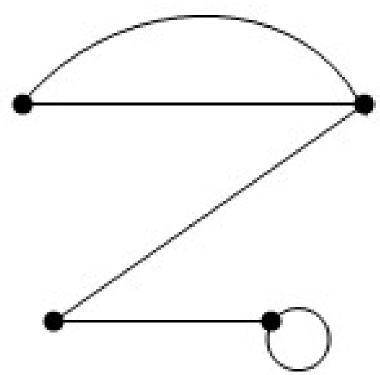
e $\in E$

$G_{\mathbf{e}}$

> **Subgraph obtained by omitting the edge e from** E **and keeping all vertices.**

Example:





<u>Euler Circuit and Euler Trails</u>:

consider a graph G, and if there is a circuit in G that contains all the edges of G, then that circuit is called Euler Circuit

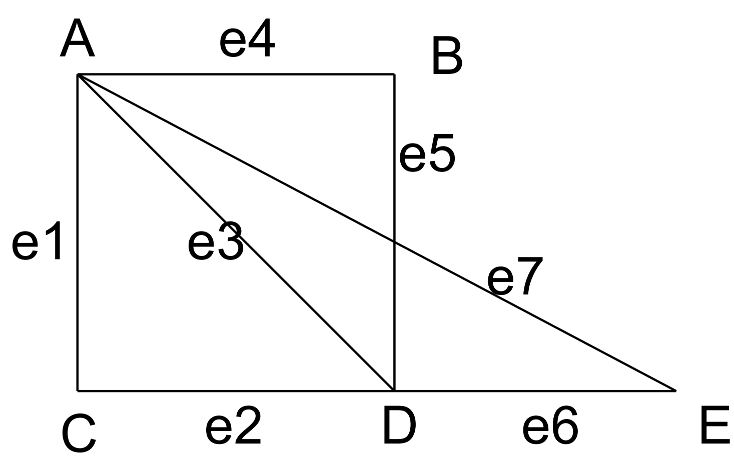
(***Circuit***

Path that begins and ends at the same vertex.

in a circuit no edge can appear more than once but the vertex can appear more than once)

A connected graph that contains a Euler circuit is called **Euler Graph**

Example:

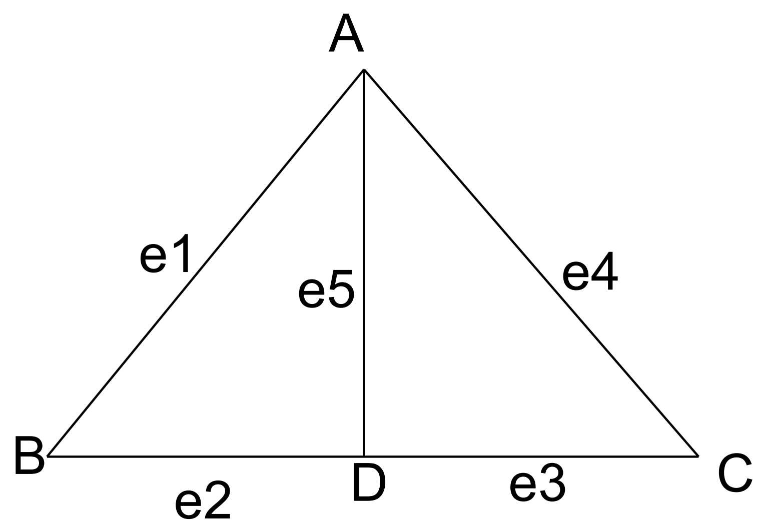


The following walk is Euler circuit:

A e1 C e2 D e3 A e4 B e5 D e6 E e7 A

Euler Trails:

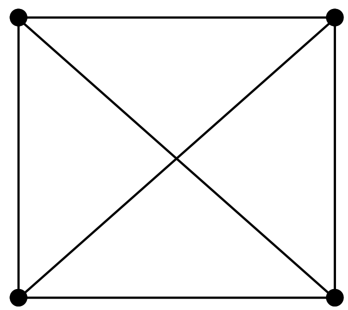
if there is a trail in G that contains all the edges of G, then that trail is called Euler trail.

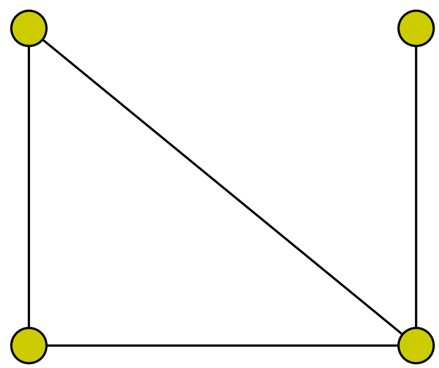


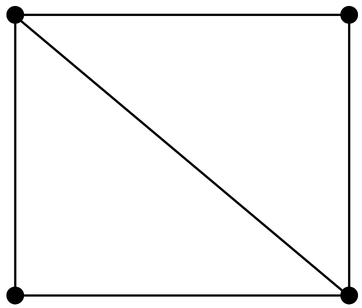
The following is the trail in the graph:
A e1 B e2 D e3 C e4 A e5 D

Note:

1. **If a graph *G* has a vertex of odd degree, there can be no Euler circuit in *G*.**
2. **If *G* is connected and has exactly two vertices of odd degree, there is an Euler path in *G* but not Euler circuit.**
   Any Euler path in *G* must begin at one vertex of odd degree and end at the other.
3. **If a graph *G* has more than two vertices of odd degree, then there can be no Euler path in *G*.**

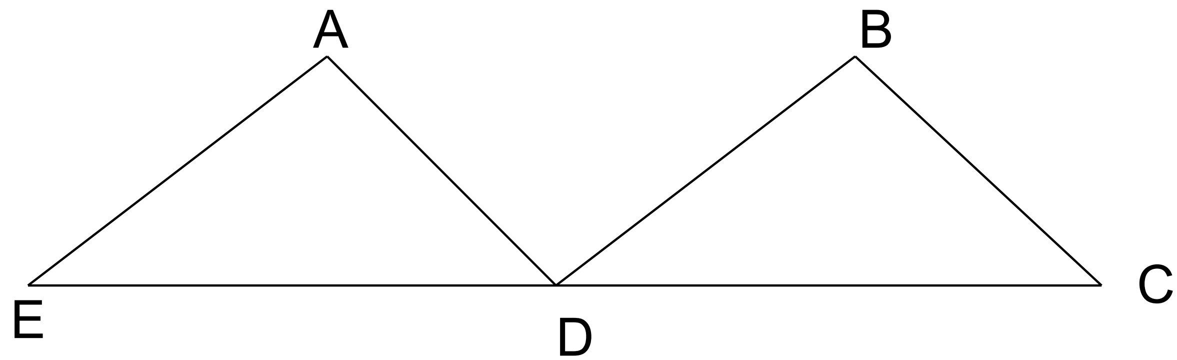


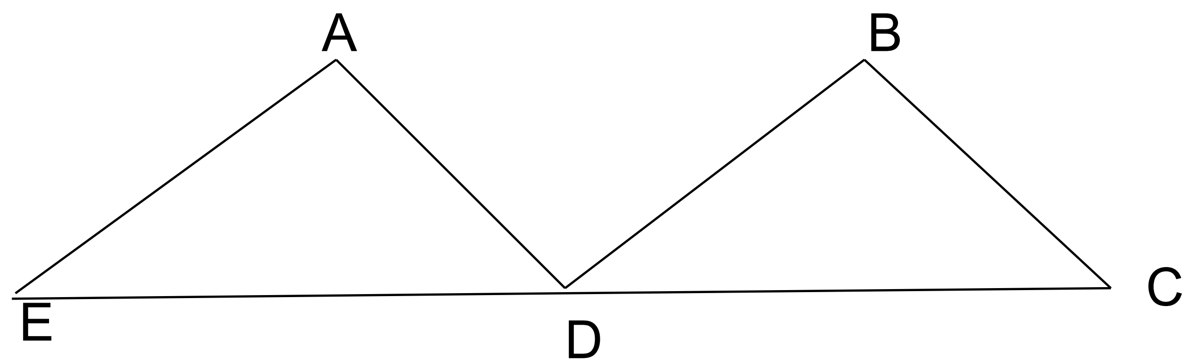




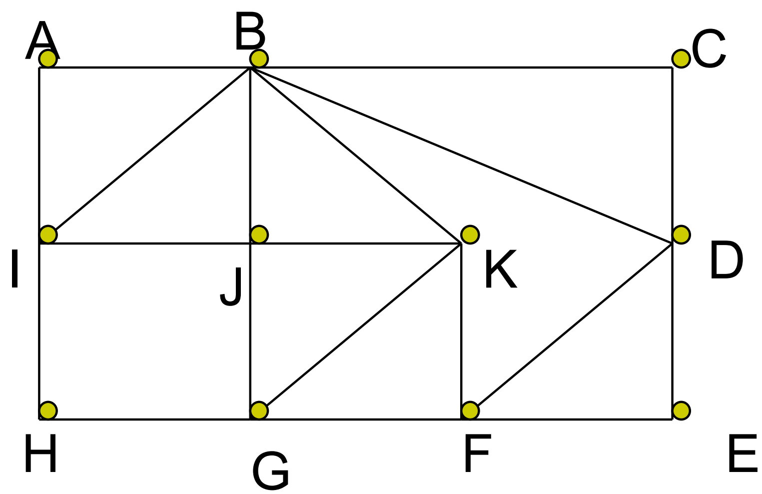
The graph has exactly two vertices of odd degree.
Hence there is no Euler circuit, but there must be an Euler path.

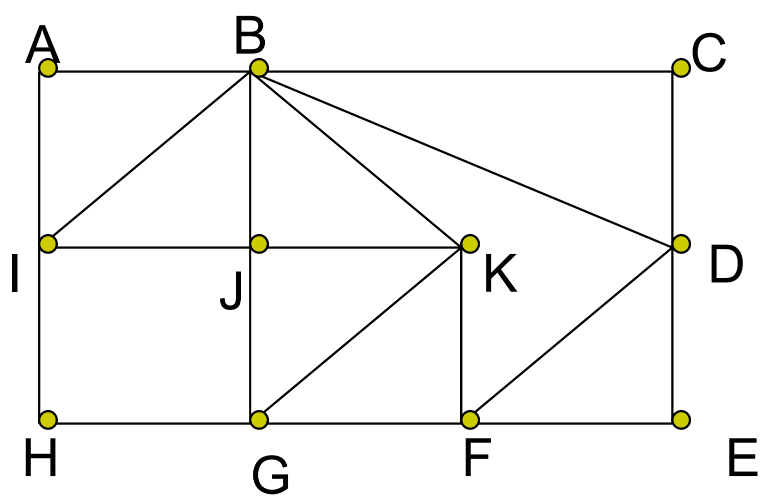
1)





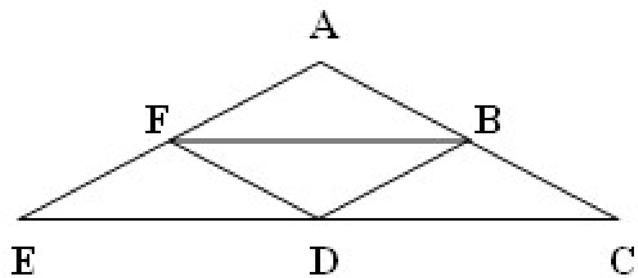
2)



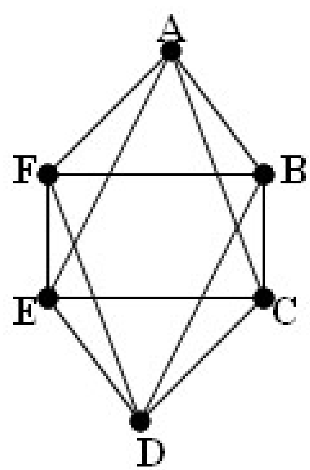


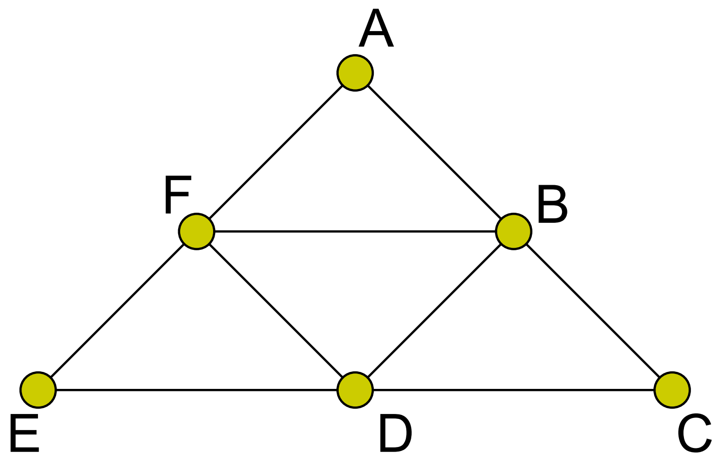
Examples

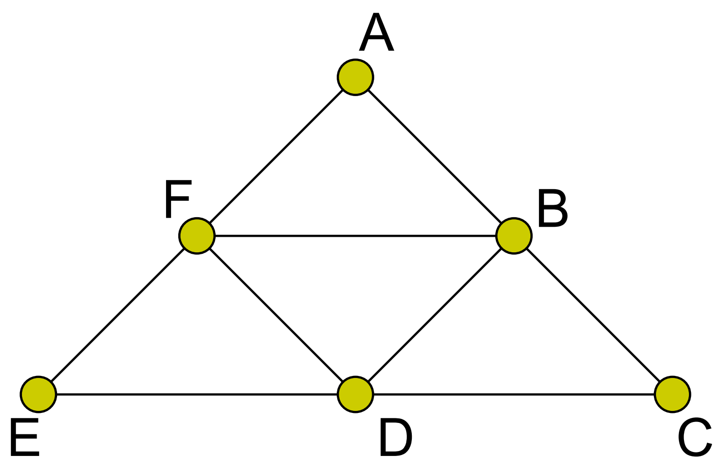
1. Give the *Euler circuit*, by giving the edges visited in the sequence, for the graph represented below, by beginning at F.

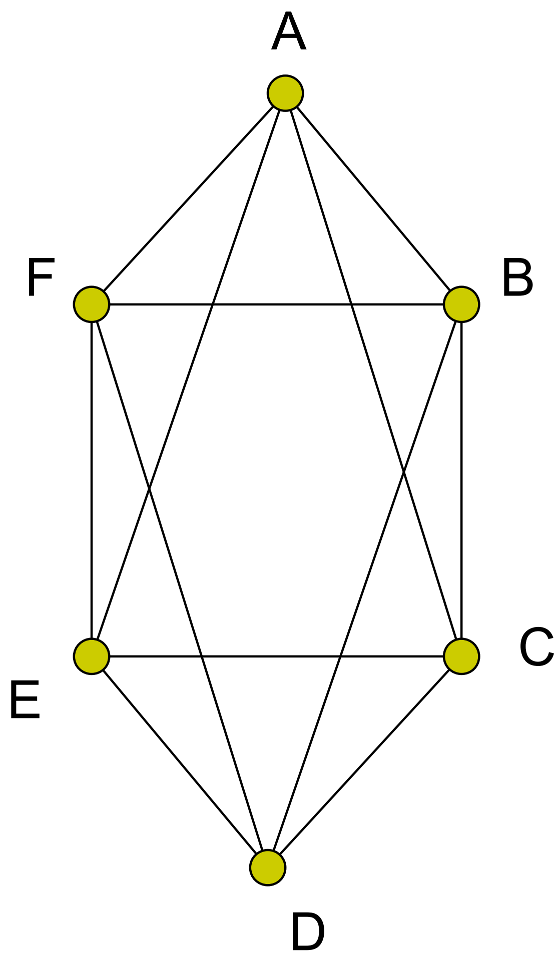


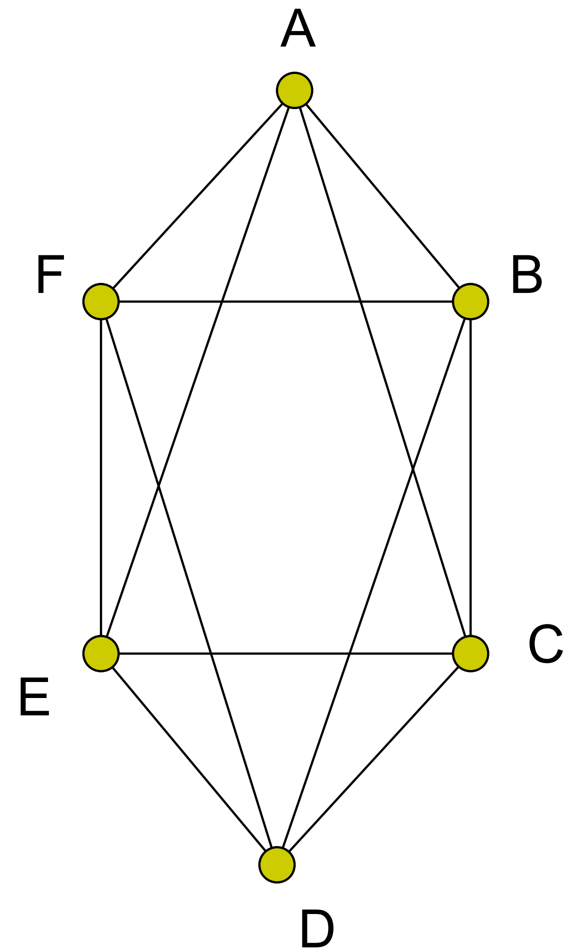
2. Give the *Euler circuit*, by giving the edges visited in the sequence, for the graph represented below, beginning at C.




A
F
B
E
D
C


A
F
B
E
D
C


A
F
B
E
C
D


A
F
B
E
C
D

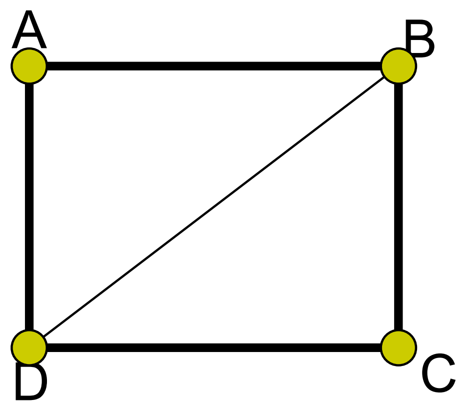
Hamilton cycles and Hamilton paths:

Let G be a connected graph, if there is a cycle in G that contains ***all vertices*** of G, then that cycle is called a **Hamilton cycle** in G

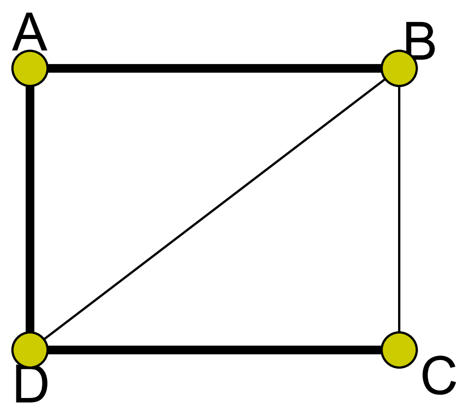
1. Each vertex must be visited only once except the first (starting) for cycle
2. A Hamilton cycle in a graph of n vertices consists of exactly n edges
3. Hamilton cycle may not include all edges of a graph

A graph that contains a Hamilton cycle is called a **Hamilton graph**

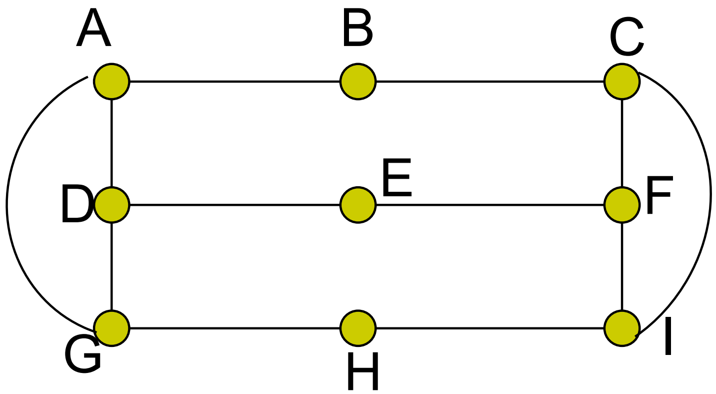
A path in a connected graph which includes every vertex of the graph is called **Hamilton path**.



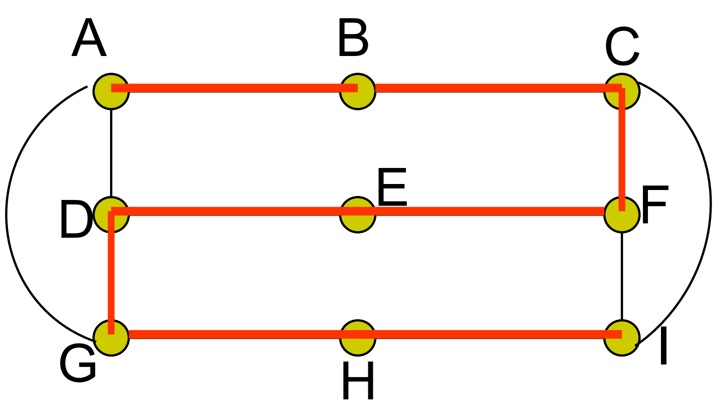
In the above graph, the cycle is shown in thick lines is Hamilton cycle



In the above graph, the path shown in thick lines is Hamilton path



In the above graph the Hamilton path is ABCFEDGHI. And there is no Hamilton Cycle as shown below



Graph Coloring and Chromatic Numbers:

Graph coloring:

For a given graph G, if we assign colors to its vertices in such a way that no two adjacent vertices have the same color, then the graph is called *properly colored* graph.

Note:

1. A graph can have more than one proper coloring
2. Two non-adjacent vertices in a properly colored graph can have the same color

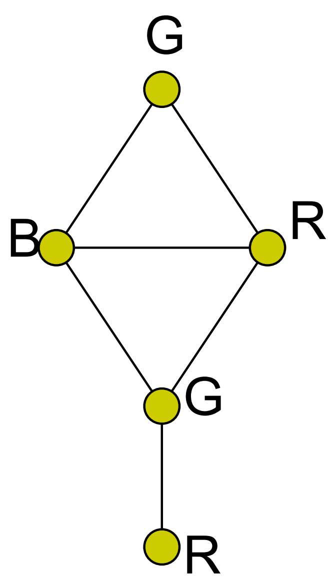


Fig: 1

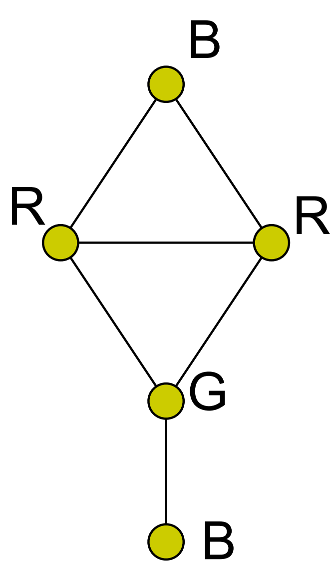


Fig: 2

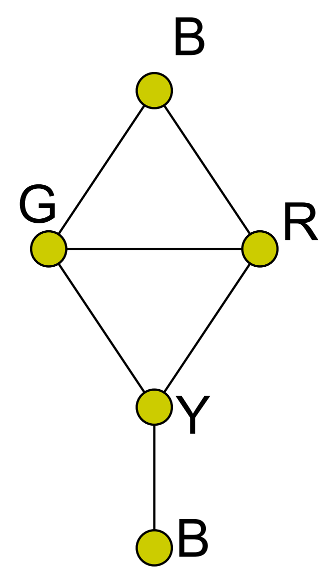


Fig: 3

Chromatic Numbers:

The chromatic number of a graph is the minimum number of colors with which the graph can be properly colored

OR

A graph G is said to be k-colorable if we can properly color it with k colors
A graph G which is k-colorable but not (k-1) colorable is called a k-chromatic graph
A k-chromatic graph is a graph that can be properly colored with k colors but not with less than k colors

If a graph G is k-chromatic, then k is called the *chromatic number* of G

Trees

A tree is a simple graph G such that there is a unique simple nondirected path between each pair of vertices of G.

A rooted tree is a tree in which there is one designated vertex, called a root

A rooted tree is a directed tree if there is a root from which there is a directed path to each vertex.

The level of a vertex v in a rooted tree is the length of the path to v from the root.

A tree T with only one vertex is called a trivial tree ; otherwise T is a nontrivial tree.

Trees

If a graph G is connected and e is an edge such that G-e is not connected, then e is said to be a bridge or a cut edge

If v is a vertex of G such that G-v is not connected , then v is a cut vertex.

Theorems:

1. A simple nondirected graph G is a tree iff G is connected and contains no cycles.
2. In every non trivial tree there is at least one vertex of degree 1.
3. A tree with n vertices has exactly n-1 edges.
4. If 2 non adjacent vertices of a tree T are connected by adding an edge, then the resulting graph will contain a cycle.
5. A graph G is a tree if and only if G has no cycles and $|E| = |V| - 1$.

Spanning Trees

A sub graph H of a graph G is called a spanning tree of G if

1. H is a tree and
2. H contains all the vertices of G.

A spanning tree that is a directed tree is called a directed spanning tree of G.

In general, if G is a connected graph with n vertices and m edges, a spanning tree of G must have n-1 edges.

Theorems:

1. A nondirected graph G is connected if and only if G contains a spanning tree. Indeed, if we successively delete edges of cycles until no further cycles remain, then the result is a spanning tree of G.

- BFS: visit all vertices sequentially on a given level before going onto the next level.

  →The edges that are added in the bfs are called tree edges, and rejected edges called the cross edges.

- DFS: proceeds successively to higher levels at the first opportunity.

  →Rejected edge called the back edge.

  →In general, dfs terminates when the search returns to the root and all vertices have been visited.

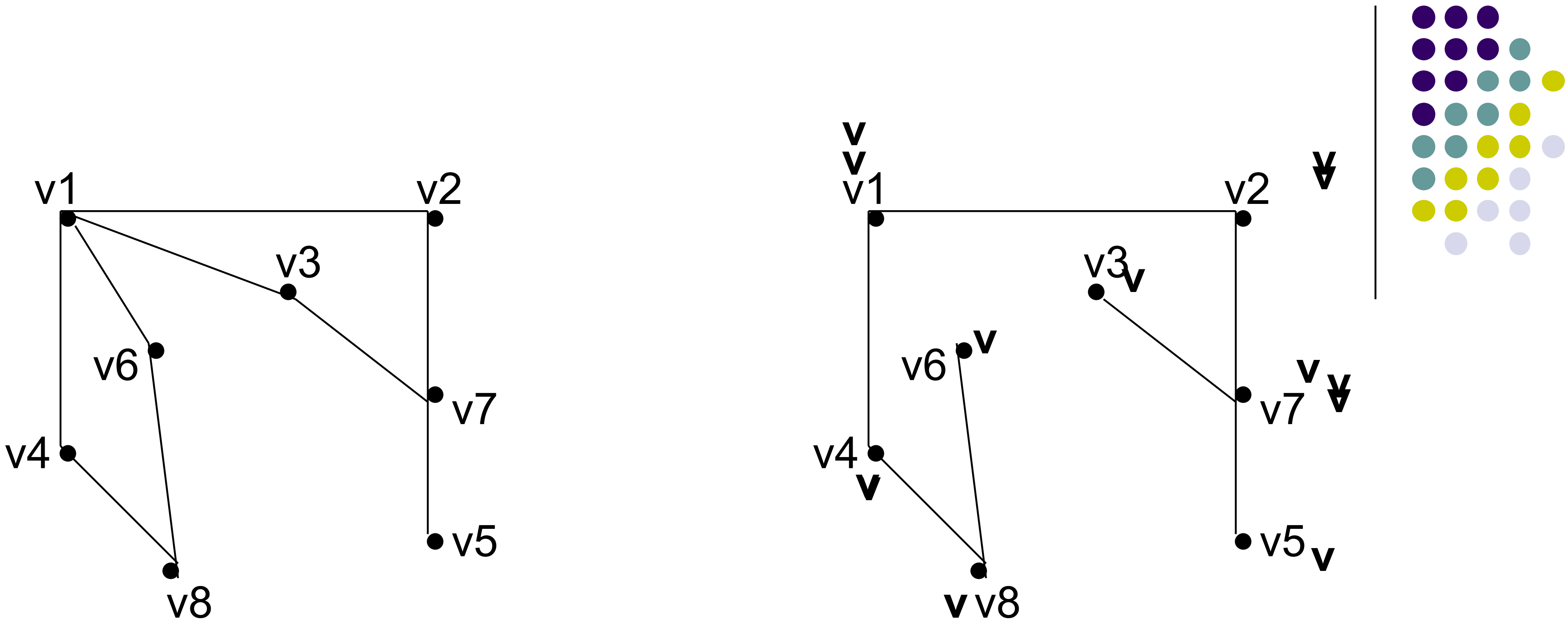
Depth First Search (DFS) algorithm:

1. Assign the first vertex $v1$ to the variable v and initialize T as the tree consisting of this vertex as the root.

2. Select the smallest subscript k for $2 \leq k \leq n$, such that $\{v, vk\} \in E$ and vk has not already been included in T.
   If no such subscript is found then go to step 3, otherwise perform the following:
   a. Attach the edge $\{v, vk\}$ to T.
   b. Assign vk to v.
   c. Repeat step 2.

3. If $v = v1$, the tree T is the spanning tree for the order specified.

4. For $v \neq v1$, backtrack from v. If u is the parent of the vertex assigned to v in T, then assign u to v and repeat step 2.

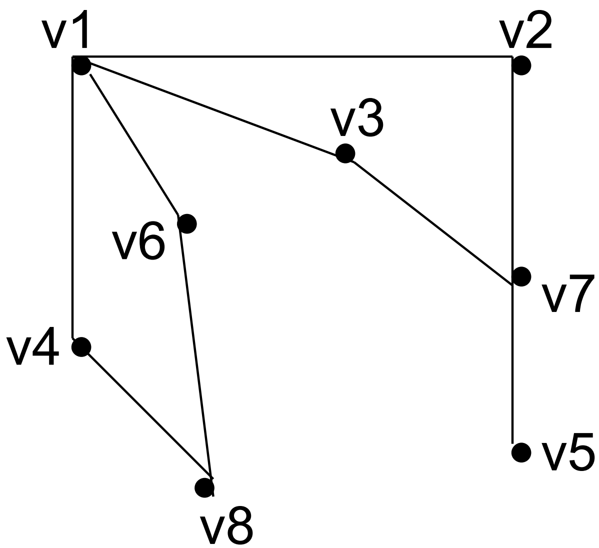

v1
v2
v3
v6
v7
v4
v5
v8
v1
v2
v3
v6
v7
v4
v5
v8

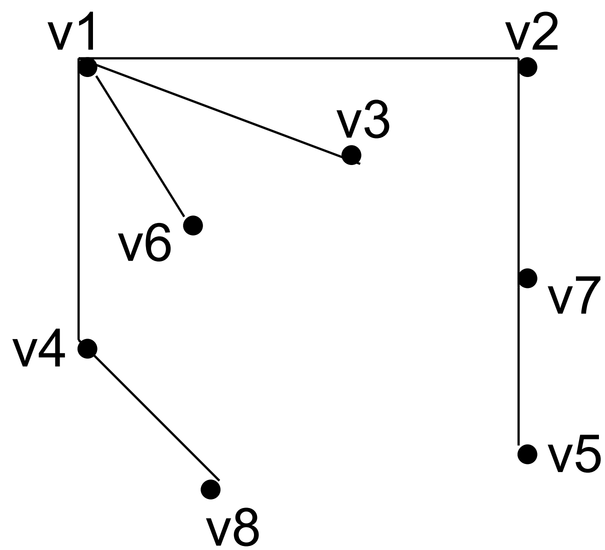
Breadth First Search (BFS) algorithm

1. Assign the first vertex v1 and insert this vertex in the queue Q and initialise T as the tree consisting of this vertex as the root.

2. Delete v from the front of Q. When v is deleted, consider vk for each $2 \leq k \leq n$. If the edge $\{v, vk\} \in E$ and vk has not been visited previously, attach this edge to T. If we examine all of the vertices previously visited and obtain no new edge, the tree T is the desired spanning tree.

3. Insert the vertices adjacent to each v at the rear of the queue Q, according to the order in which they are (first) visited. Repeat step 2.


v1
v2
v3
v6
v7
v4
v5
v8


v1
v2
v3
v6
v7
v4
v5
v8

Planar Graphs:

A *planar graph* is a graph G that can be drawn on a plane without crossovers. Otherwise G is said to be *non planar*.

(or)

A graph which can be represented by at least on plane drawing in which the edges meet only at the vertices

Plane Graph:

If a planar graph is drawn in the plane so that no two edges cross over.

Note:

1. Order means no. of nodes in the graph.
2. Size   means no. of edges in the graph

Euler's formula for connected plane graph

If G is a connected plane graph, then $|V| - |E| + |R| = 2$.

where

V No. of vertices

E No. of edges

R No. of regions

Proof (by mathematical induction):

Assume that Connected plane graph G has $k + 1$ regions, where $k \geq 1$.

Delete an edge common to the boundary of two separate regions.

The resulting graph G' has the same no. of vertices, one fewer edge and also one fewer region as two previous regions have been consolidated by the removal of the edge.

- No. of vertices in G' = $|V'| = |V|$.
- No. of edges in G' = $|E'| = |E| - 1$.
- No. of regions in G' = $|R'| = |R| - 1$.

But $|V| - |E| + |R| = |V'| - |E'| + |R'|$.

$|V'| - |E'| + |R'| = 2$, by inductive hypothesis.

$\therefore\ |V| - |E| + |R| = 2$.

Dual of a plane graph

- Corresponding to each region r of G there is a vertex r* of G*, and corresponding to each edge e of G there is an edge e* of G*; two vertices r* and s* are joined by the edge e* in G* iff their corresponding regions r and s are separated by the edge e in G.
- A loop is added at a vertex r* of G* for each cut–edge of G that belongs to the boundary of the region r.

Dual of a plane graph

- Corresponding to each region r of G there is a vertex r* of G*, and corresponding to each edge e of G there is an edge e* of G*; two vertices r* and s* are joined by the edge e* in G* iff their corresponding regions r and s are separated by the edge e in G.
- A loop is added at a vertex r* of G* for each cut–edge of G that belongs to the boundary of the region r.