TREE

TREE

(UNIT – II)

K. Maheswaran MCA, M.Phil., M.Sc (Maths), SET, Ph. D

Assistant Professor in Computer Science

School of Computing Sciences

St. Joseph's College(Autonomous)

Tiruchirappalli- 620 002

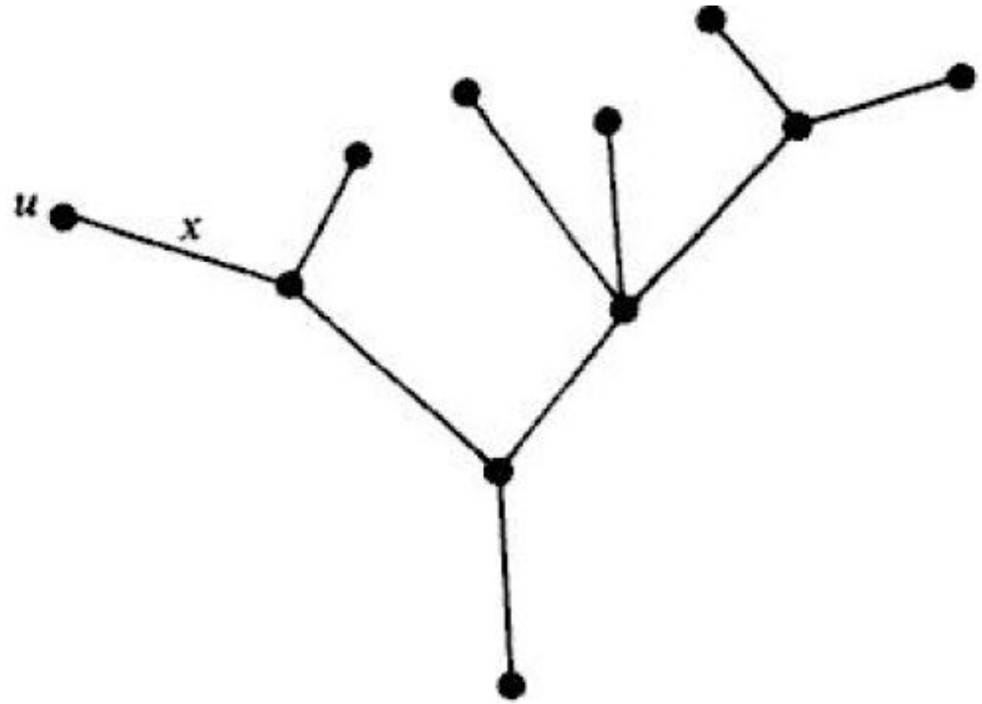
TREE

Definition : (<u>5-diferent but equivalent definition ofTree.)</u>

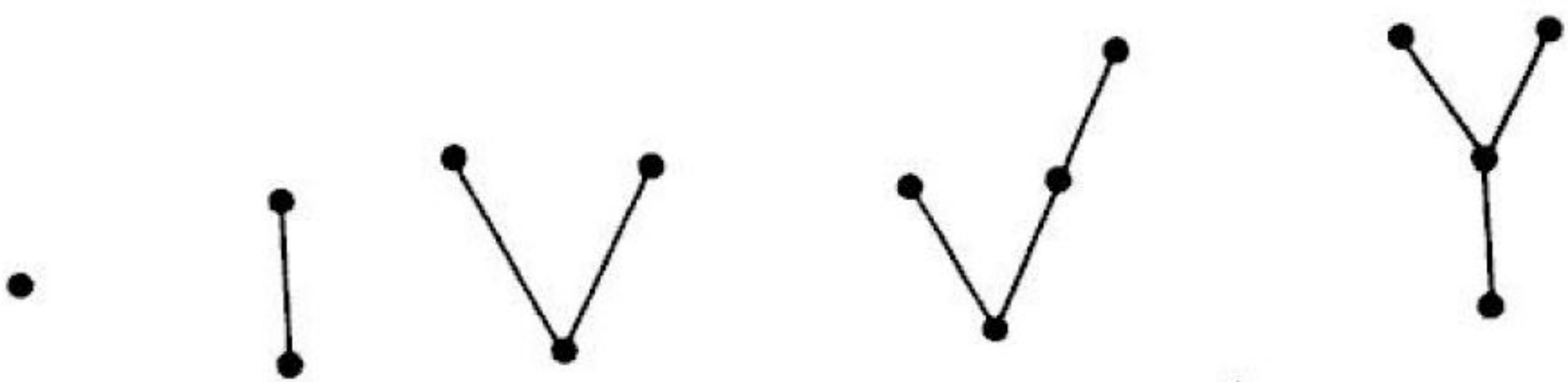
1. Tree is a connected graph without any circuits.

   i.e. G is connected and circuit less.

2. G is connected and has (n-1) edges.
3. G is circuit less and has (n-1) edges.
4. There is exactly one path between every pair of vertices.
5. G is minimally connected graph.



Tree.



Trees with one, two, three, and four vertices.

Theorem 1:

There is one and only 1 path between every pair of vertices in a tree T.

Proof:

Given T is a Tree.

i.e. 'T' is connected and circuit less.

Since T is a connected graph . There must exist at least 1-path between every pair of vertices in "T".

Suppose there are two distinct path between two pair of vertices a and b.

Then the union of these two paths will create a circuit.

which is ➔🡐 (Contradiction) to circuit less.

Therefore more 1-path between two vertices is not possible.

∴ There is one and only one path between every pair of vertices in 'T'.

Theorem 2:

If in a Graph G. There is one and only one path between every pair of vertices. Then G is a Tree.

Proof:

To prove: G is a Tree

i.e. G is a connected and circuit less.

i) Existence of a path between every pair of vertices assures that G is connected.

ii) A circuit in a Graph, implies that there is at least one pair of vertices a and b. Such that there are two distinct paths between a and b.

Since G has one and only path between every pair of vertices.

Therefore G is circuit less.

∴ G is Tree.

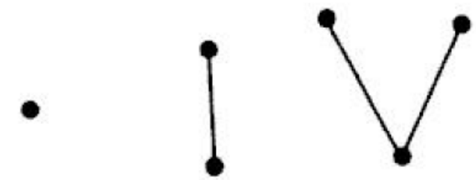
Theorem 3:

A Tree with ‘n’ – vertices has (n-1) edges.

Proof:

This theorem is proved by induction on the number of vertices.

It’s easy to prove that this result is true for n=1, n=2, n=3.



Assume that the theorem is true for all trees,

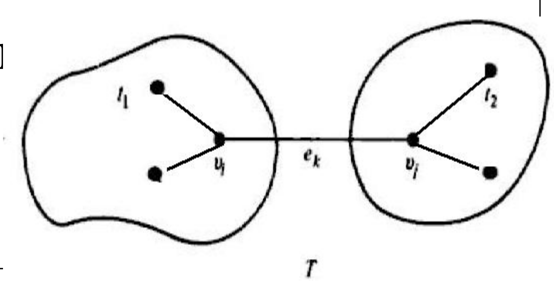
fewer than n-vertices, Now we have to prove this theorem for a tree with n-vertices.

Let us consider a tree T with n-vertices.

In ‘T’ , e_k be an edge with between vertices v_i and v_j

Since there is only one path between every pai

no other path between vertices v_i and v_j except e_k



Now , we remove e_k in 'T', then removal e_k will disconnect 'T' into T_1 and T_2, at shown in the figure.

Let n_1 be the number of vertices in T_1.

Let n_2 be the number of vertices in T_2.

where $n_1 < n$, $n_2 < n$

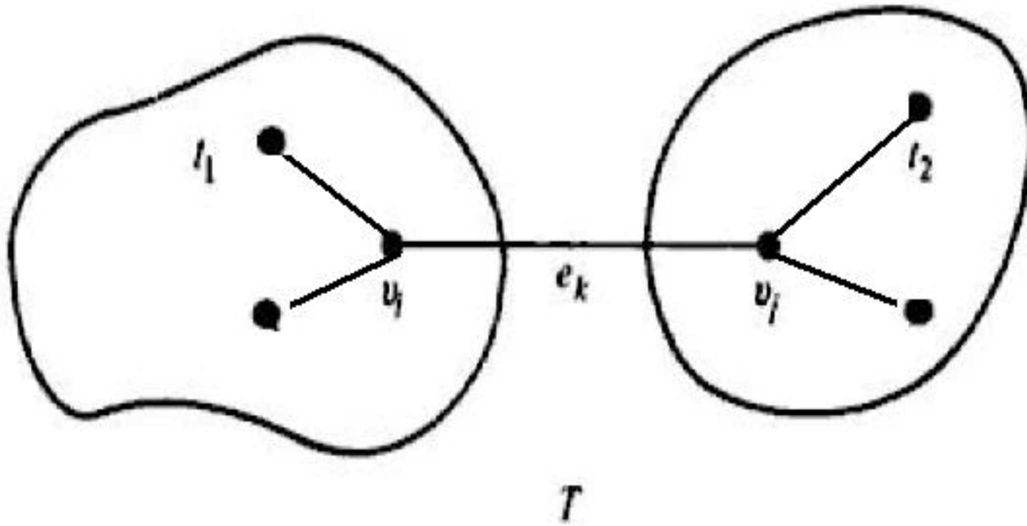
Therefore $n = n_1 + n_2$.

Here T_1 and T_2 are Trees.

Now by Induction hypothesis,

Number of edges in T_1 = **n_1-1**.

Number of edges in T_2 = **n_2-1**.



Therefore Number of edges in $T - e_k$ is = $n_1 - 1 + n_2 - 1$.

Therefore number of edges in 'T' $= (n_1 - 1) + (n_2 - 1) + 1$

$= (n_1 + n_2) - 1$

Therefore number of edges in 'T' $= (n-1)$ edges.

Hence it is proved.

Theorem 4:

A Graph G with n-vertices, (n-1) edges and no circuits is connected.

Proof:

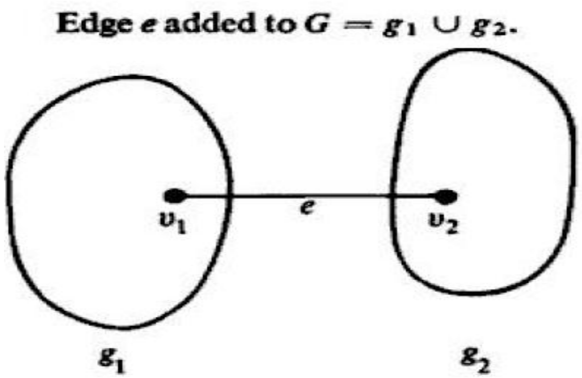
Suppose G has n-vertices (n-1) edges no circuits is disconnected.

In this case, G with consists of 2 or more circuit less component.

Without loss of generality,

Let G consist of g_1 and g_2(has two components)

add an edge 'e' between v_1 in g_1 and v_2 in g_2.



since there is no path between v_1 and v_2 in G,

adding 'e' did not create a circuit.

Thus G U E is circuit less, connected graph with n-vertices n-edges.

Which is not possible because of the theorem, "A Tree with n-vertices has (n-1) edges".

Therefore =><=. **Therefore G is connected.**

Theorem 5:

Any connected graph with n-vertices, (n-1) edges is Tree.

Proof:

Given: i) G is connected

ii) n- vertices and n-1 edges.

To Prove: G is a Tree i.e. G is a connected and circuit less.

Suppose G has a circuit, For two vertices a and b, in which there are 2 different paths between them & since G is connected, so no. of edges is greater than(>) n-1.

∴ ➔🡐 n-1 edges.

∴ G has no circuit.

G is connected and circuit less. (i.e) G is a Tree.

Definition:(Minimally Connected)

A connected G is said to be minimally connected if removal of any of edge from it disconnects the graph.

Theorem 6:

A graph is a tree **iff** it is minimally connected

Proof:

Part – I : G is a Tree i.e. G is a connected and circuit less.

Claim: G is minimally connected.

Suppose G is not minimally connected, there must be an edge e_i in G such that $G-e_i$ is connected.

i.e e_i is in some circuit.

∴ => <= to G is circuit less.

G is minimally connected

Part – II:

Given G is minimally connected, which means that G is connected .

Claim: G is circuitless

Suppose G has circuit, then we could remove one of the edges in the circuit and still leave the graph connected.

=> <= to G is minimally connected.

∴ G is Circuit less.

∴ G is a Tree

Hence the theorem.

Pendant vertices in a Tree

Theorem 7:

In any Tree (with 2 or more vertices) There are at least 2-pendant vertices.

Proof:

WKT, A Tree with n-vertices has (n-1) edges.

$$\sum d(v) = 2e$$
$$= 2(n-1)$$
$$= 2n-2.$$

Here 2n-2 degree to be divided among n-vertices.

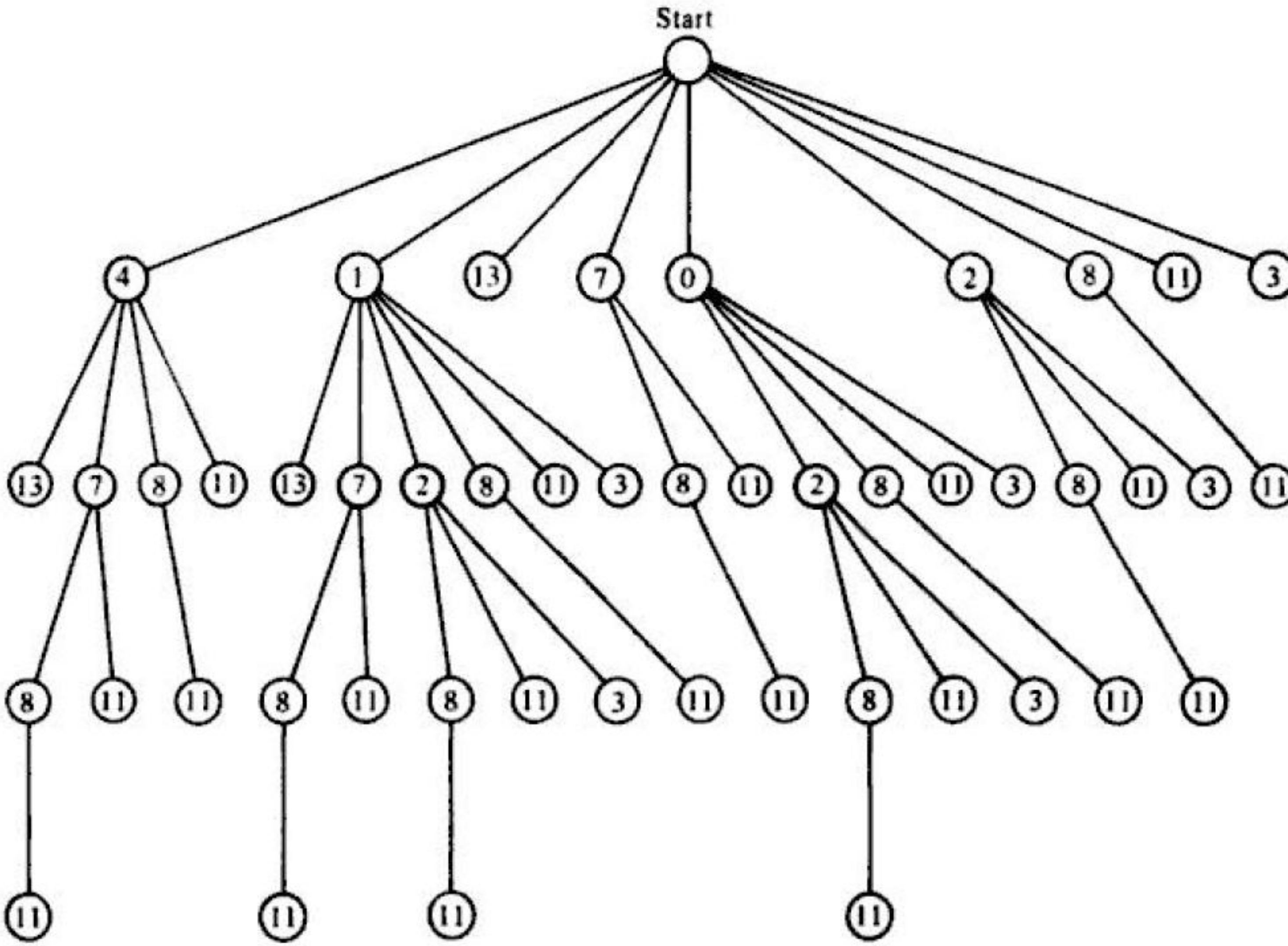
Since no-vertex have a zero degree, we must have at least two vertices of degree1 in a Tree.

Problem:

The given sequences integers 4, 1, 13, 7, 0, 2, 8, 11, 3 .

Find the largest monotonic increasing sun sequence in it.

Solution:



The given sequence contain 4-longest increasing subsequences in it.

(4, 7, 8, 11) (1 ,7, 8, 11) (1, 2, 8, 11) (0, 2, 8, 11)

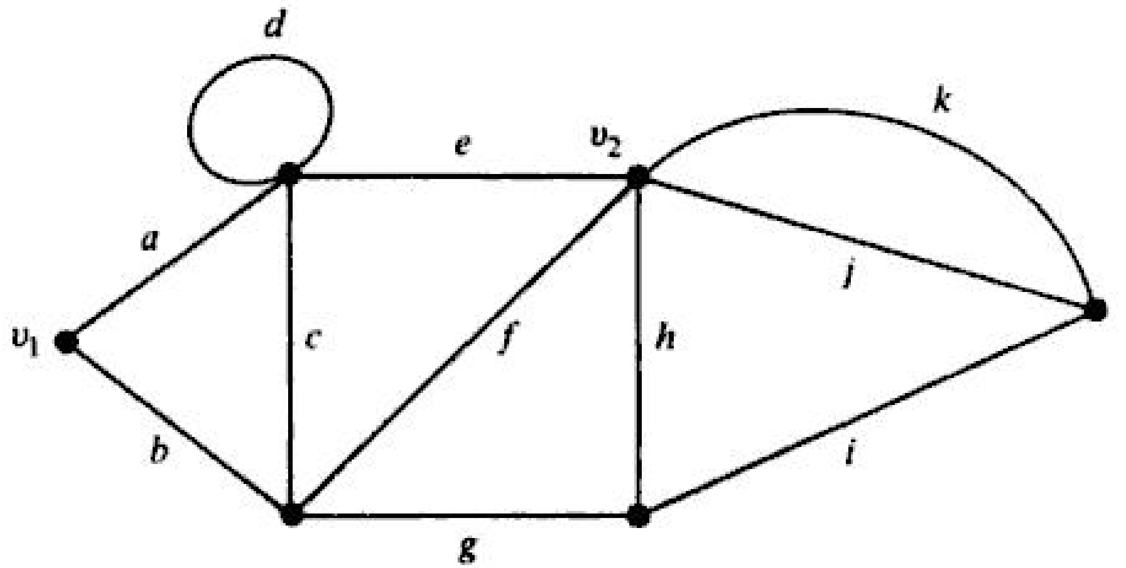
Each is of length four. Such a tree used in representing data is referred

s a data tree by computer programmers.

Definition (Distance):

In a connected graph G the distance d (v_i, v_j) between 2-vertices, v_i and v_j is the length of the Shortest path between them. (Number of edges in the shortest path.)

Note: In an tree there is only 1-path.



Distance between v_1 and v_2 is two.

paths between vertices v_1 and v_2

(a, e), (a, c, f), (b, c, e), (b, f), (b, g, h), and (b, g, i, k). There are two shortest paths, (a, e) and (b, f), each of length two. Hence $d(v_1, v_2) = 2$.

A Metric:

A function of 2-variable that satisfies following 3 condition is metric.

non-negative

i) $f(x,y) \geq 0$ and $f(x,y) = 0$ iff $x == y$.

Symmetric:

ii) $f(x,y) = f(y,x)$

Triangle Inequality:

iii) $f(x,y) \leq f(x,z)+f(z,y)$

Note:

RAT => Partial order relation.

RST => Equivalence relation.

Theorem 6:

The distance between vertices of a connected graph is metric.

Proof:

- $d(v_i,v_j) \geq 0$ and

  $d(v_i,v_j) = 0$ and $\Rightarrow v_i = v_j$.
- $d(v_i,v_j) = d(v_j,v_i)$
- $d(v_i,v_j) = d(vi,vk) + d(v_k,v_j)$

  3-Results are obvious.

Definition: (Eccentricity)

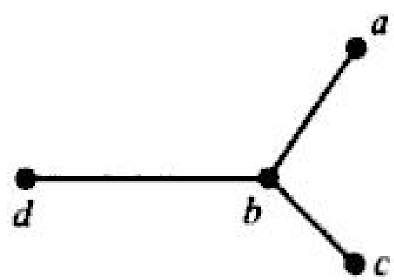
Eccentricity of a vertex is a maximum distance $d(v_i,v_j)$

$$E(v) = \max d(v_i,v_j)$$

An Eccentricity of a vertex 'v' of G is the distance between the vertex b to the vertex farthest.

Center:

A vertex with minimum Eccentricity in G is called center.



$E(a) = 2$, $E(b) = 1$, $E(c) = 2$, and $E(d) = 2$.

Mini. Eccentricity = 1

Therefore **Center is 'b'.**

Theorem 7:

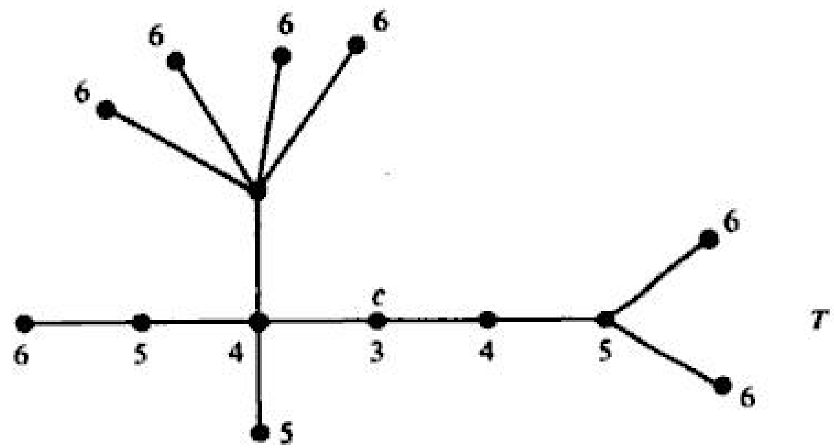
Every Tree has either 1 or 2 centers.

Proof:

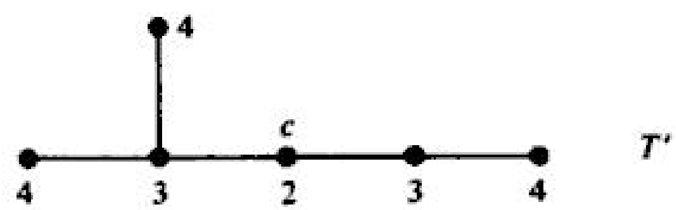
The maximum distance from a given vertex v_i to any other vertex occurs only when v_i is a pendant vertex. With this observation,

Let us start with a tree T having more than two vertices.

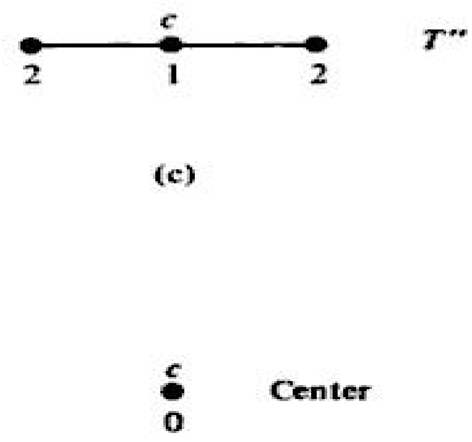
'T'-Tree must have 2 or more pendant vertices. Delete all pendant vertices from T.



(a)



(b)



The resulting graph T' is still Tree.

Eccentricity of all vertices in T' are reduced by 1.

So, all centers of T, will still remain centers in T'.

Now from T', we can again remove all pendant vertices. We can get a new Tree T".

Continue the process like this until get single vertex or edge.

If they get single vertex they graph has one center

If single edge they graph has 2 centers.

Hence it is proved.

Raduis of a Tree and Diameter of a Tree

The eccentricity of a center is called "**Radius of a Tree**".

Diameter of a Tree is longest path of the tree.

Note:

- Radius is not necessarily to Half of its diameter.

Rooted Trees

A tree in which one vertex is distinct from all the others is called the "Rooted Trees".

A Special class of rooted trees called "Rooted Binary Trees".

Note:

- Generally the term tree means a trees without root.
- The root is generally marked differently. We will show the root enclosed in a small Triangle. All rooted trees with 4-vertices are shown in the figure.

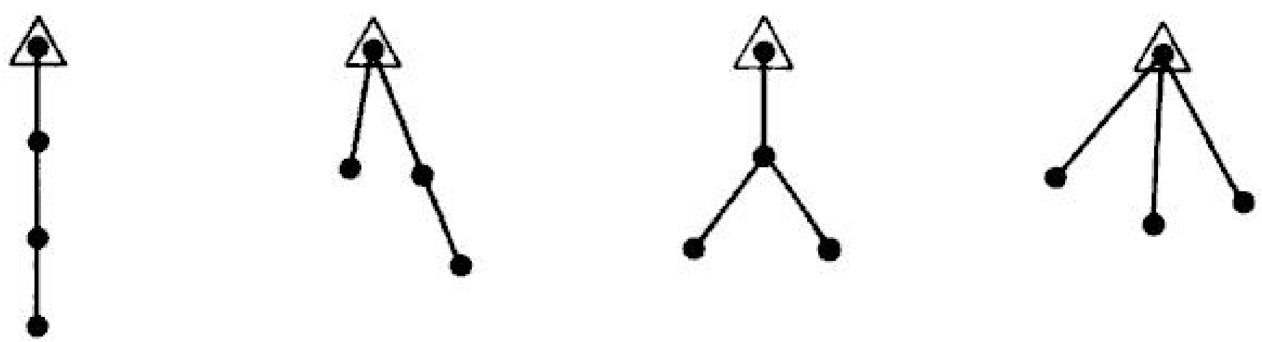


Fig. 3-11 Rooted trees with four vertices.

Defn: (Binary Tree)

A Binary Tree is defined as a tree in which there is exactly one vertex of degree 2 each of the remaining vertices degree 1 or 3.

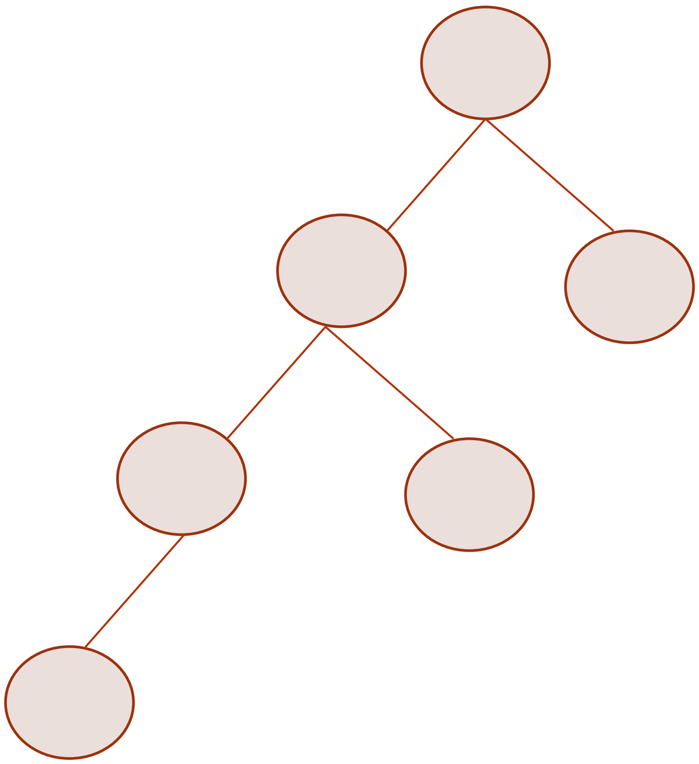
Note:

- Every Binary tree is a rooted tree.

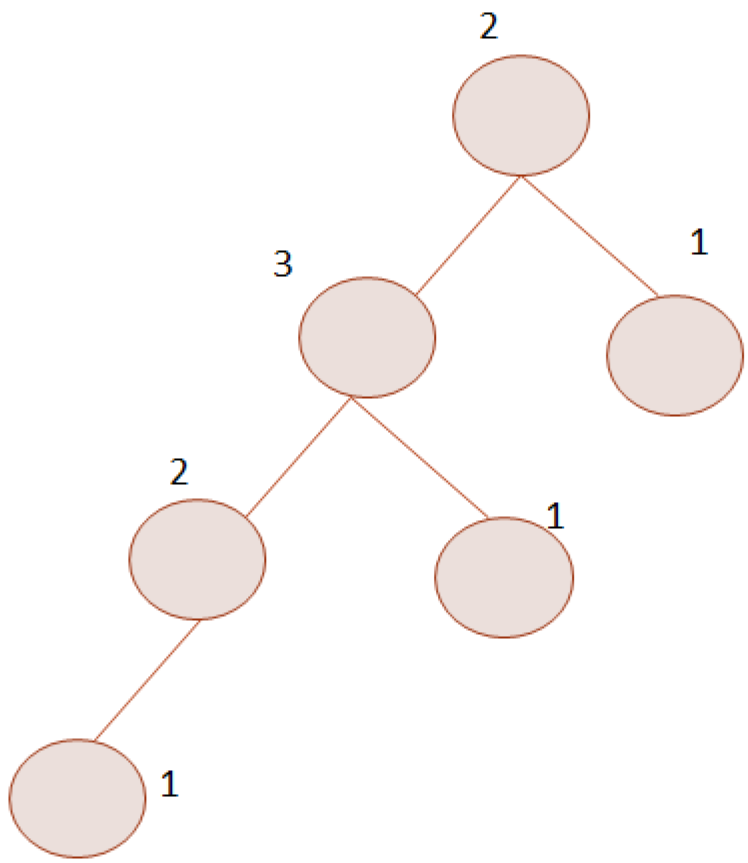
-------- --

Problem: **Draw a Binary tree with 6 - nodes**

Draw a Binary tree with 6 - nodes

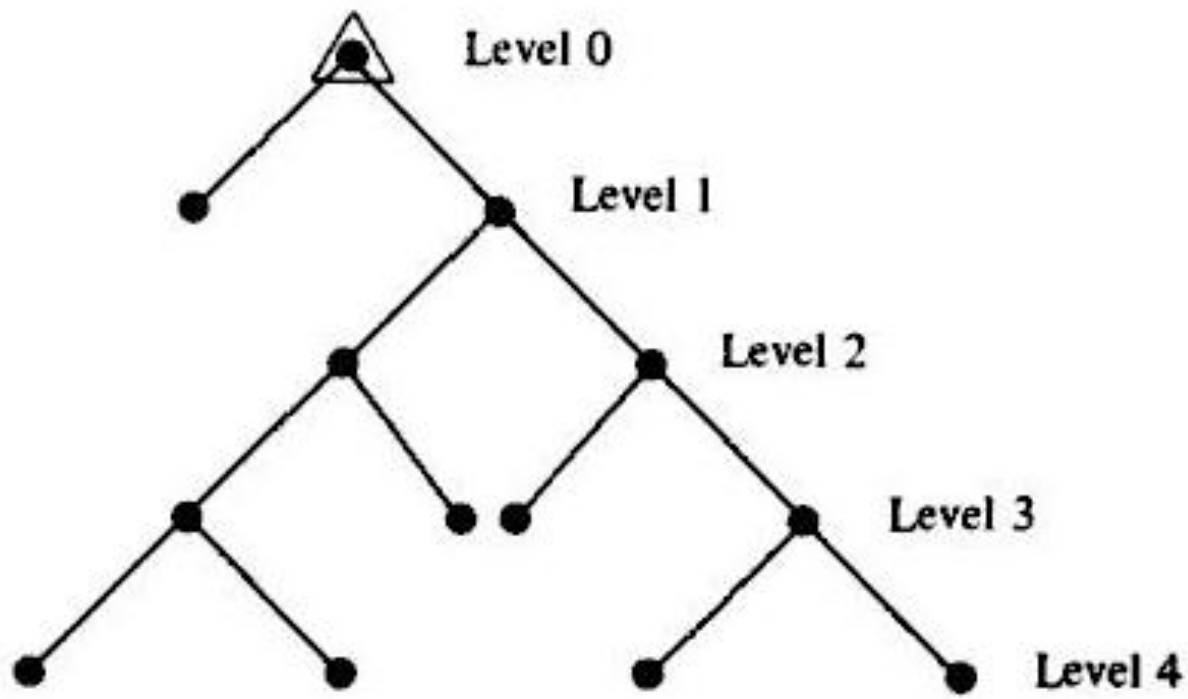


Draw a Binary tree with 6 - nodes



It is not possible to draw a Binary tree with even no. of nodes

DRAW A BINARY TREE WITH 13–NODES(VERTEX)



A 13-vertex, 4-level binary tree.

Properties of a Binary Tree

Property1: (Result)

The number of vertices in a Binary Tree is odd.

Proof:

In a Binary Tree, Root has even degree and Remaining (n-1) edges are odd degree.

WKT, the number of vertices of odd degree is always even.

Therefore (n-1) is even → (n-1) + 1

= n is odd.

Property2: (Result)

The number of pendant vertices in a tree of n-vertices has $\frac{n+1}{2}$

Proof:

Let p be the number of pendent vertices in 'T'.

Let n – be the vertices.

WKT, number of vertex of degree two is 1.

Therefore number of vertices of degree 3 is (n-p-1)

vertices	degree
1	2
p	1
n-p-1	3

WKT, $\sum d(v) = 2e$

$1.2 + P.1 + 3(n-P-1) = 2(n-1).$

$2 + P + 3n - 3P - 3 = 2n - 2$

$3n - 2p - 1 = 2n - 2$

$3n - 2n - 1 + 2 = 2p$

$n+1 = 2P.$

$p = \frac{n+1}{2}$

Result:

The number of internal vertices (non-pendant) is n-p.

Proof:

Here p is $\frac{n+1}{2}$

$$n-p = n - \frac{n+1}{2}$$

$$= \frac{2n-n-1}{2}$$

$$= \frac{n+1}{2}$$

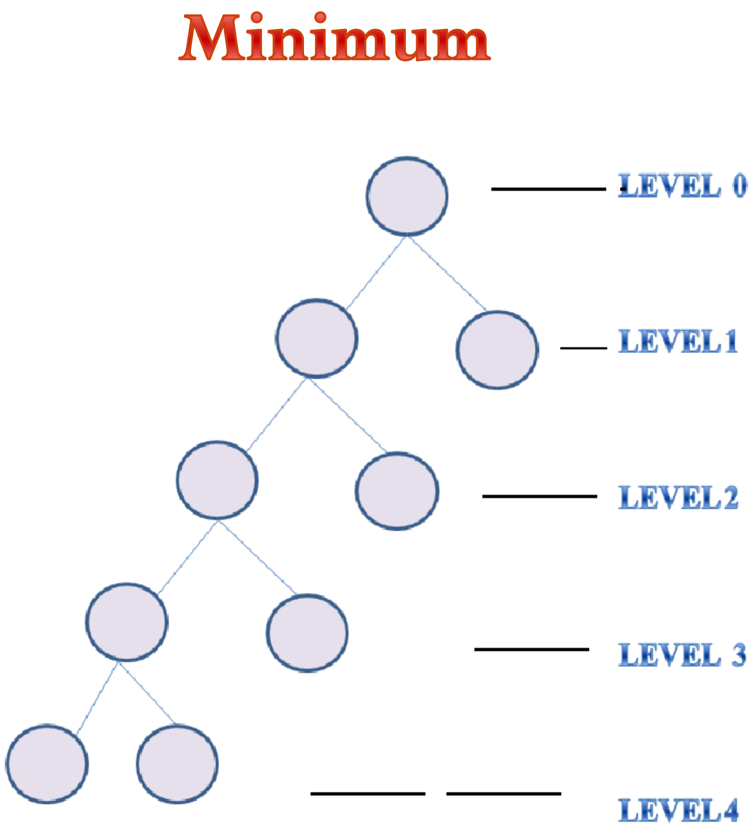
Defn: (LEVEL)

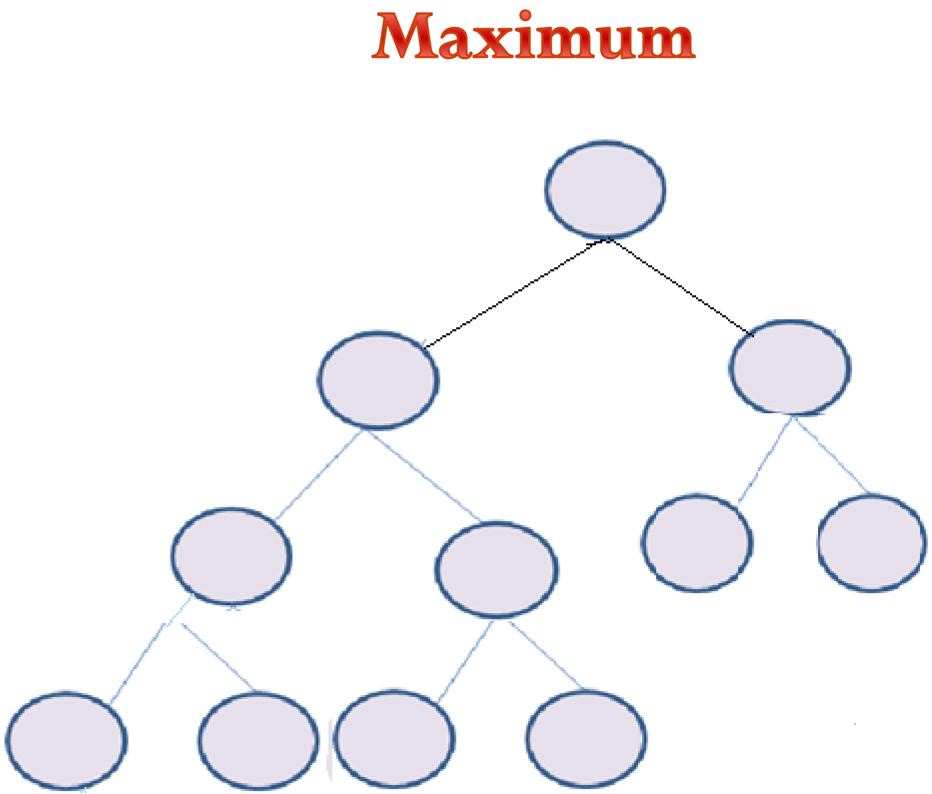
✓ In a binary Tree vertex v_i said to be **at level L_i**. If v_i is at distance of L_i from the root. Thus Root is Level at 0.

NOTE:

- ✓ The Minimum no. of vertices possible in K level binary tree is (2K+1).
- ✓ The Maximum number of vertices possible in

K-Level Binary Tree is $2^0+2^1+2^2+\ldots\ldots+2^k$





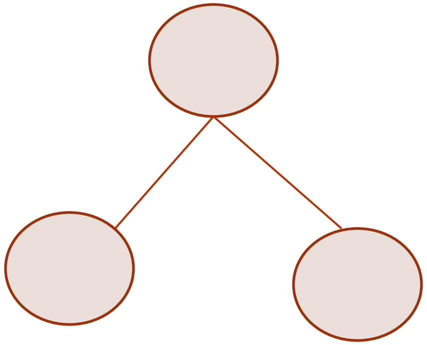
<u>The Height of the Binary tree</u>

The Maximum level of any binary is the "Height of the Binary tree."

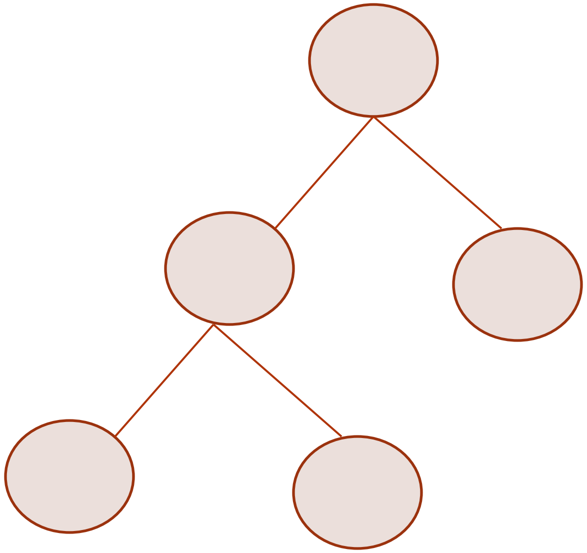
- ✓ The **Maximum** height of the binary tree with **'n' nodes is** $= \frac{n-1}{2}$.
- ✓ The Minimum height of the binary tree with **'n' node** $= \log_2(n+1) - 1$

1. **Find** out the Maximum height of the binary tree with **'n' nodes**.

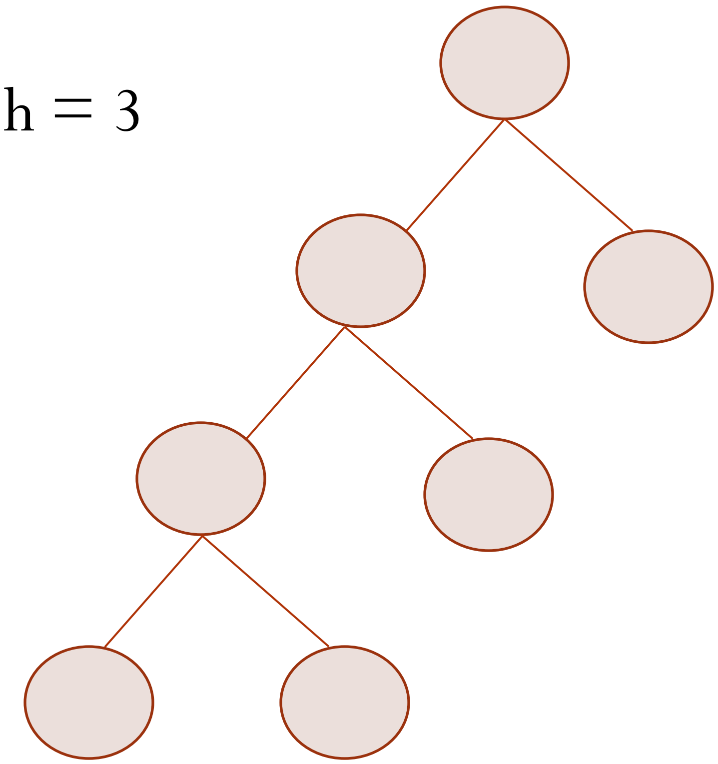
If n = 3 then h = 1



If n = 5 then h = 2

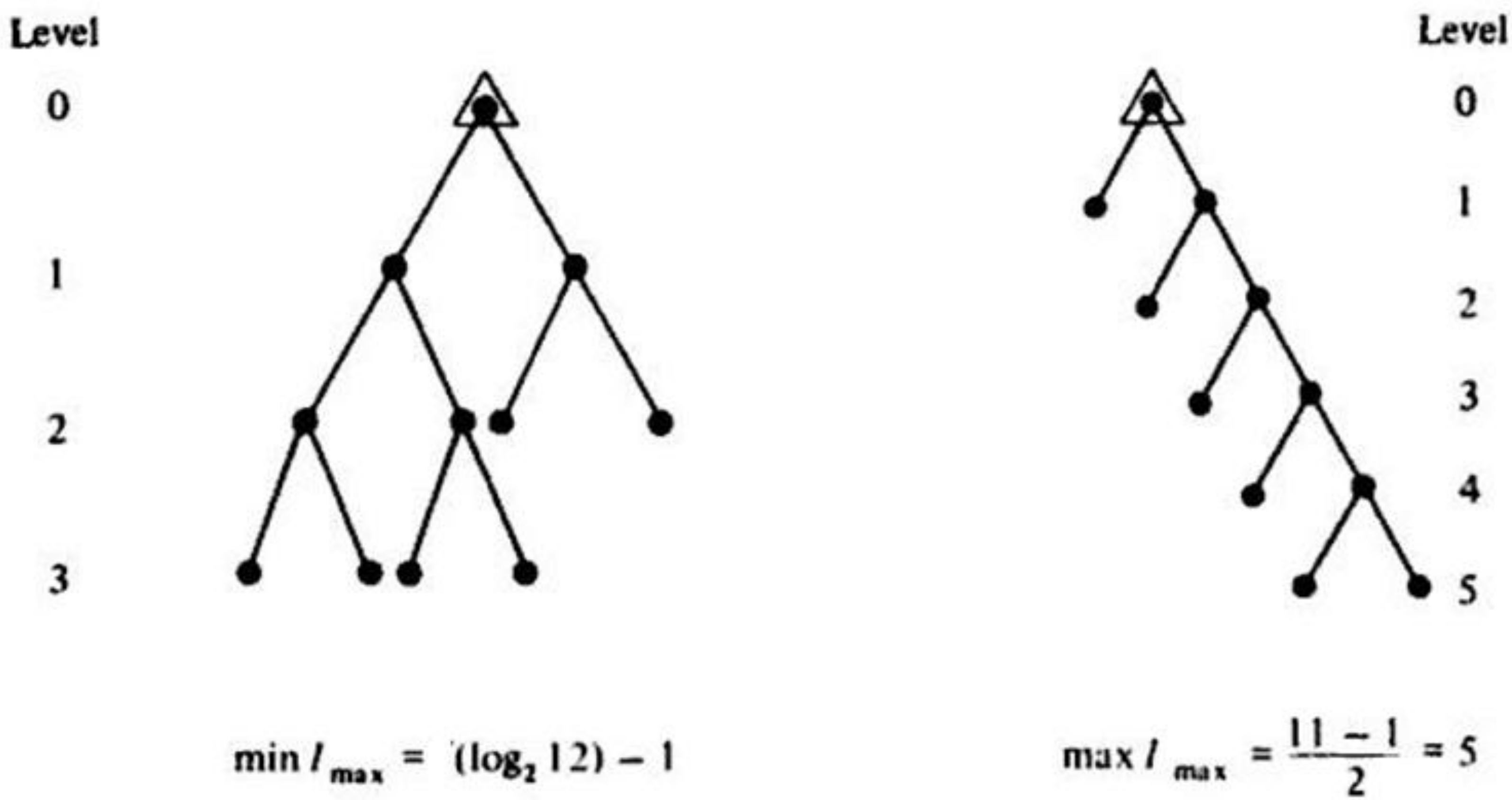


- If n = 7 then h = 3



2. Find out the Minimum and Maximum height of the binary tree with **'11' nodes**.

SOLUTION:



(a) (b)

Weighted path length

- Given w_1, w_2, ...w_n the problem is to construct a binary tree with m-pendent vertex and minimizes $\sum w_i l_i$ (weighted path length).
- where l_i is the level of the pendent vertex v_i **and** the sum is taken over all pendent vertices.

Problem:

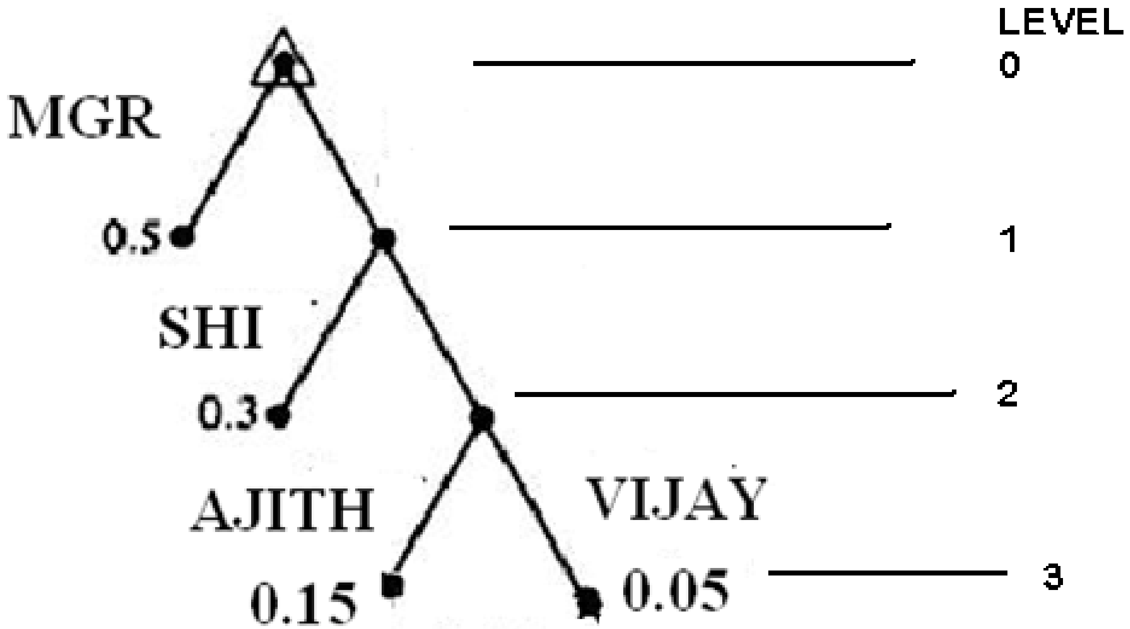
- Let us assume 4- actors namely MGR, SHIVAJI, AJITH, VIJAY.
- The Probabilities of an actor being MGR, SHIVAJI, AJITH, VIJAY is given by 0.5, 0.3, 0.15,0.05
  - If the time taken for each test is same, Identify what sequence of tests will minimize the expected time to identify the actor.

Solution:

Now we construct the binary tree with 4-pendant vertices v_1,v_2,v_3,v_4 and corresponding weights $w_1 = 0.5$, $w_2 = 0.3$, $w_3 = 0.15$, $w_4 = 0.05$ such that $\sum w_i l_i$ minimized

Maximum

fig.1



Minimum

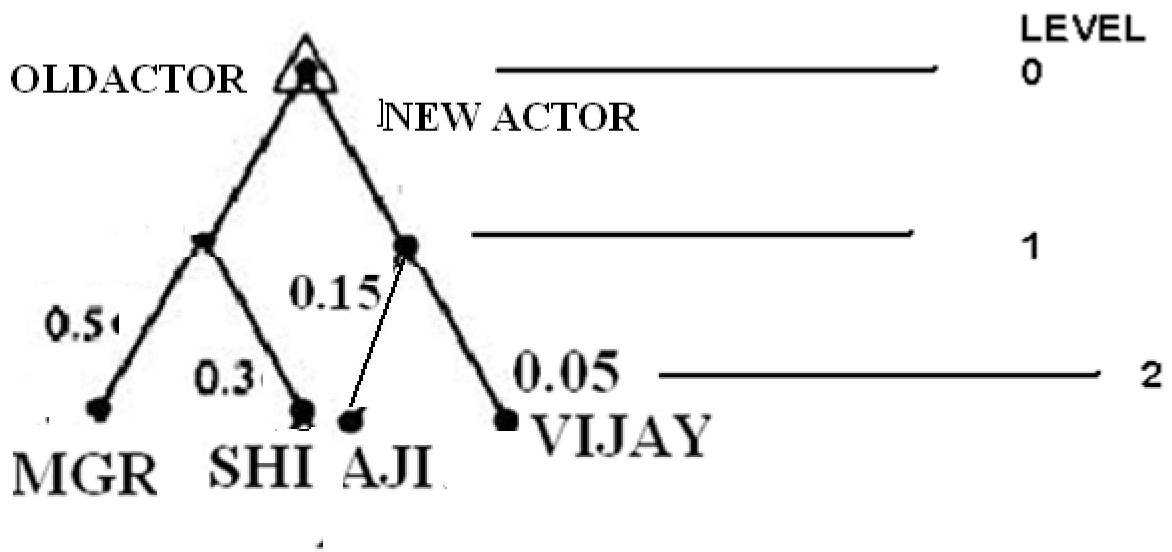


Fig.2

$\sum w_i l_i = 0.5*1+0.3*2 +0.05*3 +0.15*3$

$=1.7.$

$\sum w_i \, l_i = 0.05*2 +0.5*2 \quad +0.15*2 +0.3*2$

$= 2$

The solution is given in the s.t. fig.1, for which the expected time is 1.7t where t-is the time taken for each test.

Contrast this with fig 2nd, for which the expected time is 2t.

- **LABELED GRAPH:**

A graph in which each vertex is assigned a unique name or label

NOTE:

The number of labeled trees with n vertices ($n \geq 2$) is n^{n-2}.

Problem:

Draw and find the different number of Trees that one can construct **with 4-different vertices (labeled vertices).**

Solution:

Problem:

Draw and find the different number of Trees that one can construct **with 4-different vertices (labeled vertices).**

Solution:

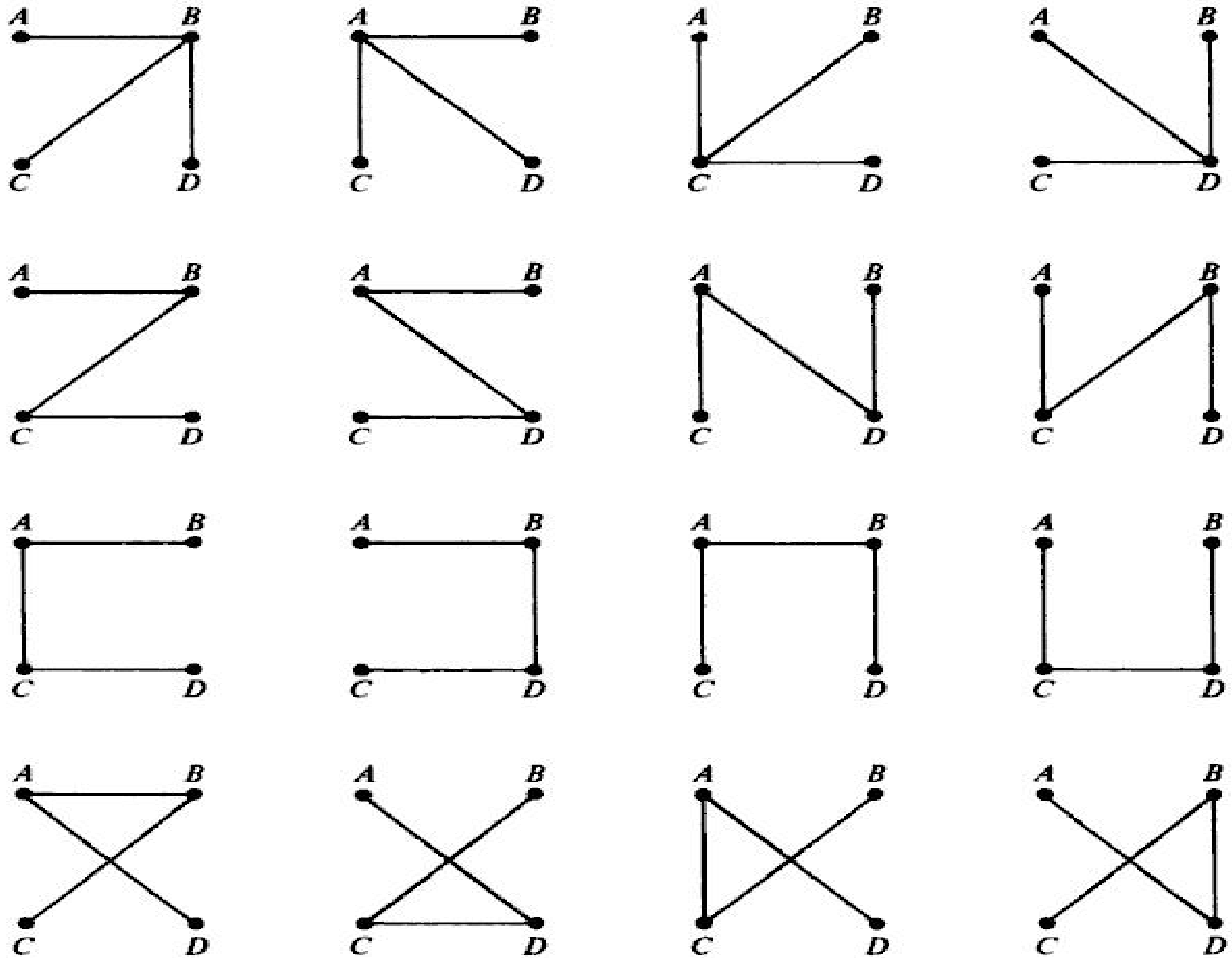


Fig. 3-15 All 16 trees of four labeled vertices.

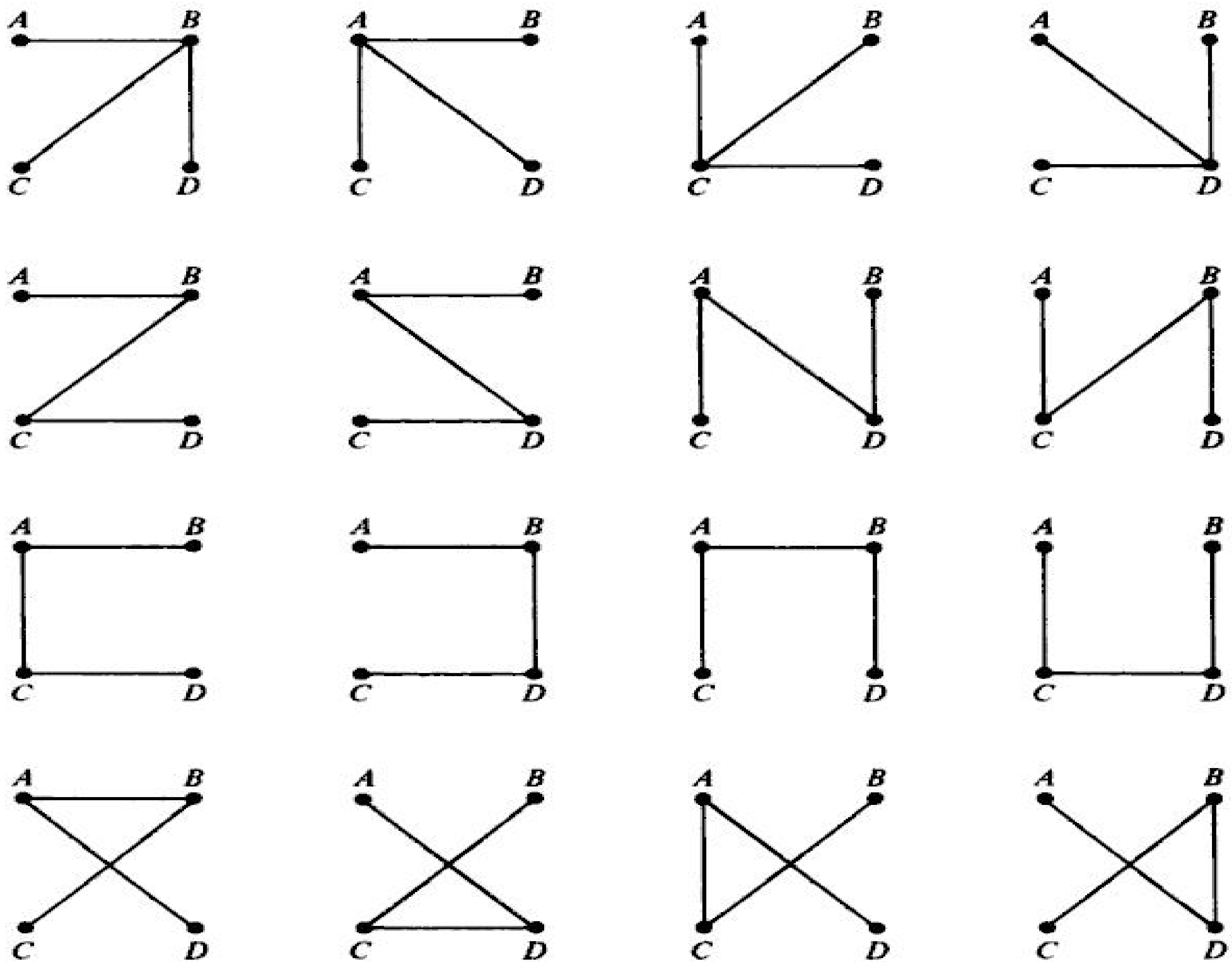
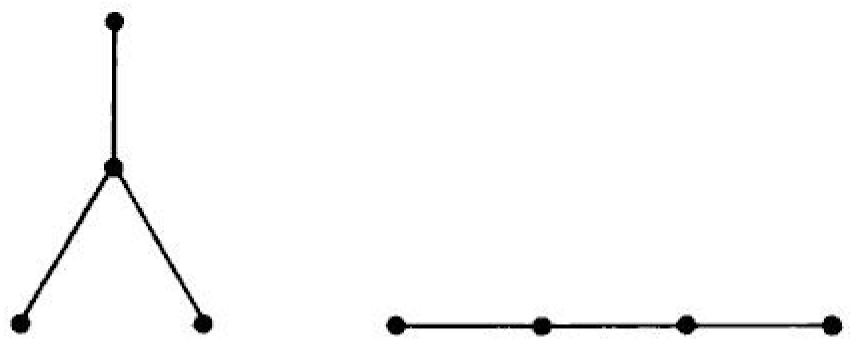


Fig. 3-15 All 16 trees of four labeled vertices.

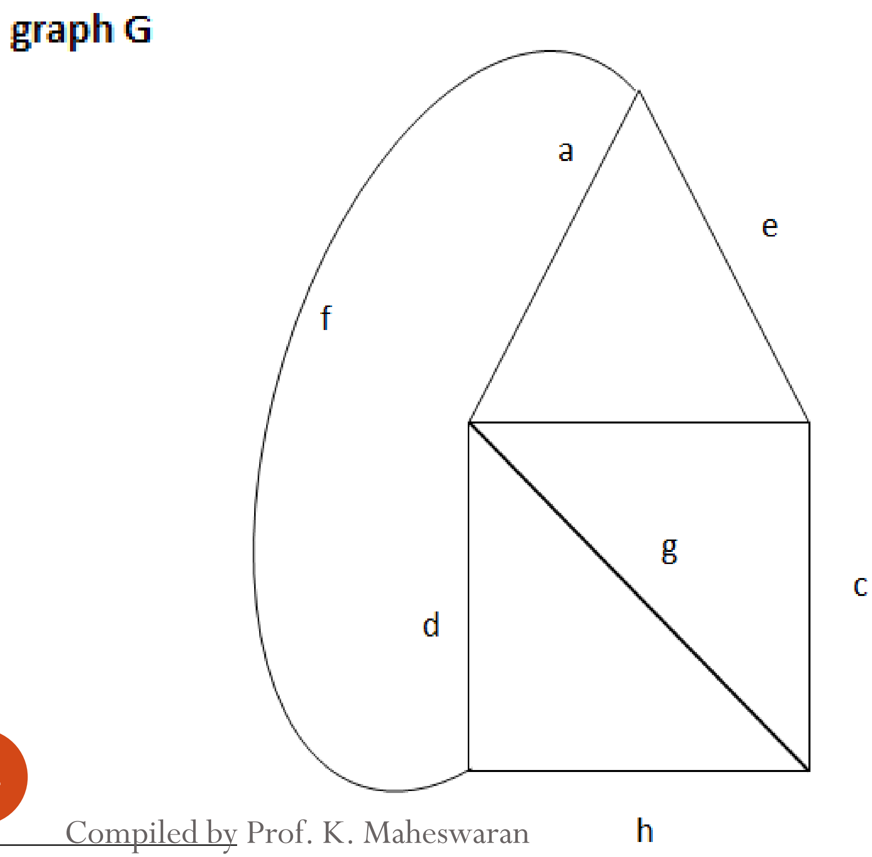
UNLABELED GRAPH

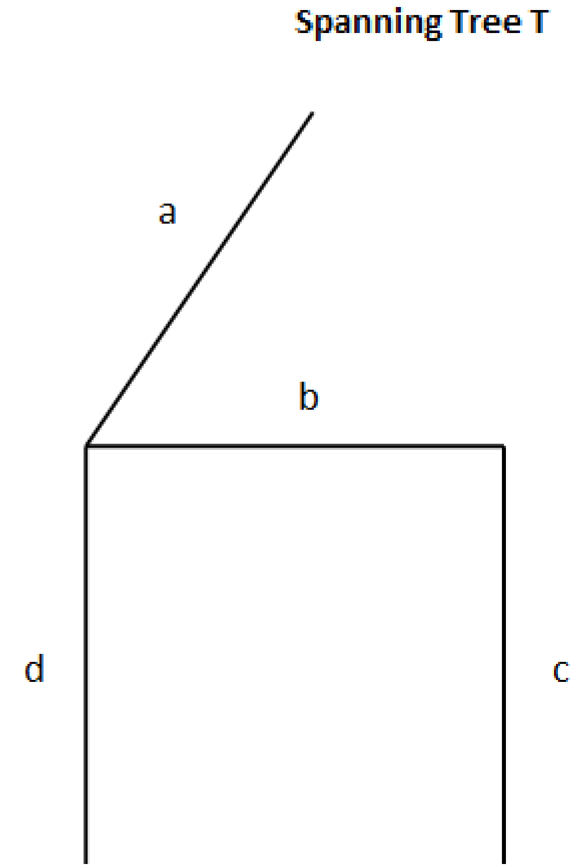


SPANNING TREES

Defn:

A tree T is said to be a spanning tree of a connected graph G if (i) T is a sub graph of G and
(ii) contains all vertices of G.





Theorem:

Every connected graph has atleast one spanning tree.

Proof:

If G has no circuit, it is its own spanning tree.

If G has a circuit, delete an edge from the circuit. This will still leave the graph connected. If there are more circuits, repeat the operation till an edge from the last circuit is deleted- leaving a connected, circuit-free graph that contains all the vertices of G.

Definition:

Branch

An edge in a spanning tree T is called Branch of T

Chord

An edge in G which is not in a spanning tree T is called **Chord** of T

Chord set (Co tree)

If T is a spanning Tree, then complement of T is T'. T' is called **chord is set or co-tree.**

Note:

In a connected graph of n vertices and e edges, w.r.t any of its spanning tree.

No of branches = n-1

No of chords = e-(n-1) = e-n+1.

(i.e) To eliminate all circuits,

the number of edges to be deleted is = e-n+1.

Definitions:

Rank(r):

Let n be the no of vertices , e be the no of edges k be the no of components then

the rank of G $r = n-1$ and

the null links of connected G $\mu = e-n+1$.

(cyclomatic, Betli no.)

Therefore rank of G = no of branches in any spanning tree of G.

Nullity of G = no of chords in G.

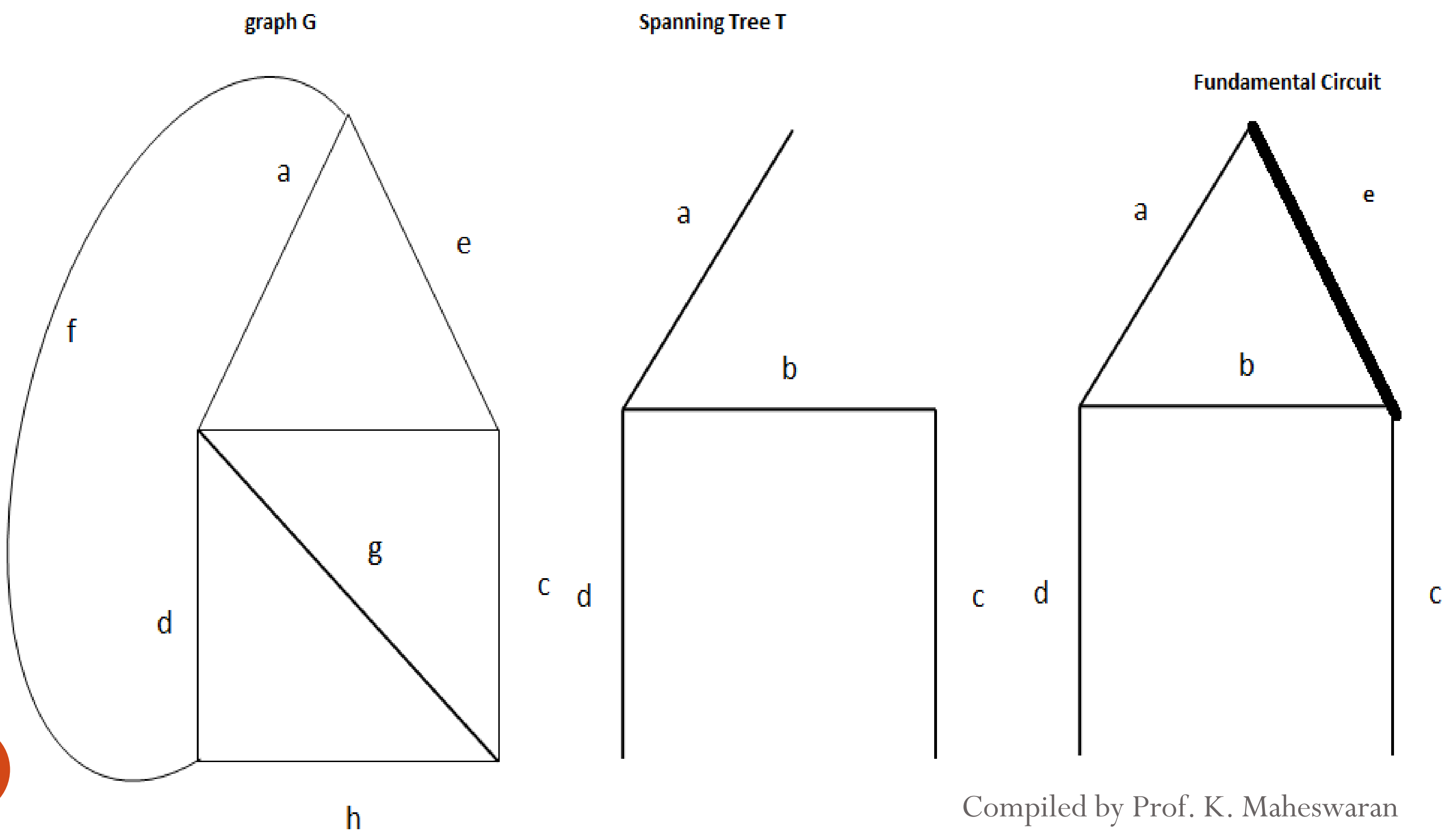
Rank + Nullity = no of edges in G.

Fundamental Circuits

Definition:

A circuit formed by adding a chord to a spanning tree, is called a fundamental circuit.

A circuit is a fundamental circuit only w.r.t a given spanning tree.



Compiled by Prof. K. Maheswaran

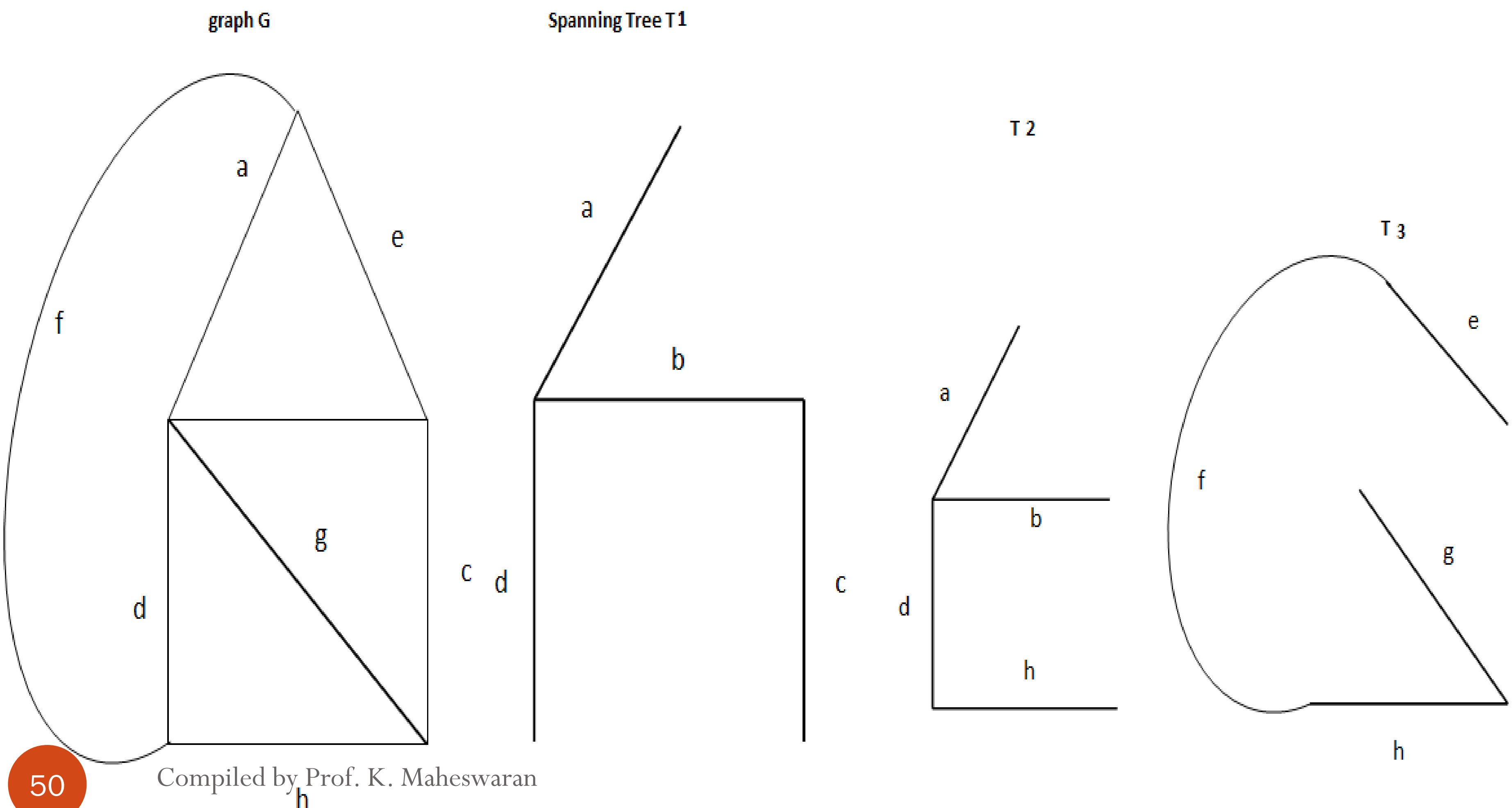
Finding all spanning trees of a graph:

Definition: (Cyclic Interchange or Elementary tree transformation.)

The generation of one spanning trees from another, through addition of a chord and deletion of an appropriate branch is called cyclic interchange.

How do get all spanning trees of a graph?

Start with a given spanning tree, and by applying cyclic interchange we will get new spanning tree.



Definition:

The **distance between two spanning trees** T_i & T_j of a graph G is defined as the number of edges of G present in one tree but not in the other. This is written as $d(T_i, T_j)$ (i.e) $d(T_2, T_3) = 3$.

$$d(T_i, T_j) = \frac{1}{2} N (T_i \oplus T_j)$$

where N(g) means Number of edges in g.

Central Tree:

For a spanning tree To of a graph G, let max $d(T_o,T_i)$ denote the maximal distance between To and other spanning tree of G. Then To is called a central tree of G if max. $d(T_o,T_i) \leq \max d(T,T_j)$ T of G.

Spanning a trees in a weighted Graph:

Definition: (1)

Weight of the spanning tree is defined as the sum of the weight of all the branches in T.

Definition: (2)

Shortest spanning tree or minimal spanning tree.

A spanning tree with smallest weight in a weighted Graph is called a shortest spanning tree.

Theorem:

A spanning tree T (of a given weighted connected graph G) is a shortest spanning tree of G iff there exists no other spanning tree of G at a distance of one from T whose weight is smaller than that of T.

Proof:

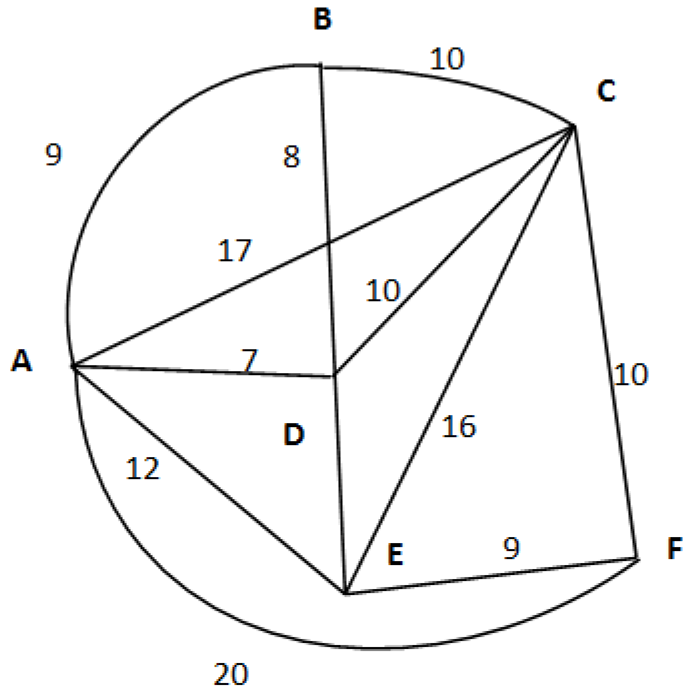
A spanning tree T is shortest of G

(obvious) There exists no other spanning tree of G at a distance of one from T, whose wt is smaller than that of T.

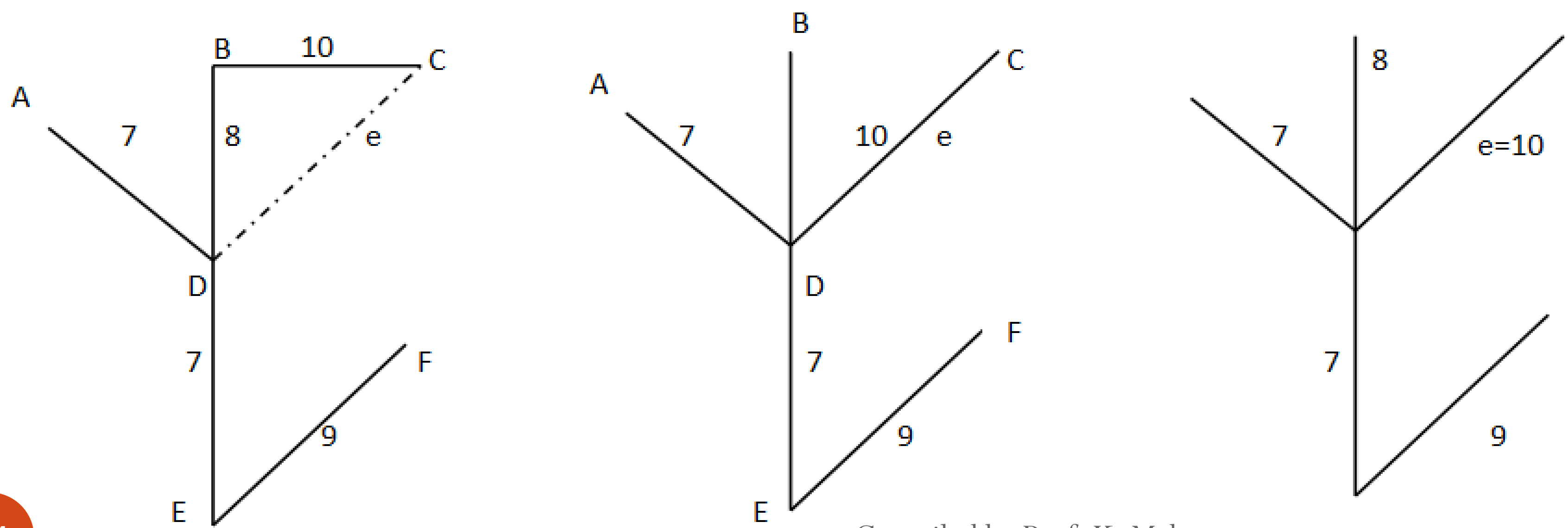
bConverse:

Let T_1 be a spanning tree in G satisfying that there is a no spanning tree at a distance of one from T_1 which is shorter than T_1. The proof will be completed by showing that if T_2 is a shortest spanning tree (different from T_1) in G, the weight of T_1 will also be equal that of T_2.

Let T_2 be a shortest spanning tree in G clearly T_2 must also [sa]tisfy the hypothesis of the theorem.



Consider an edge e in T_2 which is not in T_1 adding e to $T1$ forms a fundamental circuit which branches in T_1.



MATRIX REPRESENTATION OF A GRAPH

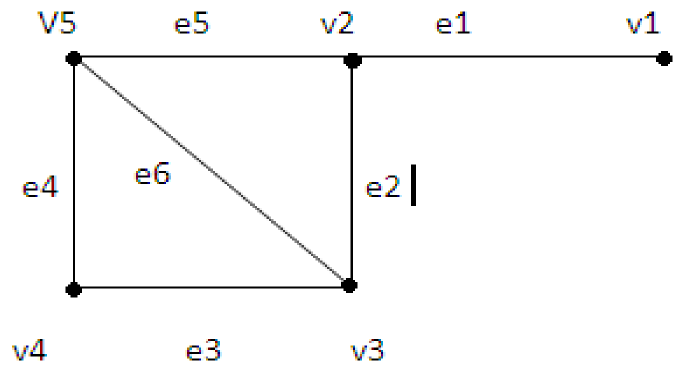
INCIDENCE MATRIX

Let G be a graph with n – vertices, e – edges and no self loops, then the incidence matrix (m X n) , whose element is given by

$$\mathbf{A(G)= [a_{ij}]}$$

where $\mathbf{a_{ij}} = \begin{cases} \mathbf{1} & \text{if } V_i \text{ is incident with } e_j \\ \mathbf{0} & \text{otherwise} \end{cases}$

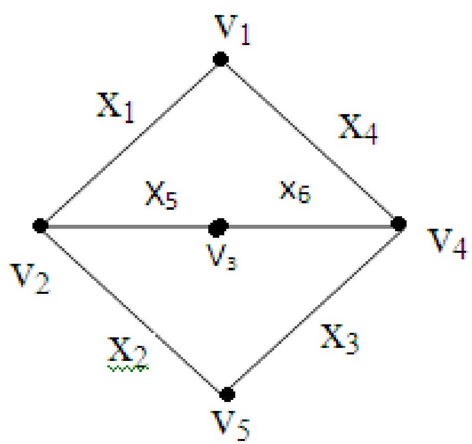
Find the Incidence Matrix for the following graph 'G'



A(G) =

	e1	e2	e3	e4	e5	e6
v1	1	0	0	0	0	0
v2	1	1	0	0	1	0
v3	0	1	1	0	0	1
v4	0	0	1	1	0	0
v5	0	0	0	1	1	1

Find the Incidence Matrix for the following graph 'G'



	x1	x2	x3	x4	x5	x6
v1	1	0	0	1	0	0
v2	1	1	0	0	1	0
v3	0	0	0	0	1	1
v4	0	0	1	1	0	1
v5	0	1	1	0	0	0

Find the Incidence Matrix for the following graph 'G'

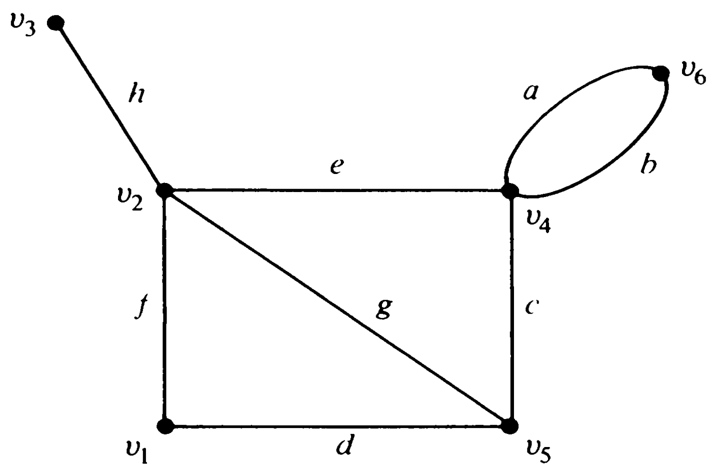


fig 7.1

A(G) =

	a	b	c	d	e	f	g	h
v_1	0	0	0	1	0	1	0	0
v_2	0	0	0	0	1	1	1	1
v_3	0	0	0	0	0	0	0	1
v_4	1	1	1	0	1	0	0	0
v_5	0	0	1	1	0	0	1	0
v_6	1	1	0	0	0	0	0	0

Properties(observations) of Incidence Matrix

1. Since every edge is incident on exactly two vertices, each column of A has exactly two 1's.
2. The number of 1's in each row equals the degree of the corresponding vertex.
3. A row with all 0's, therefore, represents an isolated vertex.
4. Parallel edges in a graph produce identical columns in its incidence matrix, for example, columns 1 and 2 in Fig. 7-1.
5. If a graph G is disconnected and consists of two components g_1 and g_2, the incidence matrix $A(G)$ of graph G can be written in a block-diagonal form as

$$A(G) = \left[\begin{array}{c|c} A(g_1) & 0 \\ \hline 0 & A(g_2) \end{array}\right], \qquad (7\text{-}1)$$

   where $A(g_1)$ and $A(g_2)$ are the incidence matrices of components g_1 and g_2. This observation results from the fact that no edge in g_1 is incident on vertices of g_2, and vice versa. Obviously, this remark is also true for a disconnected graph with any number of components.
6. Permutation of any two rows or columns in an incidence matrix simply corresponds to relabeling the vertices and edges of the same graph.

ADJACENCY MATRIX

A Graph 'G' with n – vertices and no parallel edges then the Adjacency matrix (n X n) is a Symmetric Binary Matrix, whose element is given by,

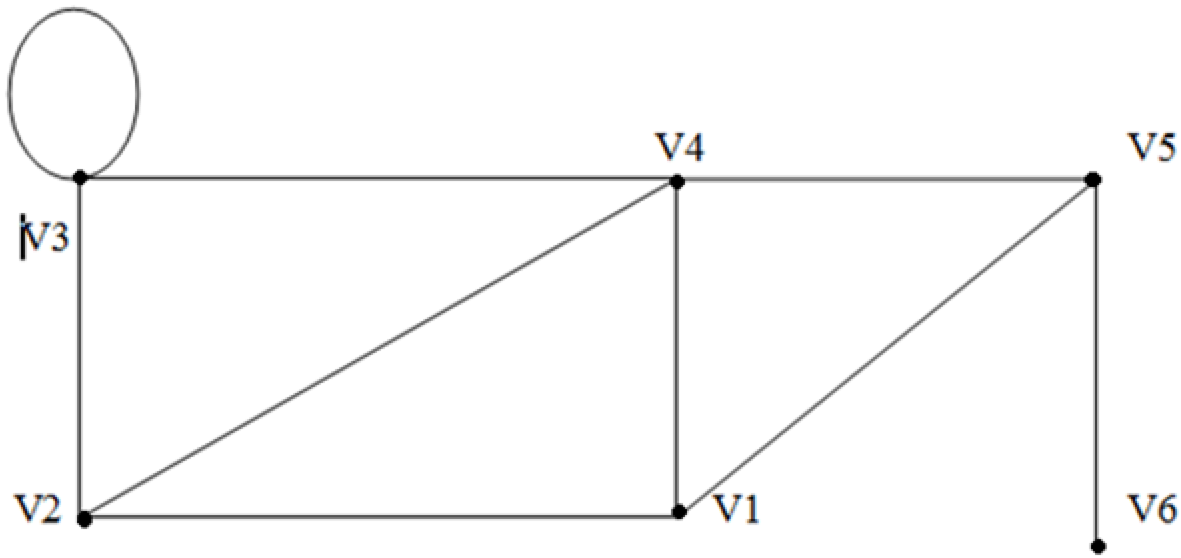
$$\mathbf{X(G) = [x_{ij}]}$$

Defined over the ring of integer such that

$\mathbf{x_{ij}}$ $\mathbf{= 1}$ if there is an edge between i^{th} and j^{th} vertices

$\mathbf{= 0}$ otherwise

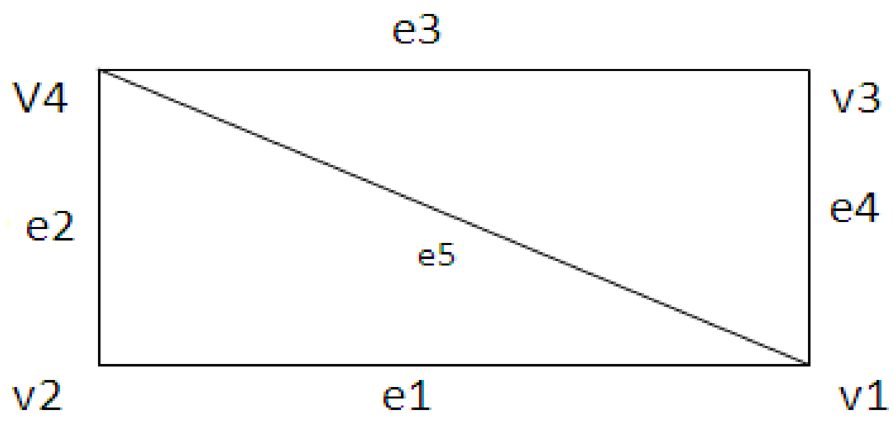
EX: Find the adjacency Matrix of following Graph 'G'.



X (G)=

	v1	v2	v3	v4	v5	v6
v1	0	1	0	1	1	0
v2	1	0	1	1	0	0
v3	0	1	1	1	0	0
v4	1	1	1	0	1	0
v5	1	0	0	1	0	1
v6	0	0	0	0	1	0

Find the adjacency Matrix of following Graph 'G'.



adjacency Matrix

	V1	V2	V3	V4
V1	0	1	1	1
V2	1	0	0	1
V3	1	0	0	1
V4	1	1	1	0

Properties(observations) of Adjacency Matrix

1. The entries along the principal diagonal of X are all 0's if and only if the graph has no self-loops. A self-loop at the ith vertex corresponds to $x_{ii} = 1$.
2. The definition of adjacency matrix makes no provision for parallel edges. This is why the adjacency matrix X was defined for graphs without parallel edges.†

3. If the graph has no self-loops (and no parallel edges, of course), the degree of a vertex equals the number of 1's in the corresponding row or column of X.

4. Permutations of rows and of the corresponding columns imply reordering the vertices. It must be noted, however, that the rows and columns must be arranged in the same order. Thus, if two rows are interchanged in X, the corresponding columns must also be interchanged. Hence two graphs G_1 and G_2 with no parallel edges are isomorphic if and only if their adjacency matrices $X(G_1)$ and $X(G_2)$ are related:

$$X(G_2) = R^{-1} \cdot X(G_1) \cdot R,$$

where R is a permutation matrix.

5. A graph G is disconnected and is in two components g_1 and g_2 if and only if its adjacency matrix $X(G)$ can be partitioned as

$$X(G) = \left[\begin{array}{c|c} X(g_1) & 0 \\ \hline 0 & X(g_2) \end{array}\right],$$

where $X(g_1)$ is the adjacency matrix of the component g_1 and $X(g_2)$ is that of the component g_2.

This partitioning clearly implies that there exists no edge joining any vertex in subgraph g_1 to any vertex in subgraph g_2.

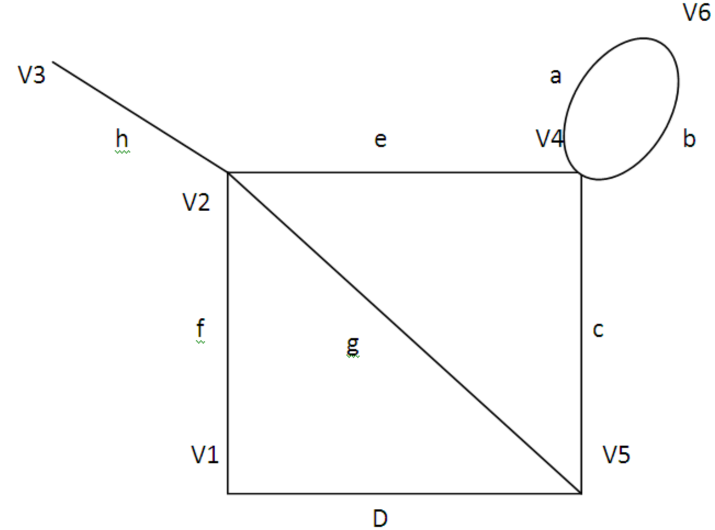
6. Given any square, symmetric, binary matrix Q of order n, one can always construct a graph G of n vertices (and no parallel edges) such that Q is the adjacency matrix of G.

CIRCUIT MATRIX:

Let the number of different circuits in a graph G be q and the number of edges be ‘e’. Then a circuit matrix (C(G)) is given by **B (G) = [b_{ij}]** of a graph is a q x e matrix defined as follows

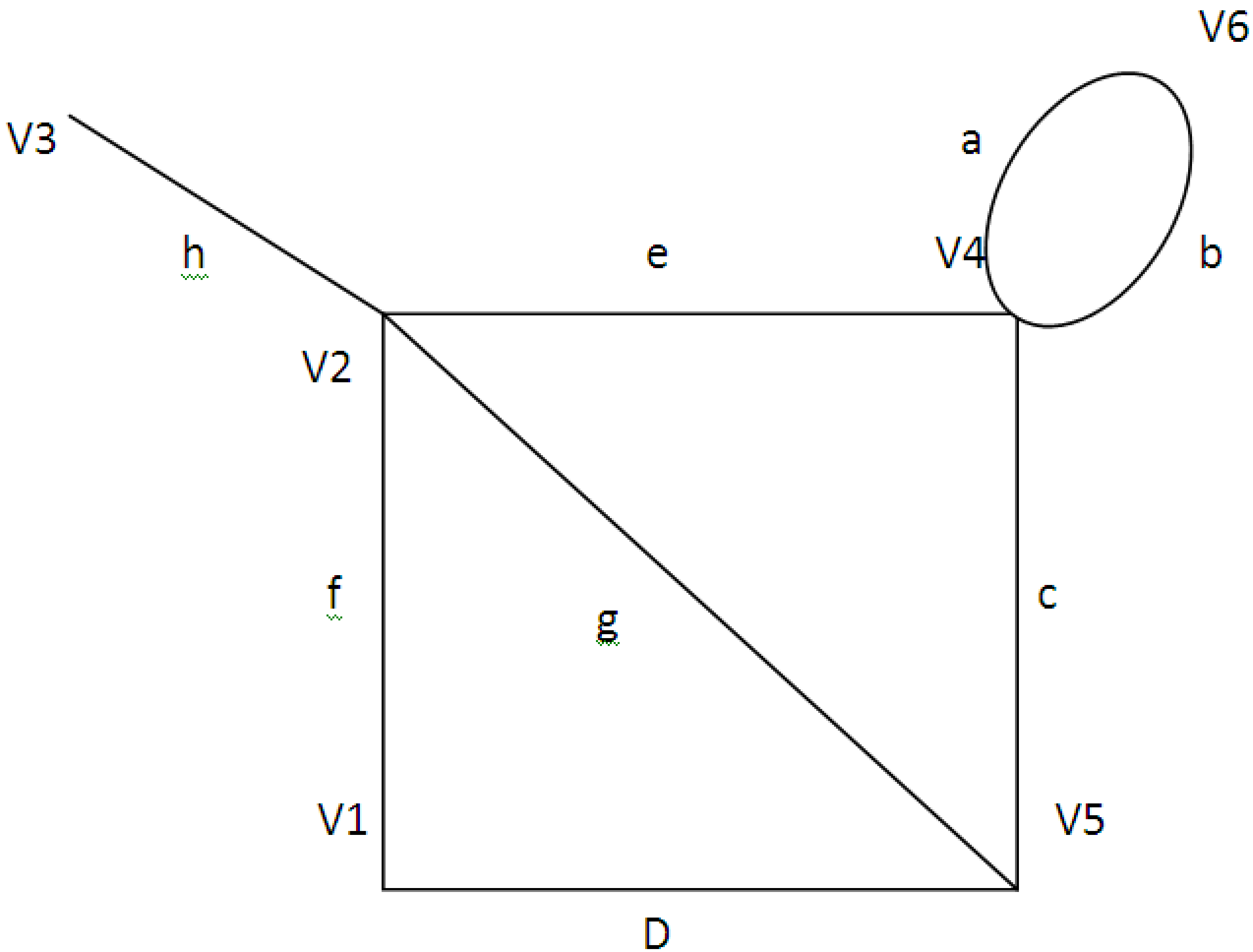
b_{ij} = 1 if i^{th} circuit includes j^{th} edge

EX:



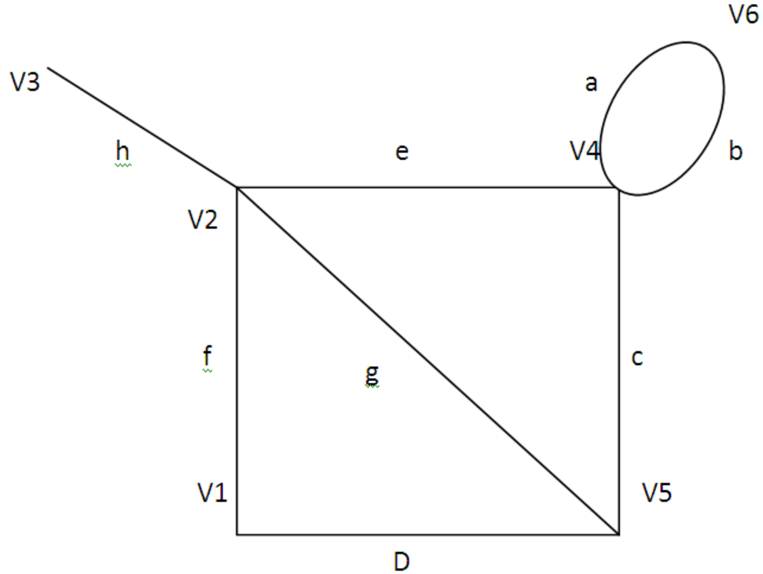
C1={ a,b} , C2={c,e,g}, C3={d,f,g}, C4={c,d,f,e}

EX:



C1={ a,b} ,   C2={c,e,g},   C3={d,f,g},   C4={c,d,f,e}

EX:



C1={ a,b} , C2={c,e,g}, C3={d,f,g}, C4={c,d,f,e}

B (G)=

	a	b	c	d	e	f	g	h
C1	1	1	0	0	0	0	0	0
C2	0	0	1	0	1	0	1	0
C3	0	0	0	1	0	1	1	0
C4	0	0	1	1	1	1	0	0

FUNDAMENTAL CIRCUIT(FC):

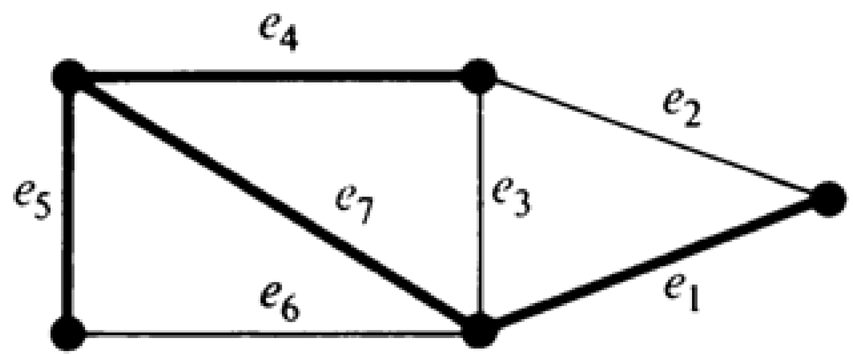
The total number of fundamental circuit is e-n+1. A sub-matrix in which all rows corresponds to an set of FC is called fundamental circuit matrix ($\mathbf{C}_f$).

Arrange the columns in $\mathbf{C}_f$ such that all the e-n+1 chords corresponds to the first e-n+1 columns.

A matrix $\mathbf{C}_f$ is arranged as,

$$\mathbf{C}_f = [\ I\mu \ |\ Bt\ ]$$

Where Iμ is an identity matrix of order μ=e-n+1 and Bt is the remaining by (n-1) sub matrix corresponds to the branches of the spanning tree

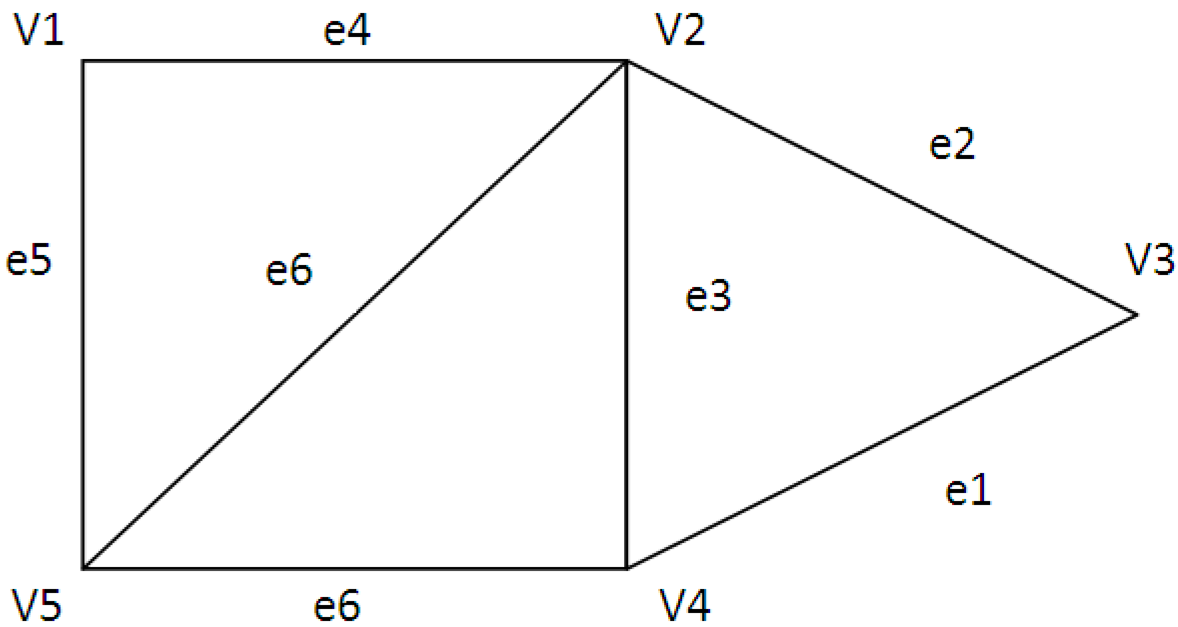


(a)

$$
\begin{array}{ccc|cccc}
e_2 & e_3 & e_6 & e_1 & e_4 & e_5 & e_7 \\
\end{array}
$$

$$
\left[\begin{array}{ccc|cccc}
1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 1
\end{array}\right]
$$

EX:



C_f =

e2	e3	e4	e1	e4	e5	e7
1	0	0	1	1	0	1
0	1	0	0	1	0	1
0	0	1	0	0	1	1