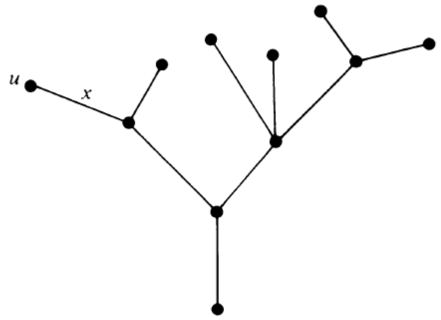
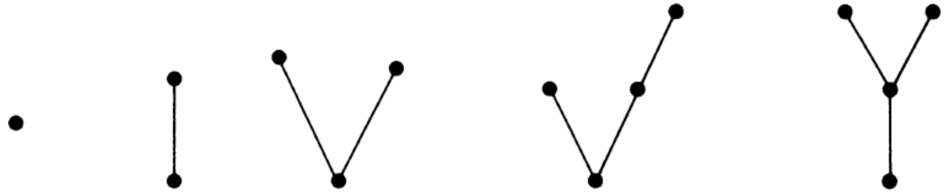
Tree and Fundamental Circuits

- Tree is a connected simple graph without any circuits.
- As a graph must have at least one vertex, and therefore so must a tree. A tree without any vertices is called *null tree*.
- Similarly, considering only finite graphs, our trees are also finite.
- a tree has to be a simple graph, that is, having neither a self-loop nor parallel edges.



Tree



Tree with no of vertices=1,2,3,4

Examples

→ a b c d e

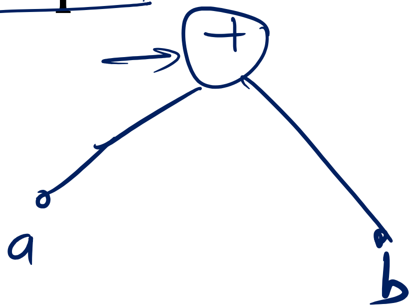
→ 35 10 5 15 35

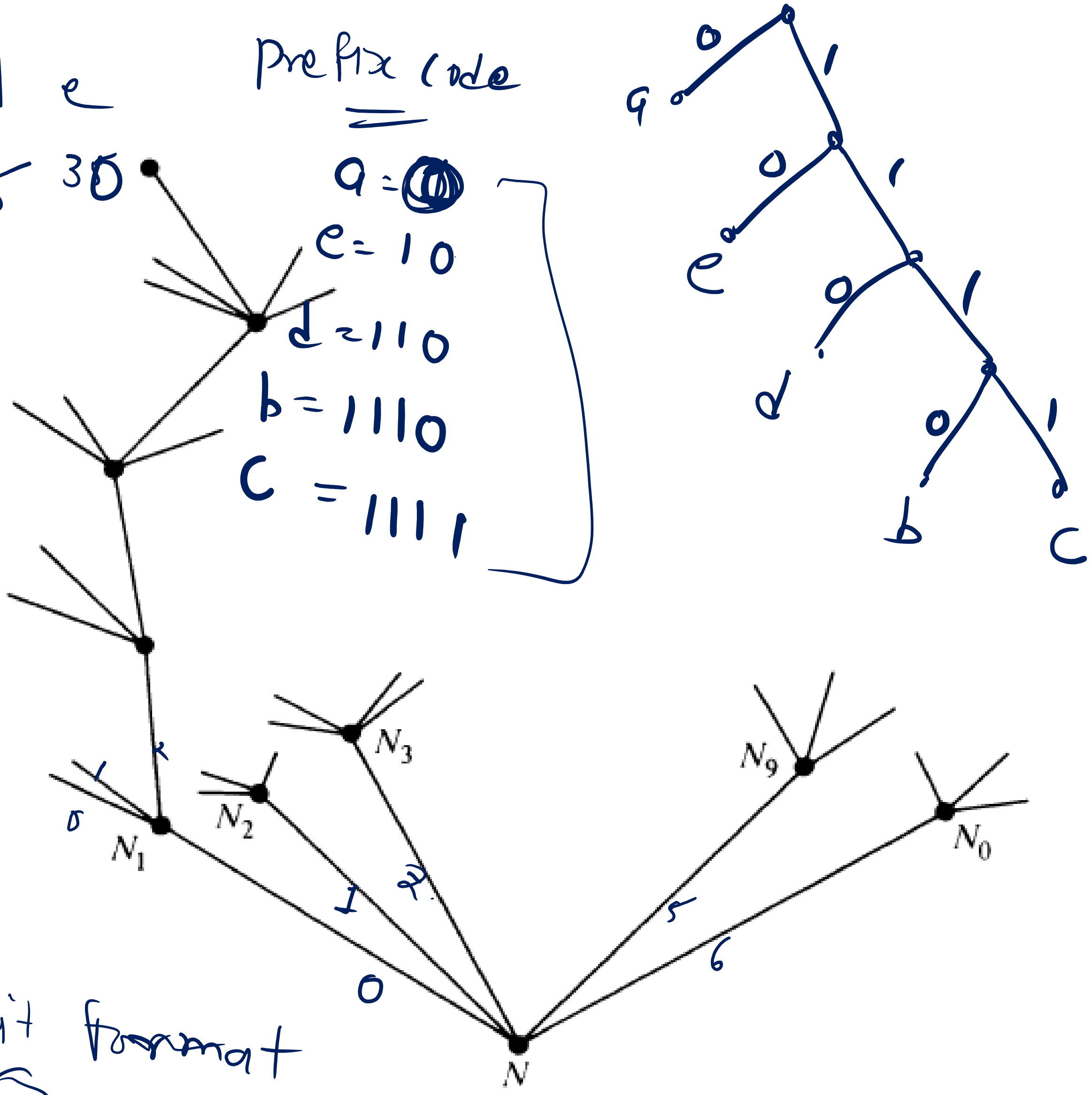
prefix code

a = 0
e = 10
d = 110
b = 1110
c = 1111

- Family Tree
- A river with its tributaries and subtributaries
- The sorting of mails according to zip code
- Huffman code →
- Arithmetic expressions

→ a + b





6 digit format

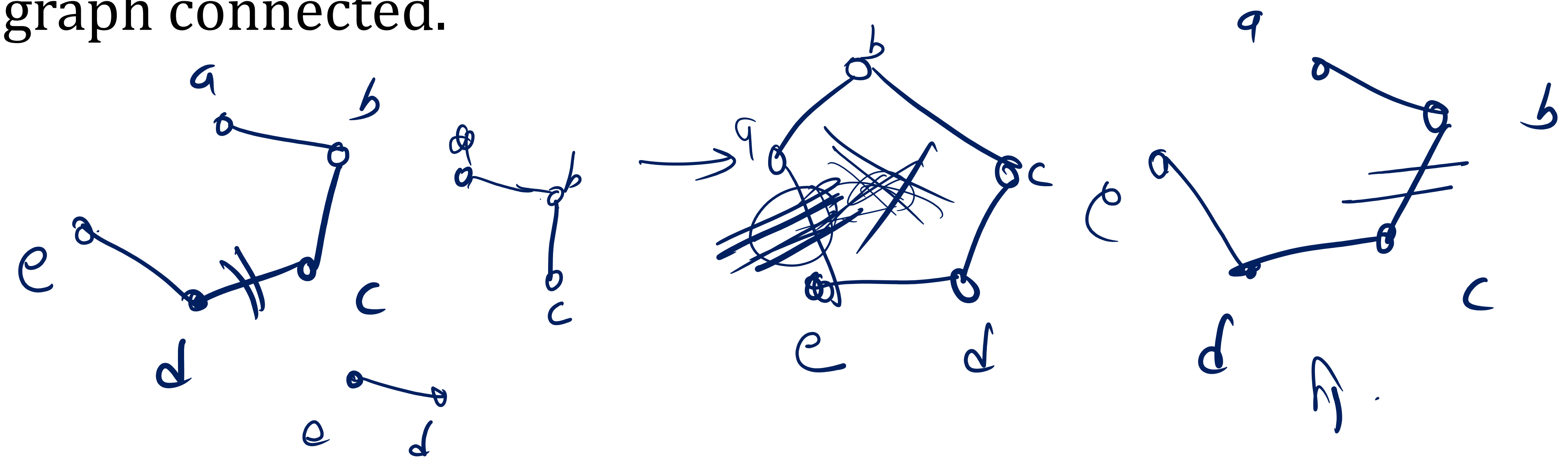
10 10 10 10 10 10

10^6

Properties of trees

- **Theorem 3-1:** There is one and only one path between every pair of vertices in a tree, T. (Prove with contradiction)
- **Theorem 3-2:** If in a graph G there is one and only one path between every pair of vertices, G is a tree.(Prove with direct proof)
- **Theorem 3-3:** A tree with n vertices has n-1 edges. (Mathematical Induction)
- **Theorem 3-4:** Any connected graph with n vertices and n-1 edges is a tree.
- **Theorem 3-5:** A graph is a tree if and only if it is minimally connected.
- **Theorem 3-6:** A graph G with n vertices, n-1 edges, and no circuits is connected. (Prove with contradiction)

- A connected graph is said to be minimally connected if removal of any one edge from it disconnects the graph.
- A minimally connected graph cannot have a circuit; otherwise, we could remove one of the edges in the circuit and still leave the graph connected.



A graph G with n vertices is called a tree if

1. G is connected and is circuitless, or
2. G is connected and has n-1 edges, or
3. G is circuitless and has n-1 edges, or
4. There is exactly one path between every pair of vertices in G, or
5. G is minimally connected graph.

If a graph G has circuits, it is not minimally connected.

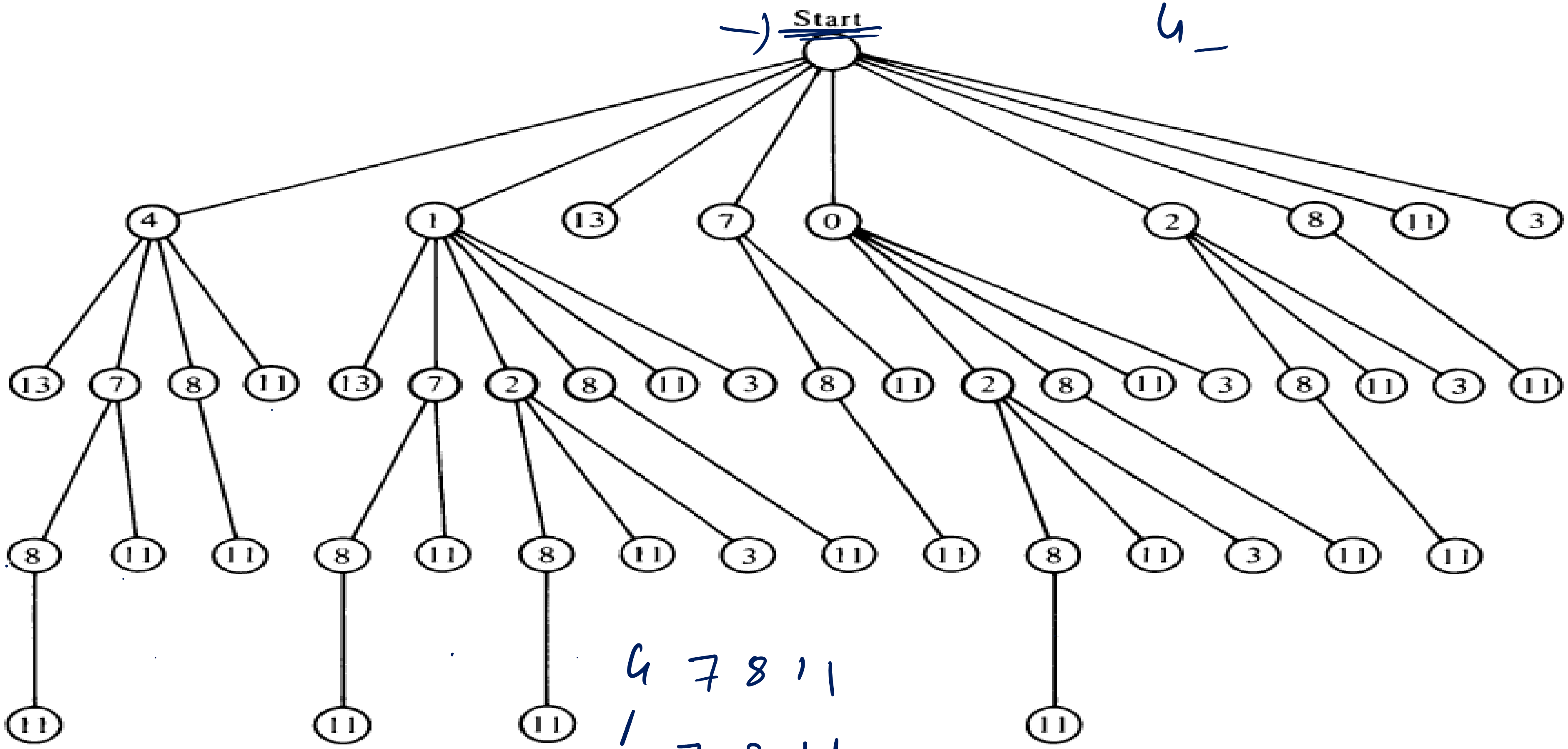
True / false

Pendant vertices in a Tree

- Pendant vertex is a vertex of degree 1.
- A tree of n vertices have n-1 edges, and 2(n-1) degrees which is to be divided among n vertices.
- Since no vertex can be of zero degree, at least two vertices of degree one in a tree, which means $n \geq 2$.
- **Theorem 3-7:** In any tree (with two or more vertices), there are at least two pendant vertices.

Application: Monotonically increasing sequences in 4,1,13,7,0,2,8,11,3.

4



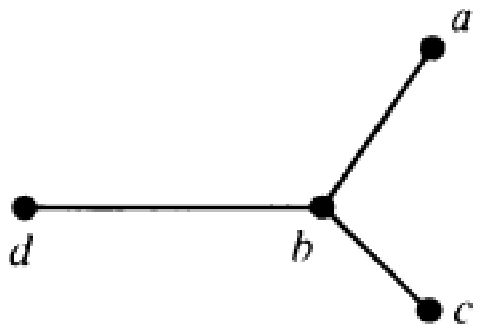
4 7 8 11

1 7 8 11

1 2 8 11, 0 2 8 11

Distance in a Tree

- In a connected graph G, the distance $d(v_i,v_j)$ between two of its vertices v_i and v_j is the length of the shortest path (number of edges in the shortest path) between them.
- In a tree, since there is exactly one path between any two vertices, the determination of distance is much easier.



d(a,b) = 1, d(a,c) = 2, d(c,b) = 1

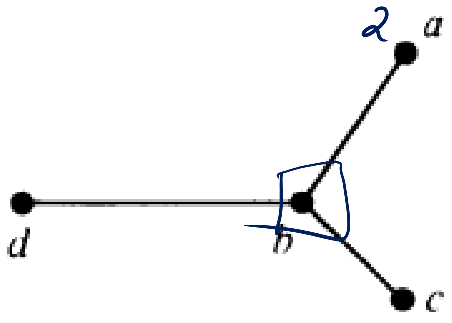
A Metric

- The function which satisfies following conditions is called a metric:
  - Nonnegativity: $f(x,y) \geq 0$ and $f(x,y) = 0$ if and only if $x = y$.
  - Symmetry: $f(x,y) = f(y,x)$.
  - Triangle inequality: $f(x,y) \leq f(x,z) + f(z,y)$ for any z.
- **Theorem 3-8**: The distance between vertices of a connected graph is a metric.

- The eccentricity E(v) of a vertex v in graph G is the distance from v to the vertex farthest from v in G,

$$E(v) = \max_{v_i \in G} d(v, v_i)$$

- A vertex with minimum eccentricity in graph G is called a center of graph G.



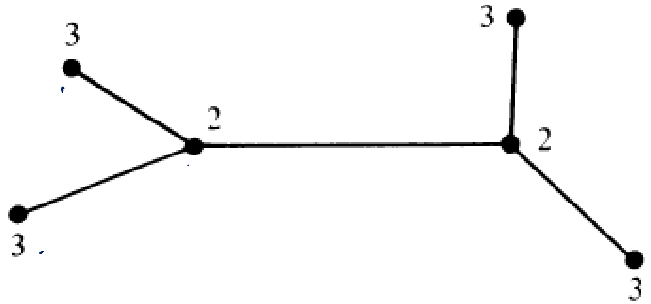
$E(a) = 2$

$E(b) = 1$

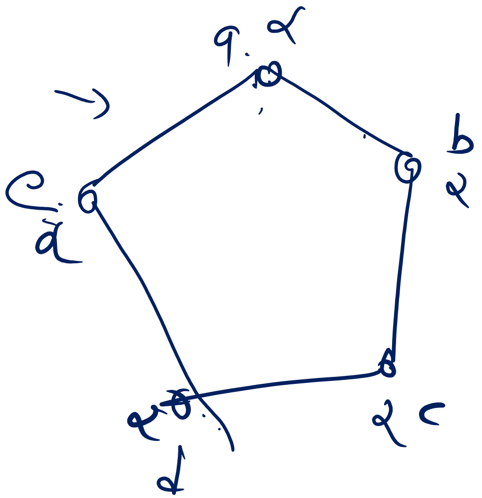
$E(c) = 2$

$E(d) = 2$

- The tree with two vertices having the same minimum eccentricity has two centers (bicenters).

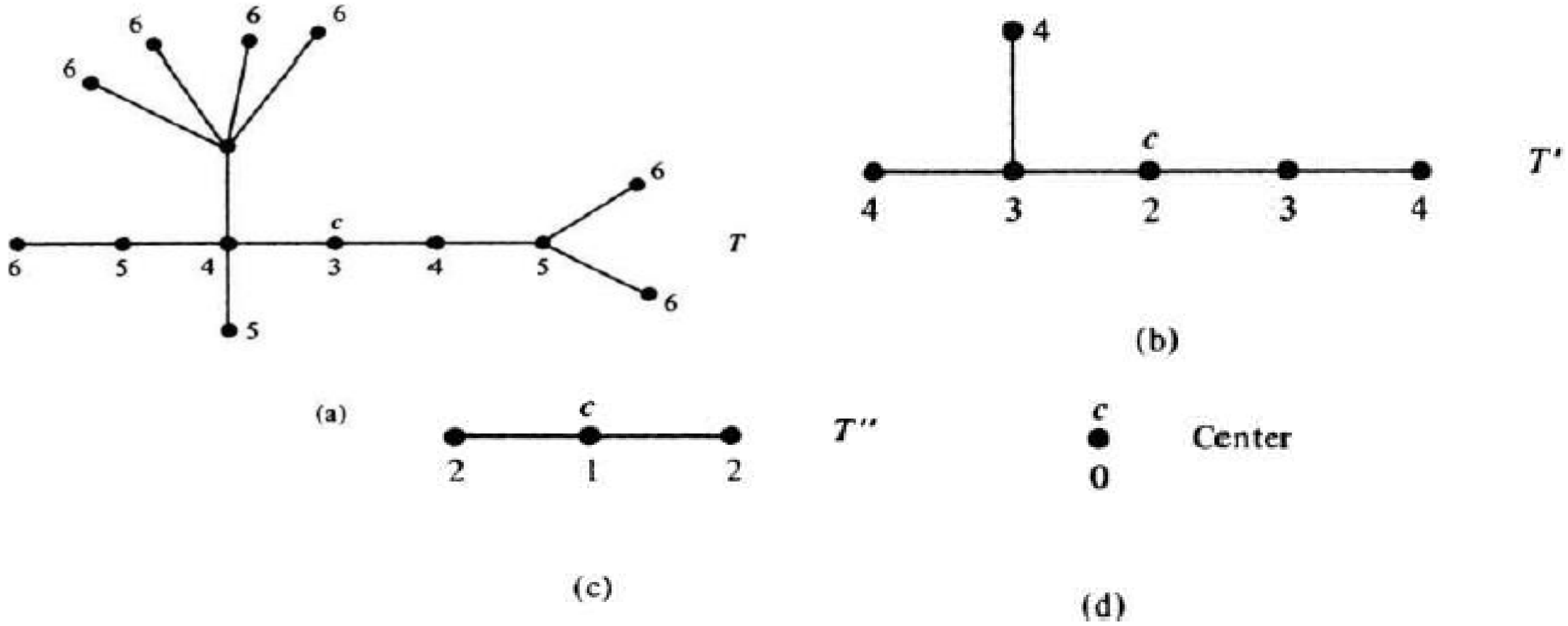


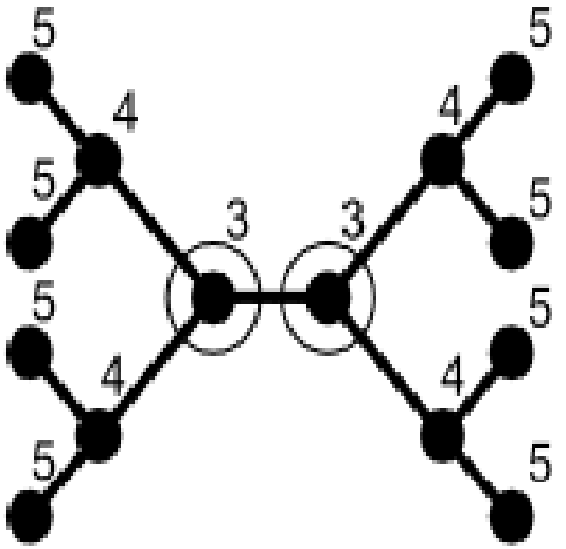
Polygon



- A graph that consists of just a circuit, has all the vertices as centers. In general, a graph has many centers.

- **Theorem 3-9**: Every tree has one or two centers.
- **Corollary**: If a tree T has two centers, the two centers must be adjacent.
- Application: Communication between group of people.

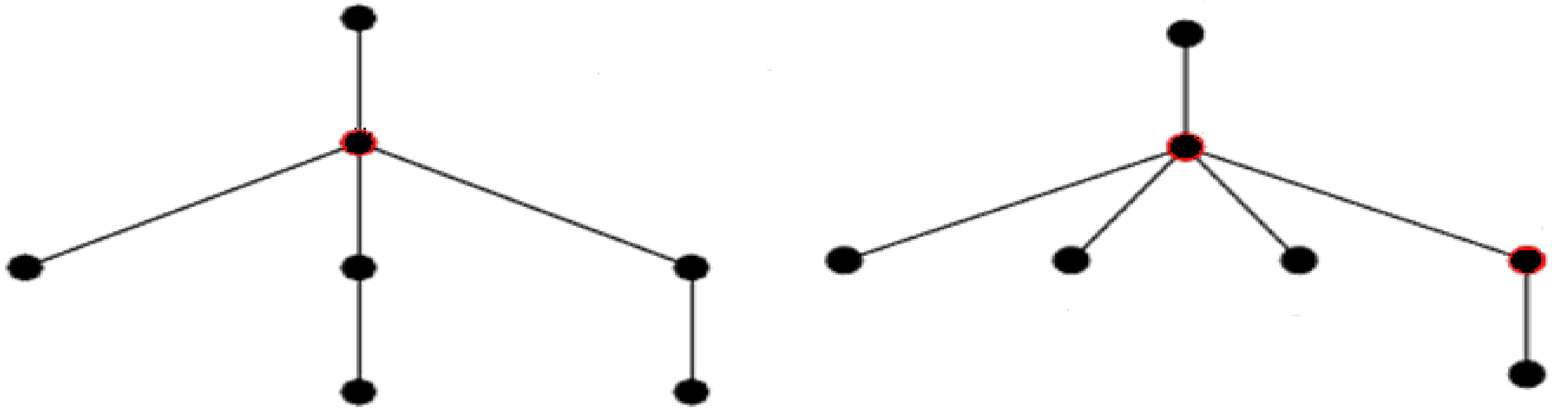




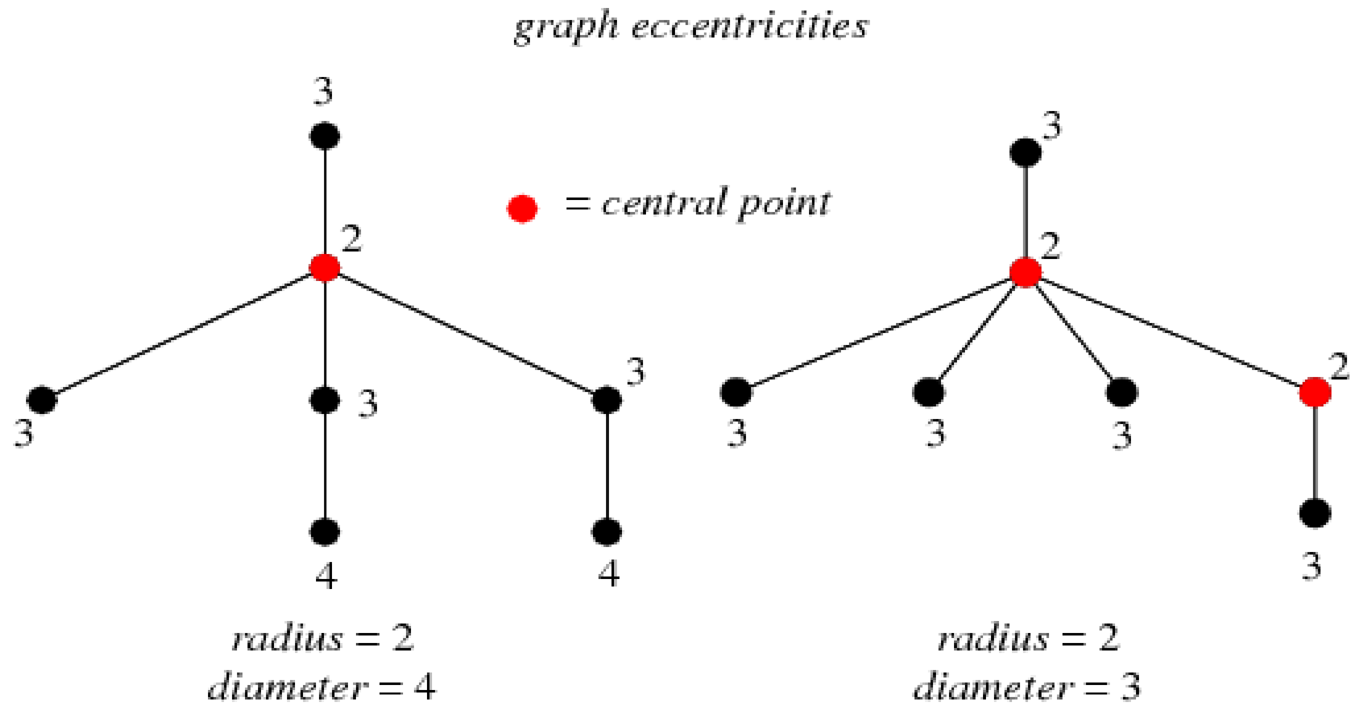
- The eccentricity of a center in a tree is defined as the radius of the tree.
- The length of the longest path in a tree is defined to be the diameter of the tree.

Example

- The eccentricity, center, radius and diameter for the following tree are as mentioned.



Solution


graph eccentricities
= central point
3
2
3
3
3
4
4
radius = 2
diameter = 4
3
2
3
3
3
2
3
radius = 2
diameter = 3

Rooted Tree

- In a tree where one vertex is distinguished from all the other is called the rooted tree.

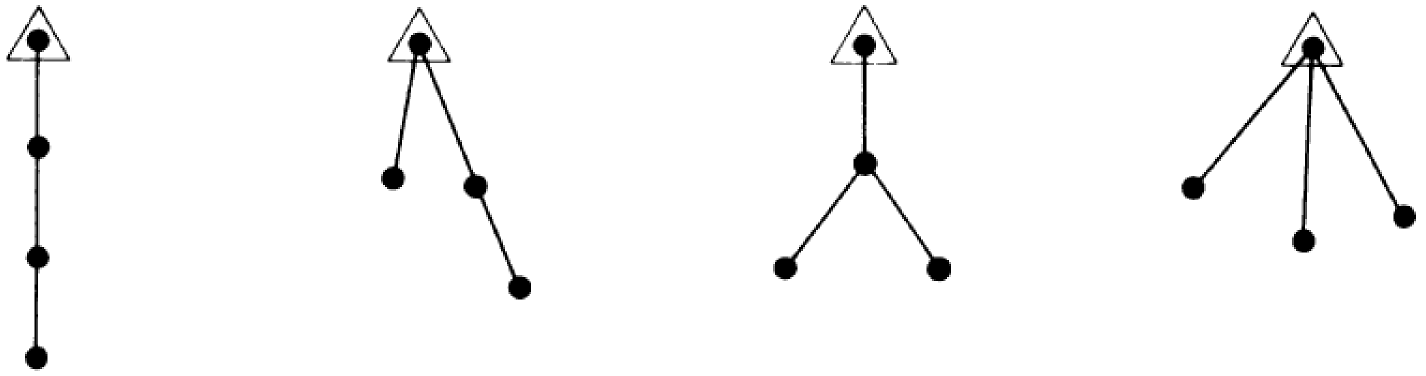


Fig. 3-11 Rooted trees with four vertices.

- Free Tree is the tree without any root.

Binary Trees

- A binary tree is defined as a special kind of rooted tree in which there is exactly one vertex of degree two and each of the remaining vertices is of degree one or three.

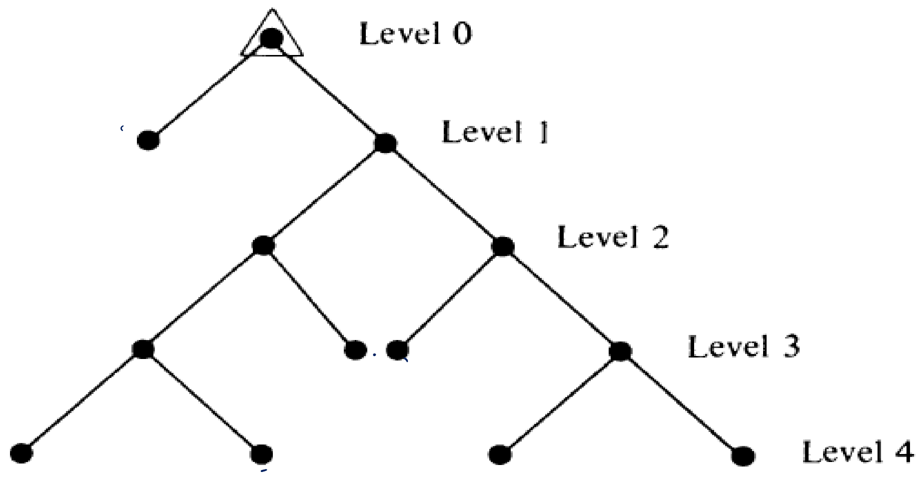


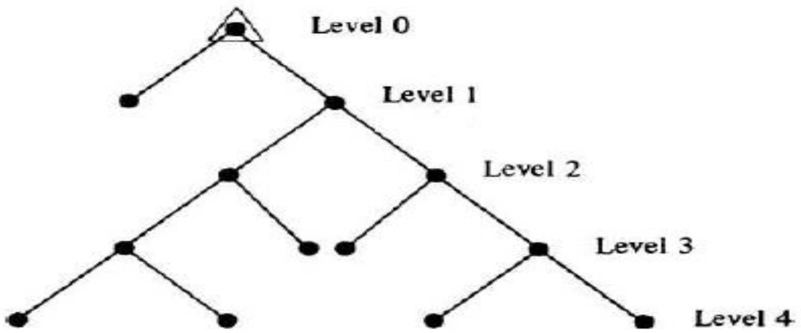
Fig. 3-12 A 13-vertex, 4-level binary tree.

Properties of binary trees

- The number of vertices *n* in a binary tree is always odd.
- Let *p* be the number of pendant vertices in a binary tree T. Then *n-p-1* is the number of vertices of degree three. Therefore the number of edges in T equals

$$\frac{1}{2}[p + 3(n - p - 1) + 2] = n - 1$$

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



- A nonpendant vertex in a tree is called an internal vertex.
- The number of internal vertices in a binary tree is one less than the number of pendant vertices.
- In a binary tree a vertex v_i is said to be at level l_i if v_i is at a distance of l_i from the root. The root is at level 0.
- The maximum number of vertices possible in a k-level binary tree is

$$2^0 + 2^1 + \ ... + 2^k = 2^{k+1} - 1 \geq n$$

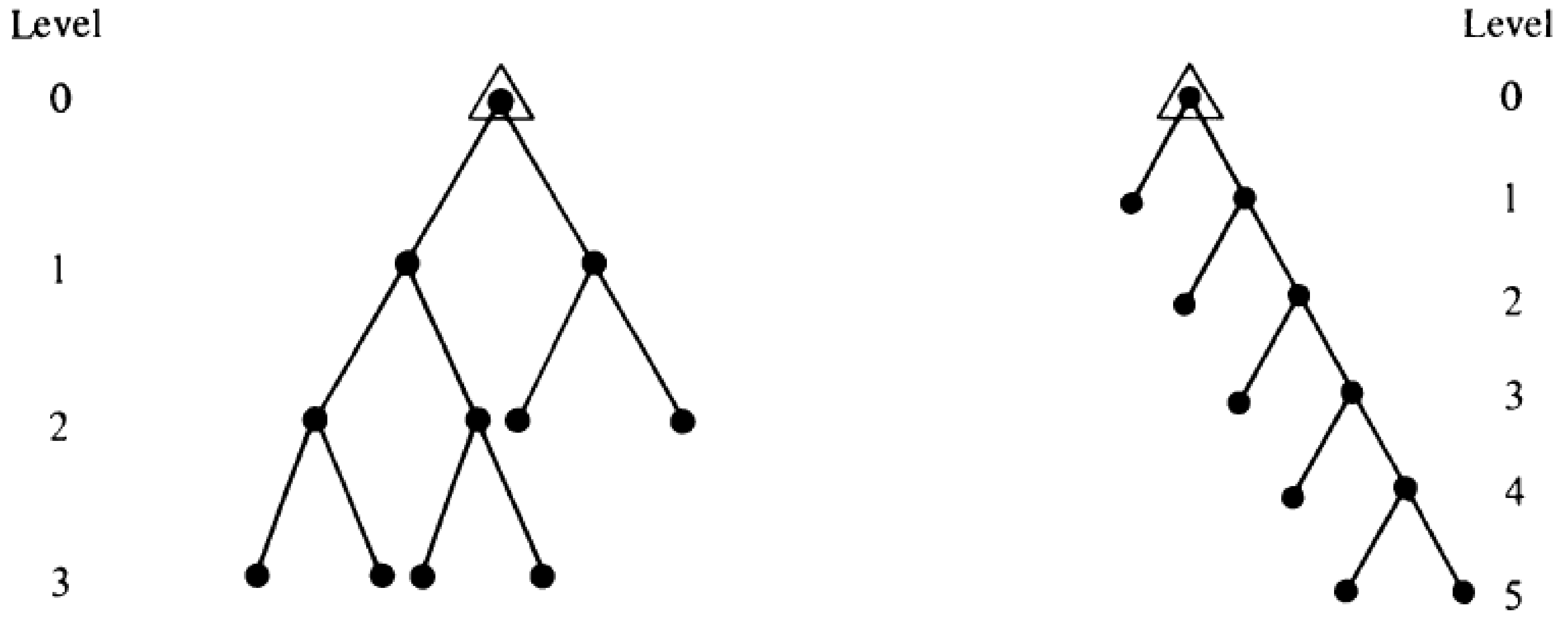
- The maximum level, l_{max} of any vertex in a binary tree is call the height of the tree.

- The minimum possible height of an n-vertex binary tree is

$$minl_{max} = \lceil log_2(n+1) - 1 \rceil$$

- The maximum possible height of an n-vertex binary tree is

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



$\min l_{\max} = \lceil (\log_2 12) - 1 \rceil$

(a)

$\max l_{\max} = \dfrac{11 - 1}{2} = 5$

(b)

Fig. 3-13 Two 11-vertex binary trees.

- The path length of a tree, can be defined as the sum of the path lengths from the root to all pendant vertices.
- The path length for the below given tree is 1 + 3 + 4 + 4 + 3 + 4 + 4 = 23.

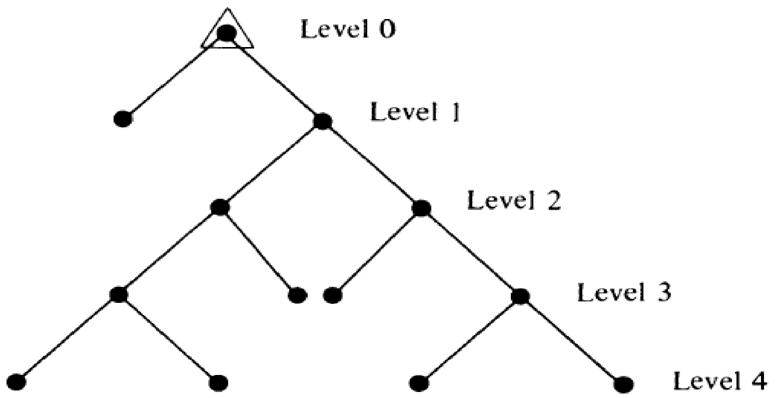


Fig. 3-12 A 13-vertex, 4-level binary tree.

- The path length of a tree is related to the execution time of an algorithm.

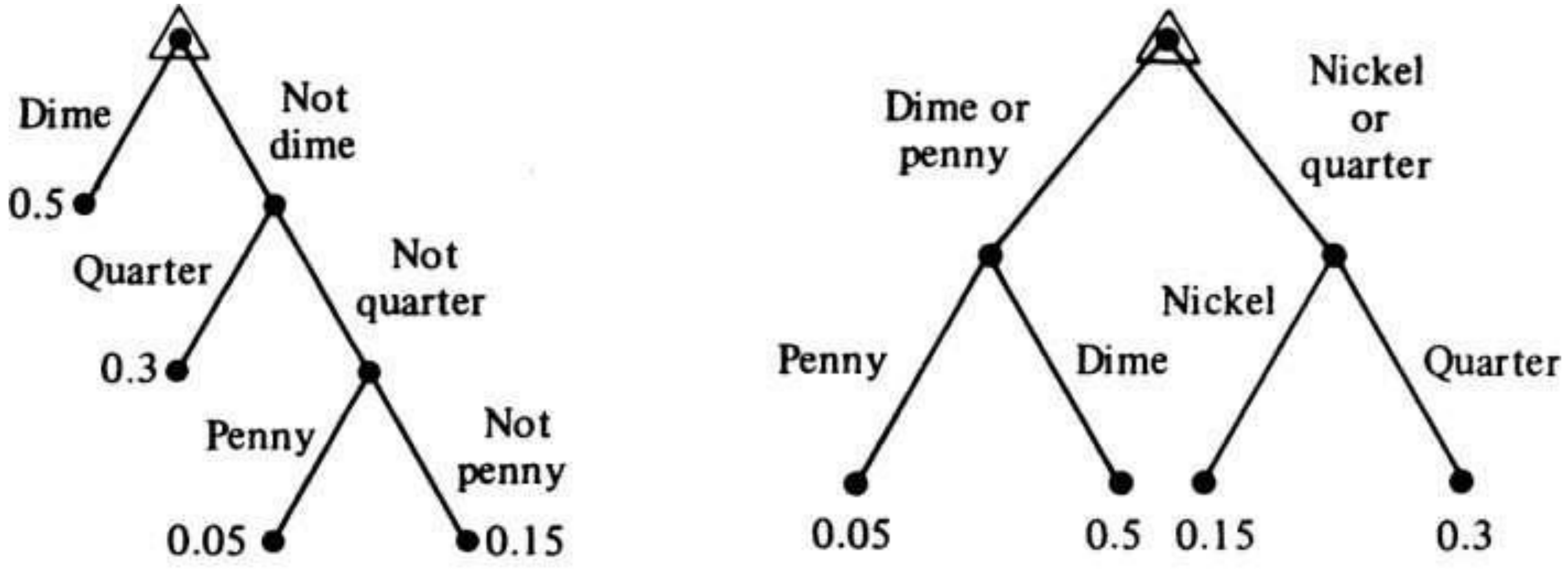
Weighted Path Length

- Every pendant vertex v_j of a binary tree has associated with it a positive real number w_j. Given w_1, w_2, … , W_m the problem is to construct a binary tree (with m pendant vertices) that minimizes

$$\sum w_j l_j$$

- Where l_j is the level of pendant vertex v_j, and the sum is taken over all pendant vertices.

Example



$\Sigma w_i \cdot l_i = 1.7$

(a)

$\Sigma w_i \cdot l_i = 2$

(b)

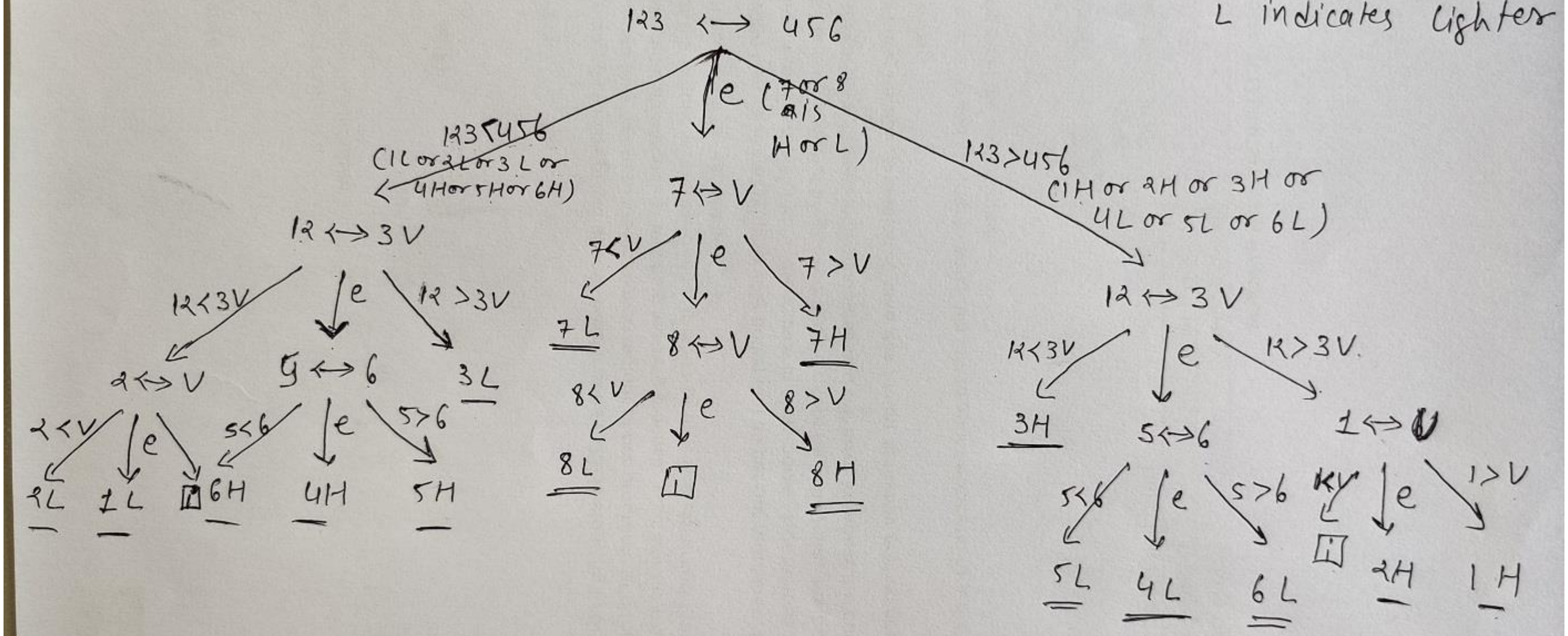
Examples

- How many edges does a full binary tree with 1000 internal vertices have?
- Suppose 1000 people enter a chess tournament. Use a rooted tree model of the tournament to determine how many games must be played to determine a champion, if a player is eliminated after one loss and games are played until only one entrant has not lost. (Assume there are no ties.)

3-9. Suppose you are given eight coins and are told that seven of them are of equal weight, and one coin is either heavier or lighter than the rest. You are provided with an equal-arm balance, which you may use only three times, for comparing coins. Sketch a strategy in the form of a decision tree for identifying the nonconforming coin, as well as for finding out whether it is heavier or lighter than the rest.

8 coins labeled as 1,2,3,4,5,6,7,8.
divide in three groups 123, 456, 78.

<-> indicates comparison, V indicates valid coin, e indicates equal, H indicates Heavier
L indicates lighter



- Given N coins, all may be genuine or only one coin is defective. We need a decision tree with atleast (2N + 1) leaves correspond to the outputs. Because there can be N leaves to be lighter, or N leaves to be heavier or one genuine case, on total (2N + 1) leaves.
- As explained earlier ternary tree at level k, can have utmost 3^k leaves and we need a tree with leaves of $3^k > (2N + 1)$.
- *In other words, we need atleast $k > \log_3(2N + 1)$ weighing to find the defective one.*

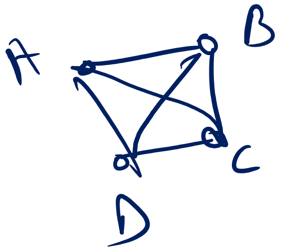
Spanning Trees

- A tree is a subgraph of another graph. A graph has numerous subgraphs – from e edges, 2^e distinct combination are possible.
- Some of these subgraphs will be trees. Out of these trees particular interest is to find the spanning trees.
- A tree T is said to be a spanning tree of a connected graph G if T is a subgraph of G and T contains all vertices of G. Spanning tree represents a sort of skeleton of the original graph G, this is why it is also referred to as a skeleton or scaffolding of G.
- Applications include
  - Cluster Analysis
  - Message Broadcasting
  - Image Registration
  - Image Segmentation
  - Feature Extraction
  - Handwriting Recognition

On Counting Trees

- Arthur Cayley discovered trees while trying to count the number of structural isomers of the saturated hydrocarbons C_kH_{2k+2}.
- Corresponding to their chemical valences, a carbon atom was represented by a vertex of degree four and a hydrogen atom by a vertex of degree one (pendent vertices).
- The total number of vertices in such a graph is
$$n = 3k + 2$$
- And total number of edges is
$$e = \frac{1}{2}(sum\ of\ degrees) = \frac{1}{2}(4k + 2k + 2) = 3k + 1$$

- Since the graph is connected and the number of edges is one less than the number of vertices, it is a tree.
- Thus the problem of counting isomers of a given hydrocarbon becomes the problem of counting trees (with certain qualifying properties).
- The question: What is the number of different trees that one can construct with n distinct (or labeled) vertices?



- n = 4.

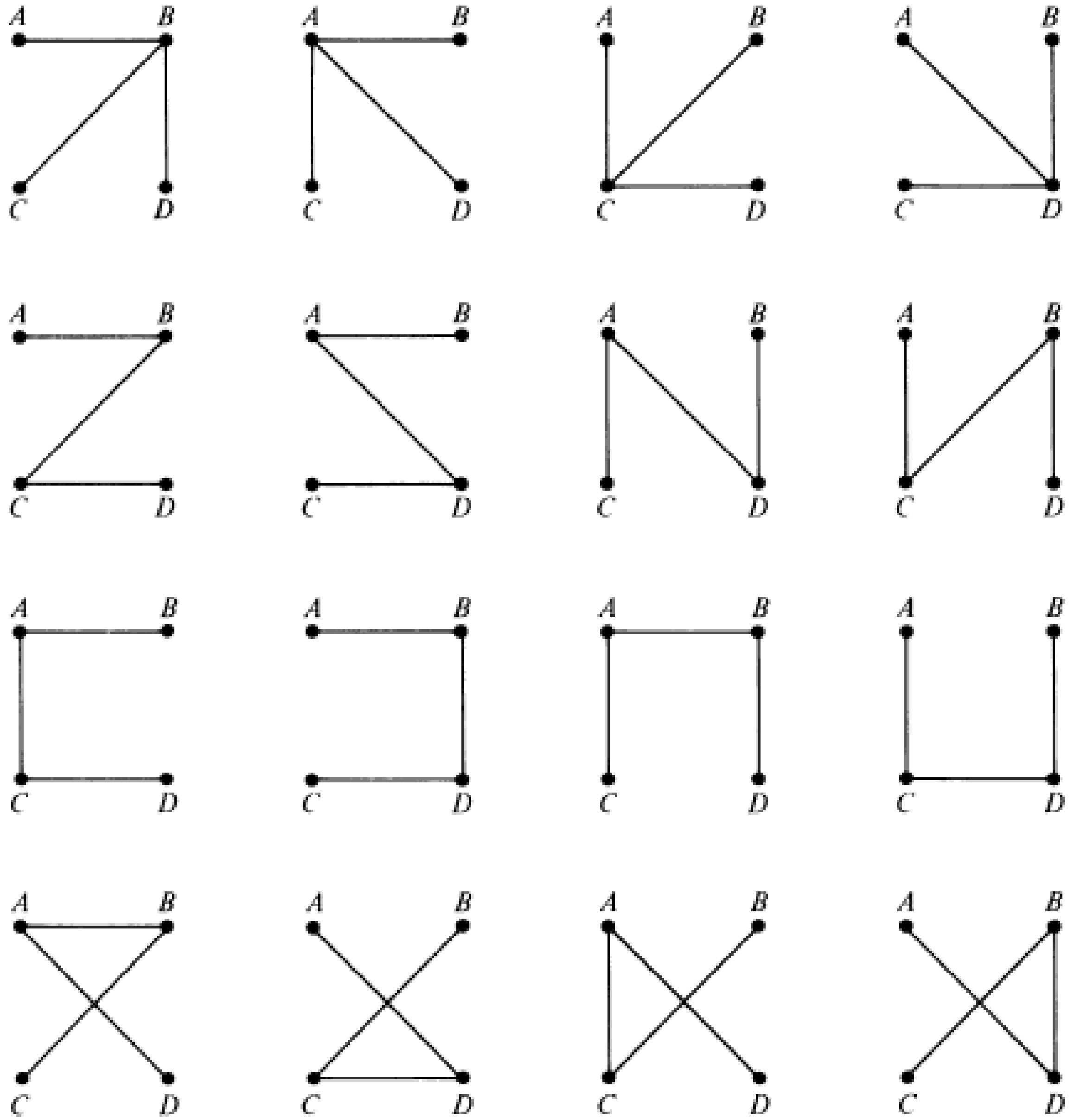


Fig. 3-15 All 16 trees of four labeled vertices.

- A graph in which each vertex is assigned a unique name or label as shown in the fig. 3.15 is called a labeled graph.
- The difference between labeled and unlabelled graph is very important when we are counting the number of different graphs.
- No. of different graphs with 4 unlabelled vertices are only two.

Theorem:

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

Cayley's Formula

- **Theorem**: There are n^{n-2} labeled trees with n vertices ($n \geq 2$).
- Proof: Let the *n* vertices of a tree T be labeled 1,2,3,….,n.
- Remove the pendant vertex (and the edge incident on it) having the smallest label, which is, say a_1. Suppose that b_1 was the vertex adjacent to a_1. Add b_1 to the resultant sequence.
- Among the remaining n-1 vertices let a_2 be the pendant vertex with the smallest label, and b_2 be the vertex adjacent to a_2. Remove the edge (a_2,b_2). Add b_2 to the resultant sequence.
- This operation is repeated on the remaining n-2 vertices and the process is stopped after n-2 steps, when only two vertices are left.
- The tree uniquely defines the sequence

$$(b_1,b_2,\ldots, b_{n-2}) \rightarrow (1)$$

- E.g.

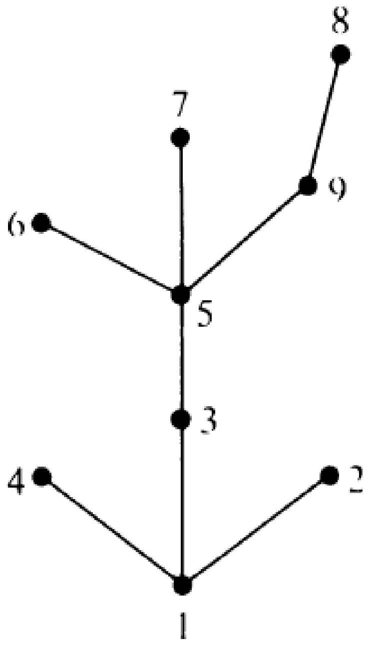


Fig. 10-1 Nine-vertex labeled tree, which yields sequence (1, 1, 3, 5, 5, 5, 9).

- Conversely, given a sequence n-2 labels, an n-vertex tree can be constructed uniquely, as follows:

$$1,2,3,\ldots,n \rightarrow (2)$$

- Determine the first number from the sequence (2) that does not appear in sequence (1).
- This number clearly is a_1. And thus the edge (a_1,b_1) is defined.
- Remove b_1 from (1) and a_1 from (2).
- Repeat the process until the sequence (1) is empty.
- Finally the last two vertices in (2) are joined.

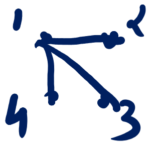
- E.g. Given the sequence (4,4,3,1,1), construct the spanning tree.

Construct all the spanning trees using Cayley's formula for $\underline{K_4}$.

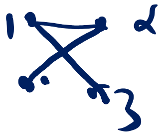
$K_5 = 125$

$4^{4-2} = 16$ prüfer sequences are:

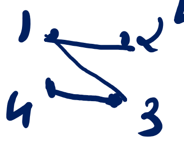
$(\cancel{1},\cancel{1}) \Rightarrow (\cancel{1},\cancel{2},\cancel{3},4)$



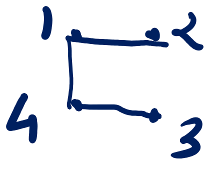
$(\cancel{1},\cancel{2}) \Rightarrow (\cancel{1},2,\cancel{3},4)$



$(\cancel{1},\cancel{3}) \Rightarrow (\cancel{1},\cancel{2},3,4)$



$(\cancel{1},\cancel{4}) \Rightarrow (\cancel{1},\cancel{2},3,4)$



$(2,1) \Rightarrow (1,2,3,4)$

$(2,2) \Rightarrow (1,2,3,4)$

$(2,3) \Rightarrow (1,2,3,4)$

$(2,4) \Rightarrow (1,2,3,4)$

$(3,1) \Rightarrow (1,2,3,4)$

$(3,2) \Rightarrow (1,2,3,4)$

$(3,3) \Rightarrow (1,2,3,4)$

$(3,4) \Rightarrow (1,2,3,4)$

$(4,1) \Rightarrow (1,2,3,4)$

$(4,2) \Rightarrow (1,2,3,4)$

$(4,3) \Rightarrow (1,2,3,4)$

$(4,4) \Rightarrow (1,2,3,4)$

Construct the spanning tree from the Prufer code (1,1,3,5,5,5,9).

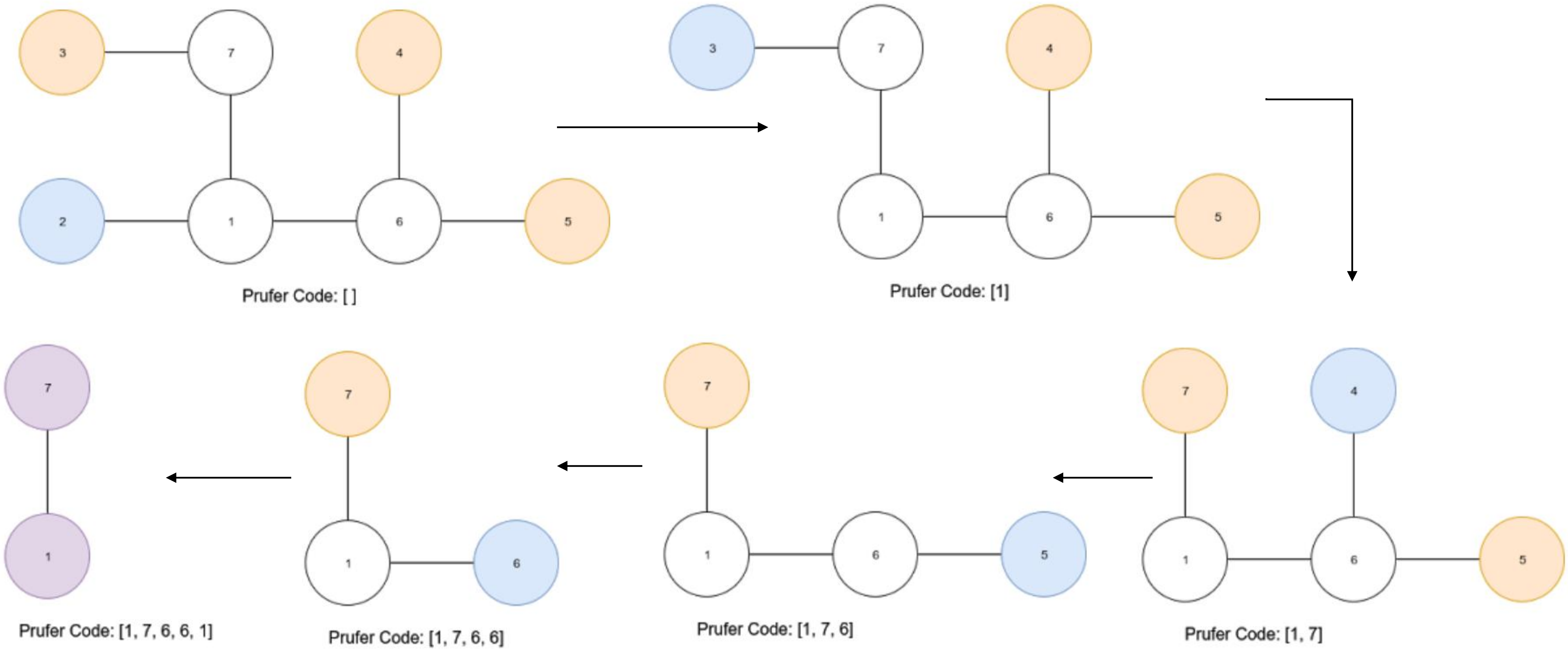
Generating Prufer Code from the tree

1. Find the smallest leaf node of the given tree. Let it be x. Let the neighbor of x be y.
2. Add the value of y in a list.
3. Remove node x from the tree.
4. Repeat step 1 to 3 until there are only 2 nodes remaining.
5. The list contains the Prufer Code.

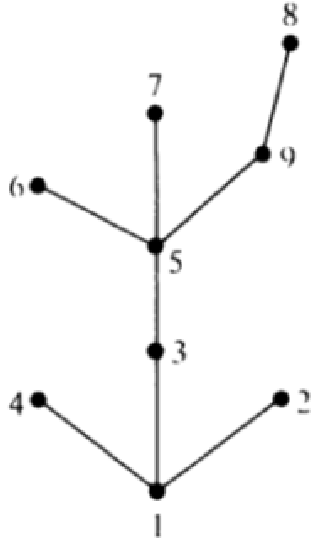
[Prufer Code: Linear Representation of a Labeled Tree | forthright48]

Example

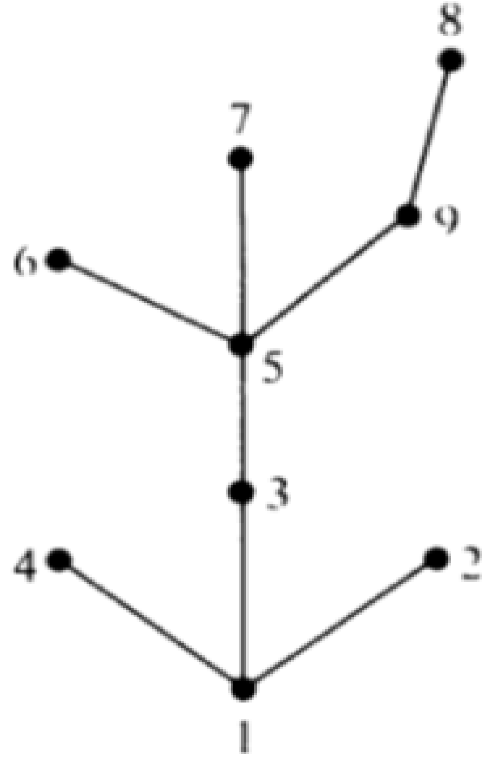
Here, the white nodes are non-leaf nodes, the orange nodes are leaf nodes and the blue nodes are the smallest leaf node for the tree.



Example: Generate the Prufer code from the given spanning tree.



Example: Generate the Prufer code from the given spanning tree.



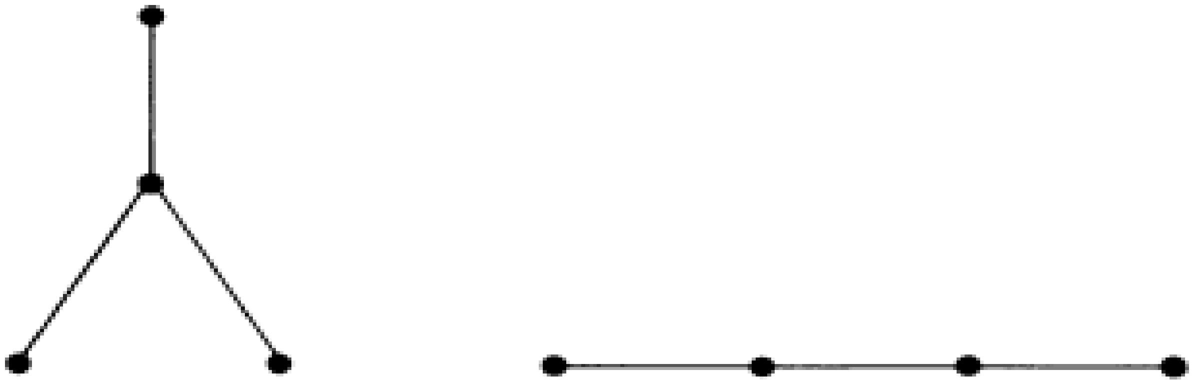
The Prufer Code sequence is (1,1,3,5,5,5,9)

Properties of Prufer Code

1. If a node has degree d, then that node will appear in prufer code exactly d−1 times.
2. Leaves never appear in Prufer Code.

- Unlabeled Trees: In actual counting of isomers of C_kH_{2k+2}, given theorem is not enough. In addition to the constraints on the degree of the vertices, two observations should be made:
  1. Since the vertices representing hydrogen are pendant, they go with carbon atom only one way, and hence make no contribution to isomerism. Therefore, we need not show any hydrogen vertices.
  2. Thus the tree representing C_kH_{2k+2} reduces to one with k vertices, each representing a carbon atom. In this tree no distinction can be made between vertices, and therefore the it is unlabelled.

- For butane C_4H_{10}, there are only two distinct trees representing n-butane and isobutane.
-



(a) (b)

Fig. 3-16 All trees of four unlabeled vertices.

- As spanning trees are the largest (with maximum number of edges) trees among all trees in G, they are also referred to as ***maximal tree subgraphs*** or ***maximal trees*** of G.
- The spanning tree is defined only for a connected graph, because a tree is always connected.
- Each component in a disconnected graph does have a spanning Tree. A disconnected graph with k components has a *spanning forest* of *k spanning trees*.

Finding spanning tree from a graph

- If a connected graph G has no circuit, it is its own spanning tree.
- If G has a circuit, delete an edge from the circuit. The edge removal does not affect the connectivity of a graph. If there are more circuits, repeat the operation till an edge from the last circuit is deleted which results in circuit-free connected graph with all the vertices of G, which is a spanning tree.
- **Theorem:** Every connected graph has at least one spanning tree.

- Branch: An edge in a spanning tree T is called a branch of T.
- Chord: An edge of G that is not in a given spanning tree T is called a chord (tie or link).

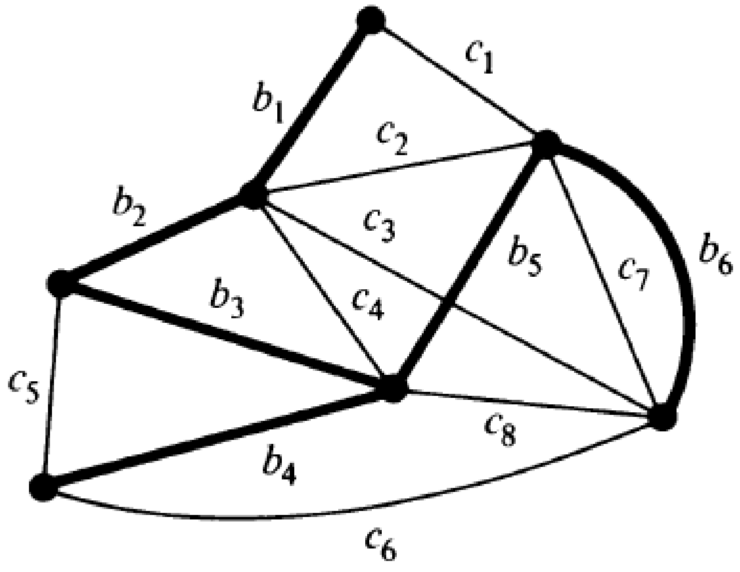


Fig. 3-17 Spanning tree.

- Branches and chords are defined with respect to a given spanning tree. An edge that is a branch of one spanning tree T_1 may be a chord with respect to another spanning tree T_2.
- The graph G is a union of two subgraphs, T and T’

$$T \cup T' = G$$

- Where T is a spanning tree, and T’ (chord set or tie set or cotree) is the complement of T in G.
- **Theorem**: With respect to any of its spanning trees, a connected graph of *n* vertices and *e* edges has *n-1* tree branches and *e-n+1* chords.

Example

- A farm consisting of six walled plots of land, as shown in the figure. All the plots are full of water, how many walls will have to be broken so that all the water can be drained out?

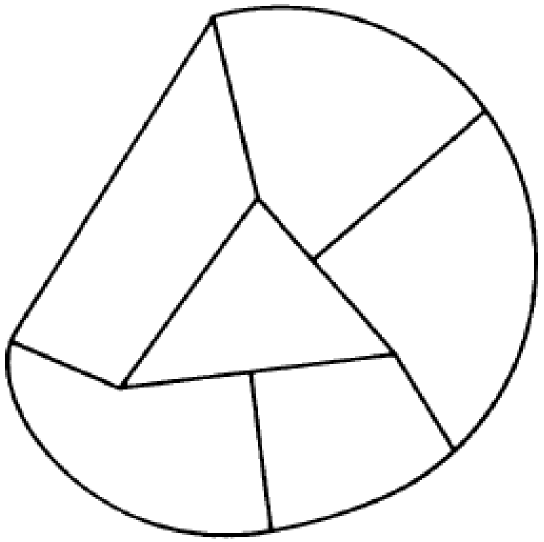


Fig. 3-18 Farm with walled plots of land.

- Let’s relate the fundamental numbers of graph: n (no of vertices), e (no of edges) and k (no of connected components) of a graph G.
- Every component of a graph must have at least one vertex, $n \geq k$ $(n - k \geq 0)$.
- The number of edges in a component can not be less than the number of vertices in that component minus one, $e \geq n - k$ $(e - n + k \geq 0)$.
- n, e, and k define
- **Rank** $r = n - k$
- **Nullity** $\mu = e - n + k$.

- The rank of a connected graph is n-1 and nullity, e-n+1.
- Rank of G = number of branches in any spanning tree (or forest) of G,
- Nullity of G = number of chords in G,
- Rank + nullity = number of edges in G.
- The nullity of a graph is also referred to as its *cyclomatic number* or *first Betti number.*

Fundamental Circuits

- **Theorem**: A connected graph G is a tree if and only if adding an edge between any two vertices in G creates exactly one circuit.
- Consider a spanning tree T in a connect graph G. Adding any one chord to T will create exactly one circuit. Such a circuit, formed by adding a chord to a spanning tree, is called a *fundamental circuit*.
- How many fundamental circuits does a graph have?
- How many circuits does a graph have in all?
- Suppose that we add one more chord. Will it create exactly one more circuit? What happens if we add all the chords simultaneously to the tree?

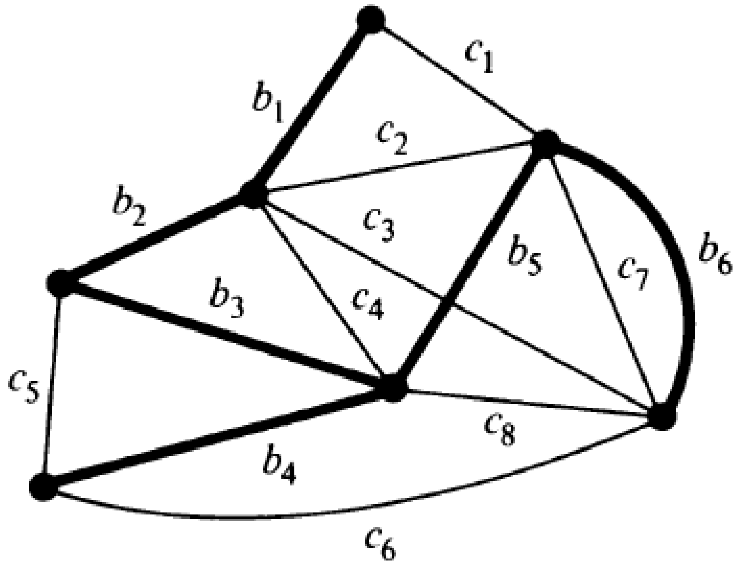


Fig. 3-17 Spanning tree.

- Find all fundamental circuits w.r.t. the given spanning tree.

Comments

- A circuit is a fundamental circuit only with respect to a given spanning tree. A given circuit may be fundamental with respect to one spanning tree, but not with respect to a different spanning tree of the same graph.
- Kirchhoff showed that: no matter how many circuits a graph contains, there is a need to consider/find only fundamental circuits with respect to spanning trees. The rest of the circuits are combinations of some fundamental circuits.

Questions

- What is the nullity of a complete graph with n vertices?
- Prove that any circuit in a graph G must have at least one edge in common with a chord set.
- Can you construct a graph if you are given all its spanning trees? How?
- Prove that the nullity of a graph does not change when you either insert a vertex in the middle of an edge, or remove a vertex of degree two by merging two edges incident on it.
- Show that Hamiltonian path is a spanning tree.

Finding all spanning trees of a graph

- Start with a given spanning tree, say T_1. Add a chord, h to the tree T_1, which forms a fundamental circuit (b c h d). Remove any branch say *c*, from the fundamental circuit *bchd*, to create new spanning tree T_2.

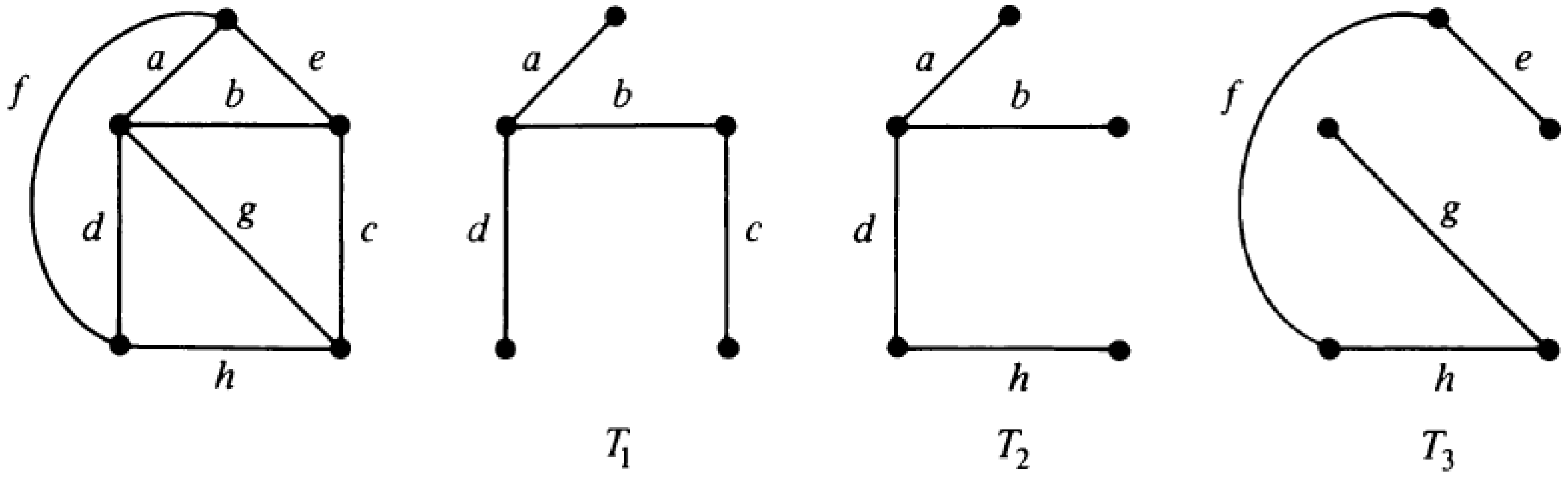


Fig. 3-19 Graph and three of its spanning trees.

- This generation of one spanning tree from another, through addition of a chord and deletion of an appropriate branch, is called a *cyclic interchange* or *elementary tree transformation*.
- After generating tree new spanning trees with a single chord, restart with T_1 and add a different chord (e, f or g) and repeat the process of obtaining a different spanning tree each time a branch is deleted from the fundamental circuit formed.

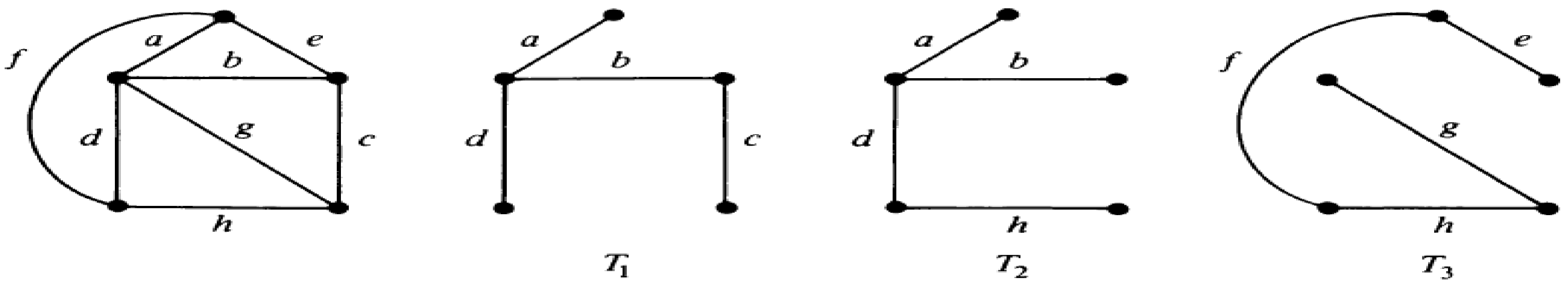


Fig. 3-19 Graph and three of its spanning trees.

Questions

- Can we start from any spanning tree and get a desired spanning tree by a number of *cyclic exchanges*?
- Can we get all the spanning trees of a given graph using *cyclic exchanges*?
- How long we have to continue exchanging edges?
- Out of all possible spanning trees that we can start with, is there a preferred one for starting?

- *The distance between two spanning trees* T_i and 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 distance may be written as $d(T_i, T_j)$.
- Let $T_i \oplus T_j$ be the ring sum of two spanning trees T_i and T_j of G. Let N(g) denote the number of edges in a graph g. Then

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

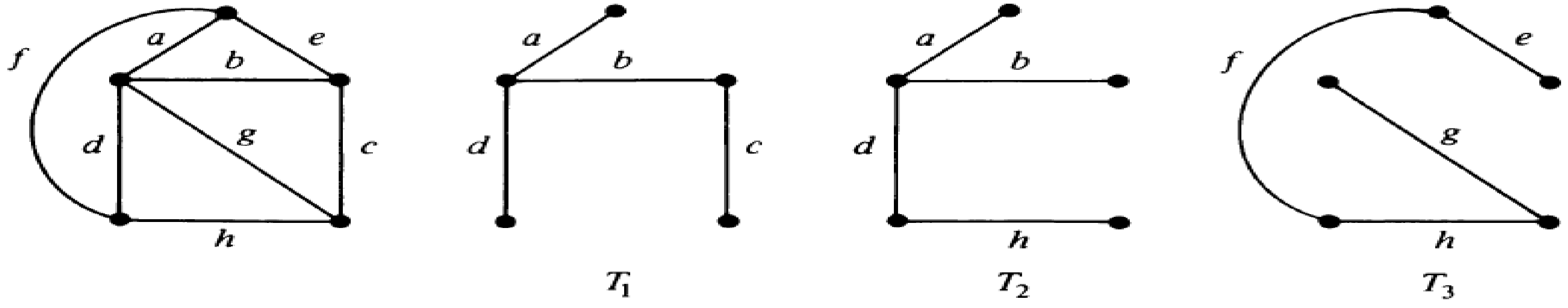


Fig. 3-19 Graph and three of its spanning trees.

- $d(T_i, T_j)$ is the minimum number of cyclic interchanges involved in going from T_i to T_j.
- **Theorem**: The distance between the spanning trees of a graph is a *metric*.
- **Theorem**: Starting from any spanning tree of a graph G, we can obtain every spanning tree of G by successive cyclic exchanges.

- For a connected graph G of rank *r*(*r+1* vertices) a spanning tree has r edges, following results can be derived:

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

$$\max d(T_i, T_j) \leq r, the\ rank\ of\ G.$$

- If μ is the nullity of G, no more than μ edges of a spanning tree T_i, can be replaced to get another tree T_j.

$$\max d(T_i, T_j) \leq \mu$$

- Combining the two

$$\max d(T_i, T_j) \leq \min(\mu, r)$$

Out of all possible spanning trees that we can start with, is there a preferred one for starting?

- **Central Tree:** For a spanning tree T_0 of a graph G, let $\max_i d(T_0, T_i)$ denote the maximal distance between T_0 and any other spanning tree of G, Then T_0 is called a *central tree* of G if

$$\max d(T_0, T_i) \leq \max_j d\left(T, T_j\right) \text{ for every tree T of G.}$$

- A central tree in a graph is in general not unique.
- ***Tree Graph***: The tree graph of a given graph G is defined as a graph in which each vertex corresponds to a spanning tree of G, and each edge corresponds to a cyclic interchange between the spanning trees of G represented by the two end vertices of the edge.
- The tree graph of any finite and complete graph is connected.

Draw a tree graph for K4.

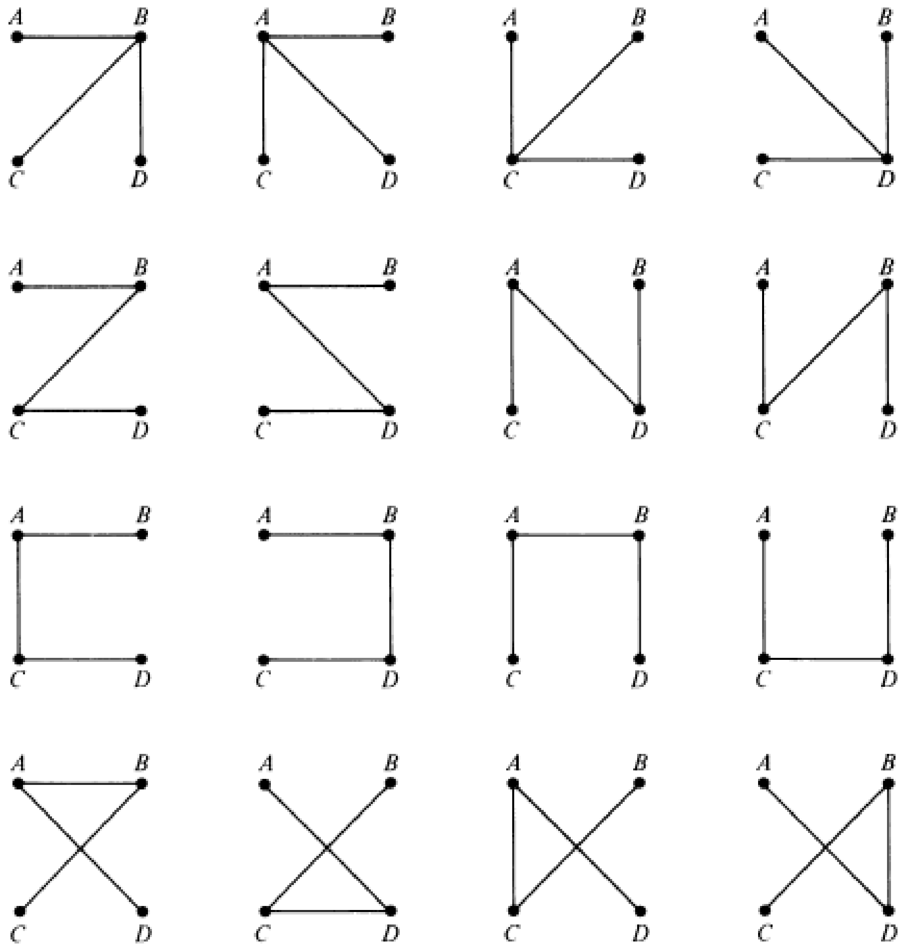
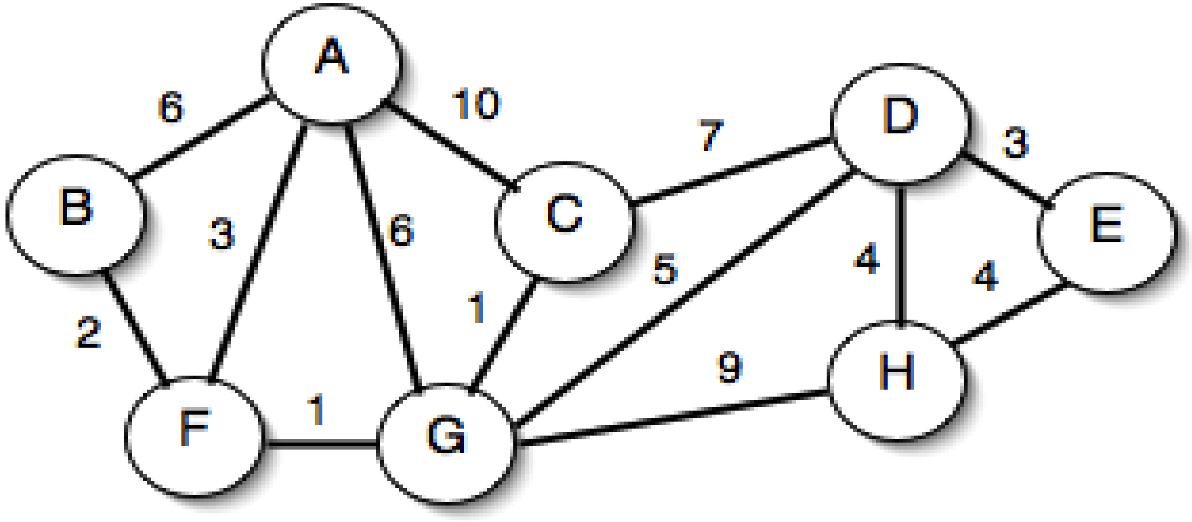
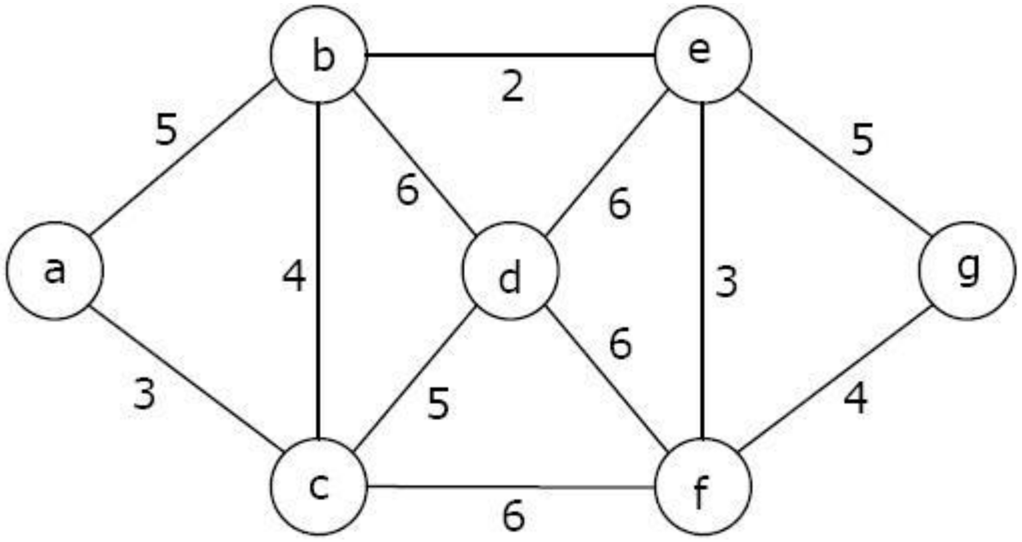


Fig. 3-15 All 16 trees of four labeled vertices.

Spanning trees in a weighted graph

A spanning tree with the smallest weight in a weighted graph is called a shortest spanning tree or shortest-distance spanning tree or minimal spanning tree.





- **Theorem:** Let (G, W) be a connected weighted graph. A spanning tree T is a minimum cost spanning tree if and only if there exists no other spanning tree T' of distance one from T and with weight W(T') less than W(T).
- Proof: Let T1 be a spanning tree in G satisfying the hypothesis(there is no spanning three at a distance of one from T1 which is shorter than T1).

Degree Constrained Shortest Spanning Tree

- In a shortest spanning tree, any vertex v_i can end up with any degree; i.e. $1 \leq d(v_i) \leq n-1$. In some practical cases an upper limit on the degree of every vertex has to be imposed, e.g. an electrical wiring problem. Thus, in this particular case,

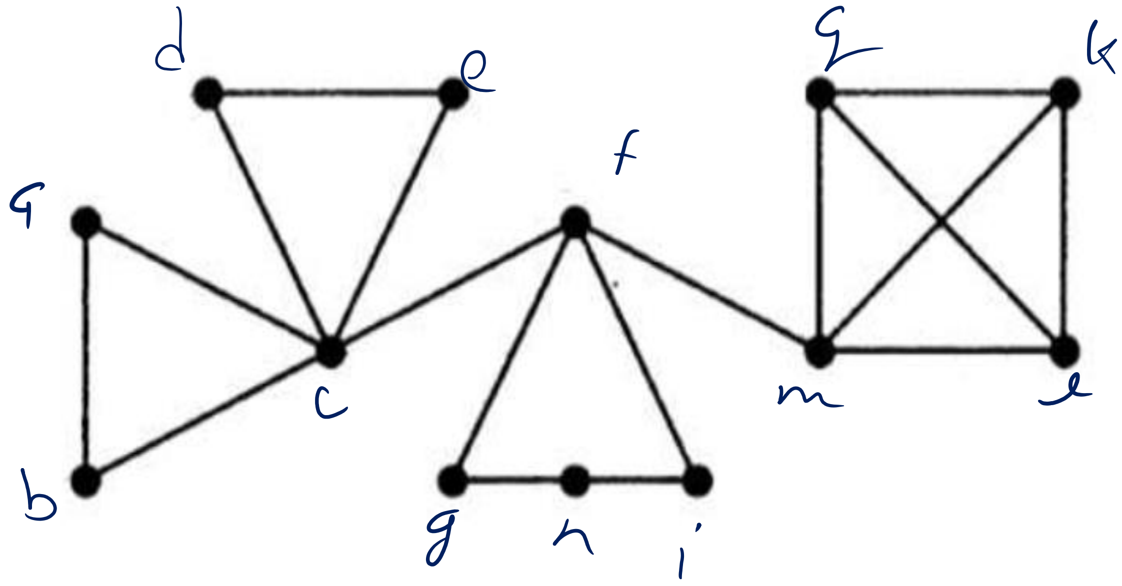
$$d(v_i) \leq k \text{ for every vertex } v_i \text{ in } T.$$

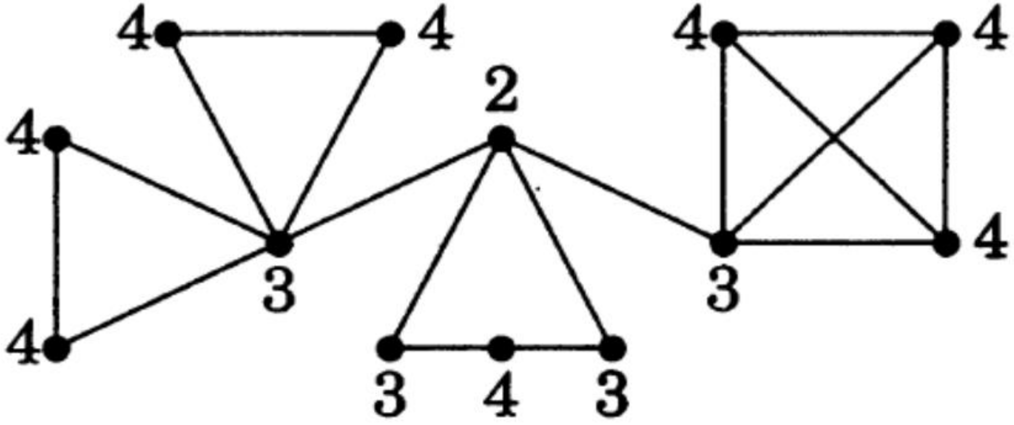
Such a spanning tree is called a degree-constrained minimum spanning tree.

- If $k = 2$, this problem reduces to the problem of finding the shortest Hamiltonian path, as well as travelling-salesman problem without the salesman returning to the home city.

Questions

- Find the center, the radius and diameter of the below given graph. Also mention the eccentricity of all vertices.





If T, T' are spanning trees of a connected graph G and $e \in E(T)-E(T')$, then there is an edge $e' \in E(T')-E(T)$ such that _____________ is a spanning tree of G.

1. T-e
2. T-e+1
3. T-e+e’
4. T+e-e’

- Every graph with n vertices and k edges has atleast k components.
    - True
    - False

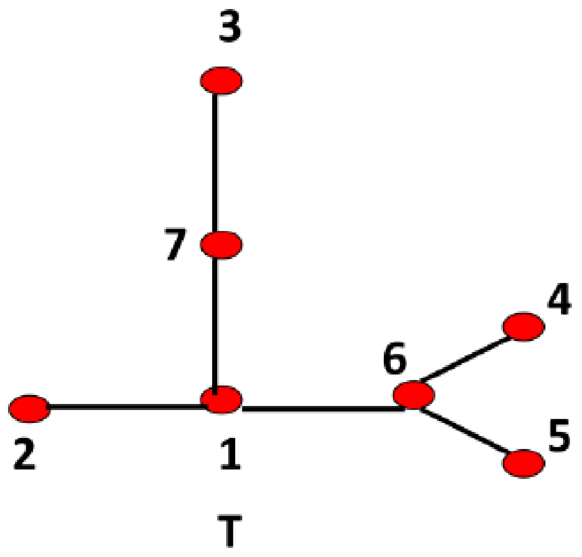
Questions

- Match the following

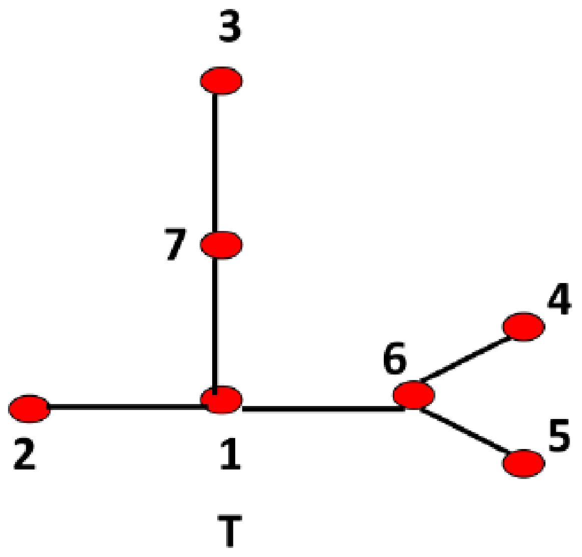
1) Diameter of a graph G	P) Upper bound of the distance from vertex u to the others.
2) Eccentricity of a vertex u	Q) Upper bound of distance between every pair
3) Radius of a graph G	R) Lengths of paths
4) Distance	S) Lower bound of the eccentricity.

1-Q, 2-P, 3-S,4-R

Find the Prufer sequence for the given spanning tree.



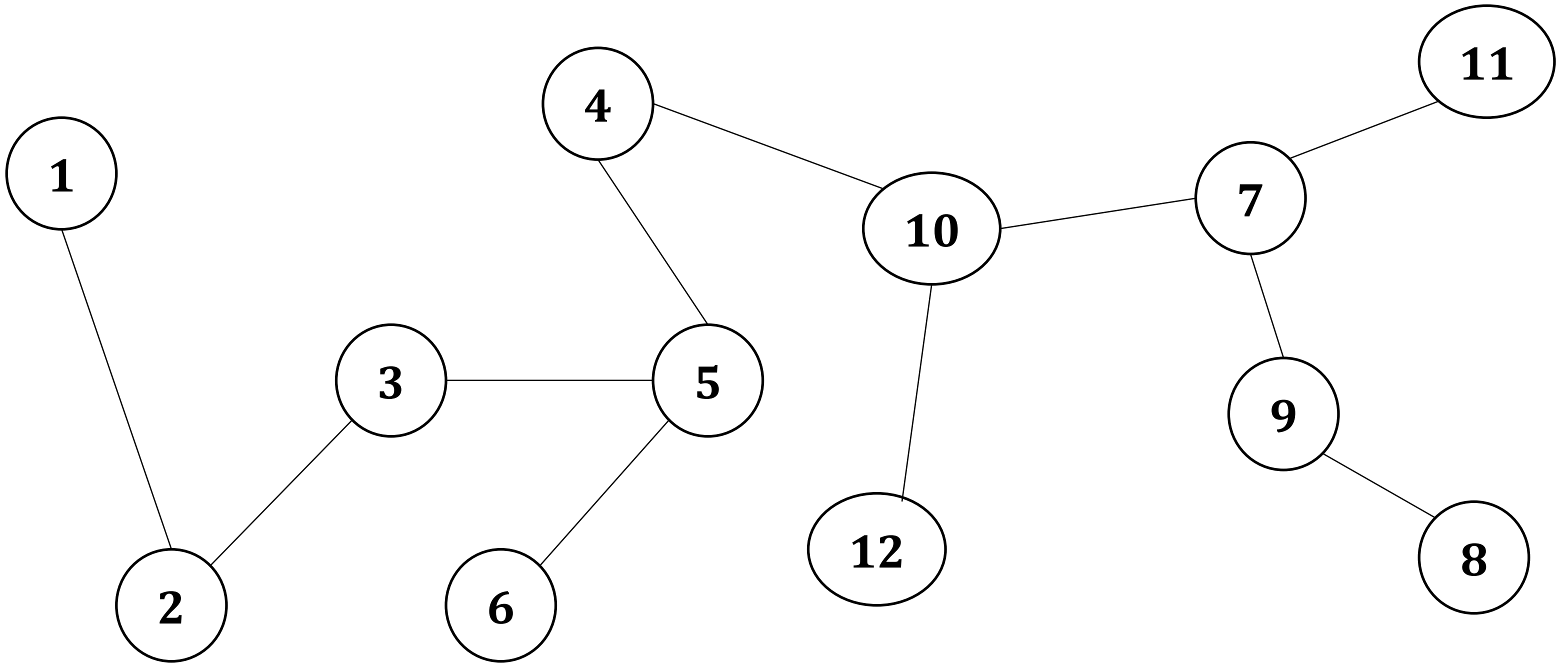
Find the Prufer sequence for the given spanning tree.



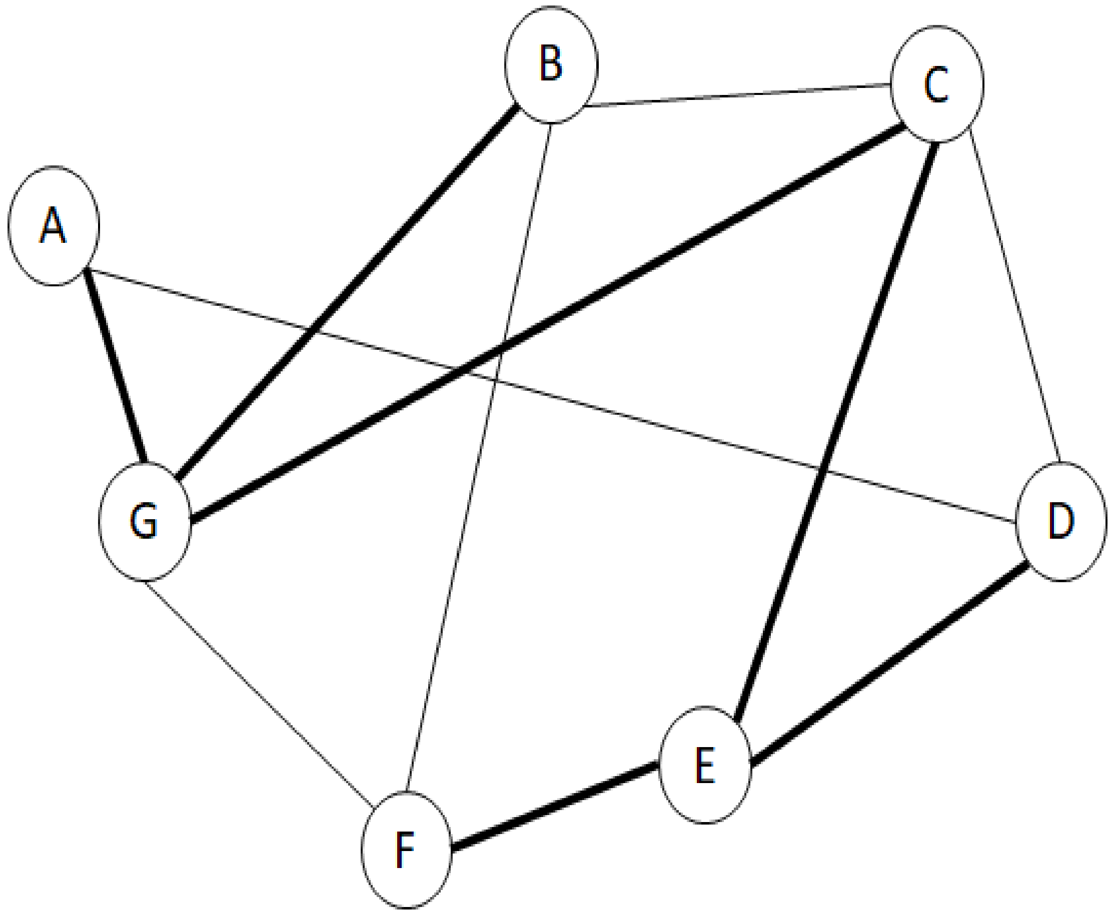
S = (1, 7, 6, 6, 1)

Generate the spanning tree using the prufer sequence (1,2,1,3,3,5)

Generate the prufer code from the given spanning tree.



Find a spanning tree at a distance of five from spanning tree as shown in the graph. List all fundamental circuits with respect to the new spanning tree.



G is an undirected graph with n vertices and 25 edges such that each vertex of G has degree at least 3. What is the maximum possible value of n?

- No. of Vertices are: 16.

Count the number of edges in a regular graph of n vertices with degree d.

- No. of edges in a regular graph of n vertices with degree d = (n*d)/2.