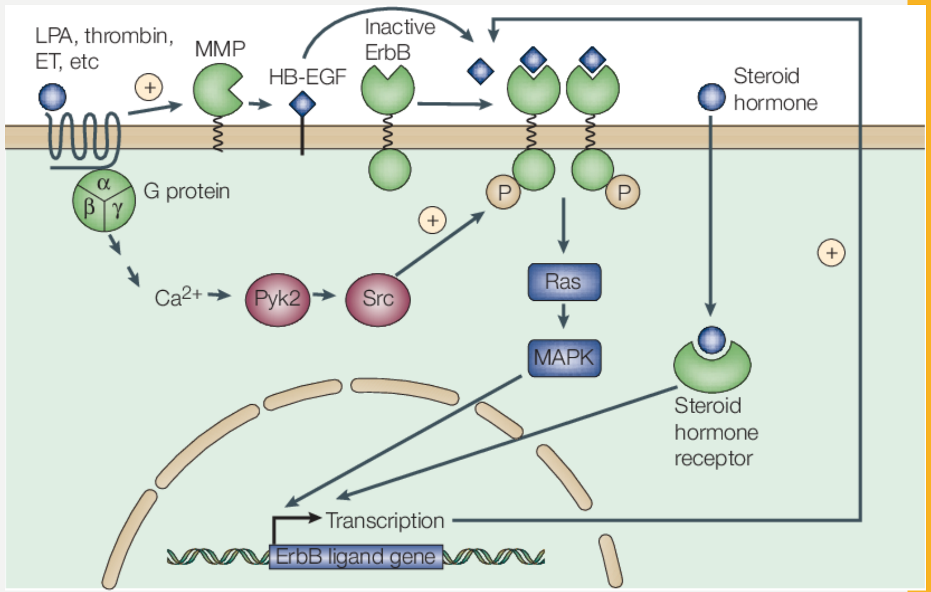
BASICS OF GRAPH & NETWORK COMPARISONS

Text Book: Graph Theory with Applications to Engineering and Computer Science
Narasingh Dev, Prentice Hall, New Jersey
https://www.edutechlearners.com/download/Graphtheory.pdf

HISTORY – EULER (1976) KONIGSBERG BRIDGE PROBLEM
1847- KIRCHHOFF/CAYLEY DEVELOPED THEORY OF TEES

SOME OF THE SCIENTIFIC PROBLEM WHICH ARE ADDRESSED BY GRAPHS

- To compare and analyse biological network
- To calculate the net output at a point in electronic circuit
- Flux in a Signalling network
- To make matching between applicant and vacancy,
- To locate and open a outlet in a city
- To compare two brains
- Etc. etc.



- These problems can be described and analysed easily in terms of network or graph

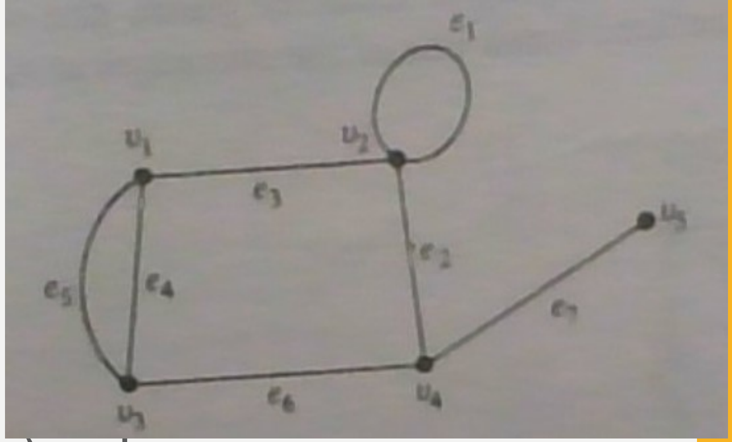
- Terms - Graph vs network: more or less same meaning. Graph term is used in mathematics, while Network term is used in applied sciences

GRAPHS ARE MATHEMATICAL STRUCTURES

Graphs are mathematical structures that represent pairwise relationships between objects. A graph is a flow structure that represents the relationship between various objects. It can be visualized by using the following two basic components:

- Nodes: These are the most important components in any graph. Nodes are entities whose relationships are expressed using edges. If a graph comprises 2 nodes A and B and an undirected edge between them, then it expresses a bi-directional relationship between the nodes and edge.
- Edges: Edges are the components that are used to represent the relationships between various nodes in a graph. An edge between two nodes expresses a one-way or two-way relationship between the nodes.

DEFINITION



- **What is Graph?**

A graph G=(V,E) consists of

- a set of objects V={v1,v2,v3..} called vertices (or nodes, point or junction) and
- another set E={e1,e2,..}, whose elements are called edges (branches, connections)

Such that each edge e_k is identified with an unordered pair (vi,vj) of vertices.

- Vertices are represented by points and edges as a line
- **Self loops-** Such an edge having same vertex as both its end vertices is called a self-loop e.g. e_1
- **Parallel edge -** More than one edge associated with a given pair e.g. e4 and e5 in the top fig.

Note: It is immaterial whether the lines are drawn straight or curved, long or short

Term: Incidence – संयोग, connection between edge and vertices.

e.g. we say that edge e4 is incident on vertex v1

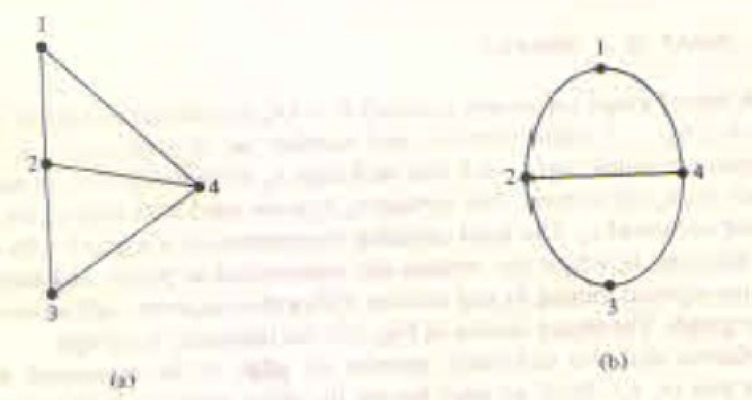
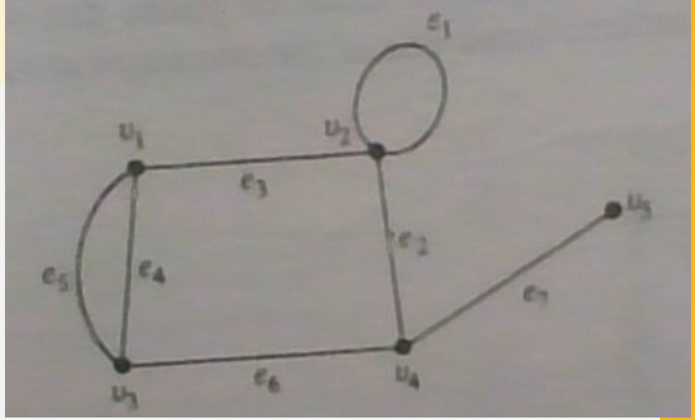


Fig. 1-2 Same graph drawn differently.

IMPORTANT TERMS RELATED TO THE GRAPH DATA STRUCTURE:



- **Vertex:** each node of the graph
- **Edge**: a path or a line or a connection between two vertices
- **Adjacency:** two nodes or vertices are adjacent (Nearby) if they are connected to each other through an edge
- **Path:** a sequence of edges between the two vertices
- **Cycle**: a path where the first and last vertices are the same

- **Degree**: The number of edges incident (connected) on a vertex vi with self loops counted twice is called the degree d(vi) of vertex vi, e.g. d(v1)=3 in the figure
- What is degree of d(v2) and d(v5)?

The **degree sum formula** states that, given a graph $G = (V, E)$,

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

This above relation is referred **as handshaking lemma**

Terms- Lemma - a subsidiary or intermediate theorem in an argument or proof.

PROBLEM-
CALCULATE TOTAL NO. OF EDGES IN THIS GRAPH

Source-Example from unacademy

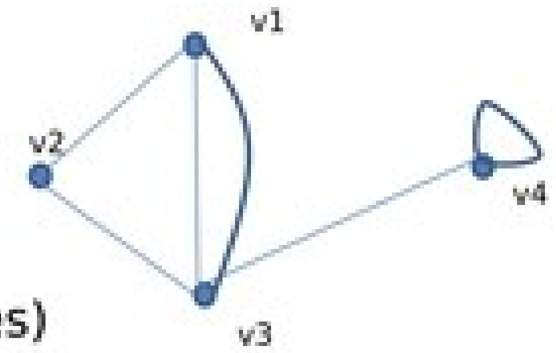
Handshaking Lemma

- **Degree of vertex**

  deg(v)=no. of incident edges + 2*loop edges

  No. of edges = 6

  ∑deg(v) = 3+2+4+3 = 12 = 2*(no. of edges)



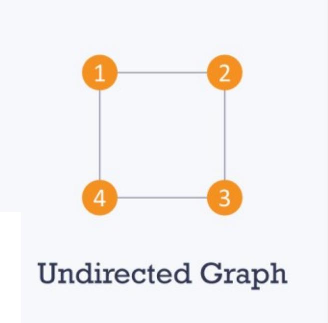
Theorem 1 (Handshaking Lemma):

In any graph G(V,E) the sum of degrees of all the vertices is equal to the twice of no. of edges in that graph.

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

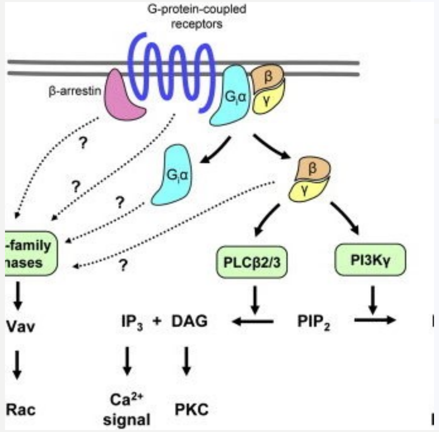
TYPES OF GRAPHS

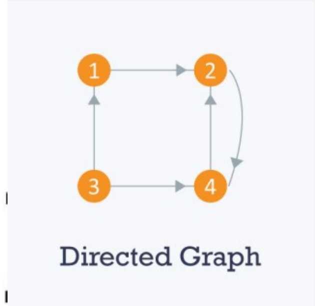
- **Undirected**: An undirected graph is a graph in which all the edges are bi-directional i.e. the edges do not point in any specific direction.



Directed: A directed graph is a graph in which all the edges are uni-directional i.e. the edges point in a single direction.

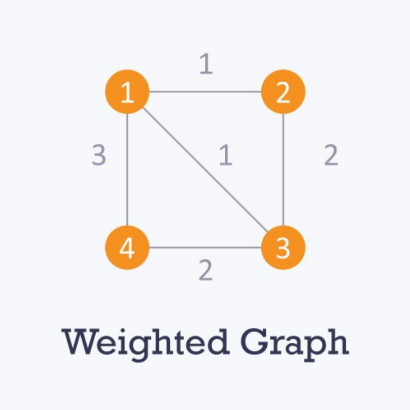
- E.g. network of biological signaling pathways.





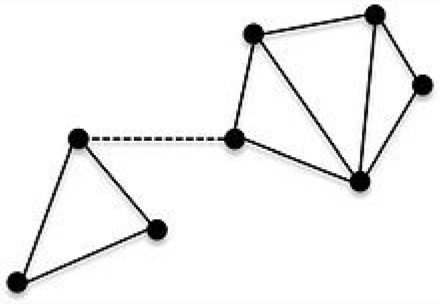
Weighted: In a weighted graph, each edge is assigned a weight or cost (e.g. to represent distance between food outlets, or traffic in road network). Consider a graph of 4 nodes as in the diagram. Each edge has a weight/cost assigned to it. If you want to go from vertex 1 to vertex 3, you can take one of the following 3 paths:

- 1 -> 2 -> 3
- 1 -> 3
- 1 -> 4 -> 3



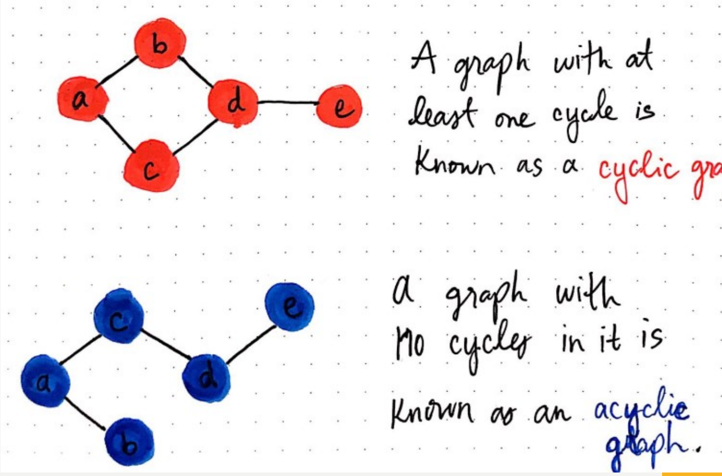
Therefore the total cost of each path will be as follows: - The total cost of 1 -> 2 -> 3 will be (1 + 2) i.e. 3 units - The total cost of 1 -> 3 will be 1 unit - The total cost of 1 -> 4 -> 3 will be (3 + 2) i.e. 5 units

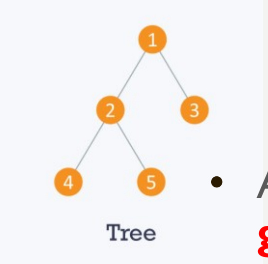
- **Connected and disconnected graphs**
- A graph G is connected if there is at least one path between every pair of vertices in G, Otherwise G is disconnected



- **Cyclic**: A graph is cyclic if the graph comprises a path that starts from a vertex and ends at the same vertex. That path is called a cycle. An acyclic graph is a graph that has no cycle.

- **Finite vs infinite graph** – a graph with finite no. of edges and vertices
- Example – infinite graph – road network in a country.
- Brain network, social media



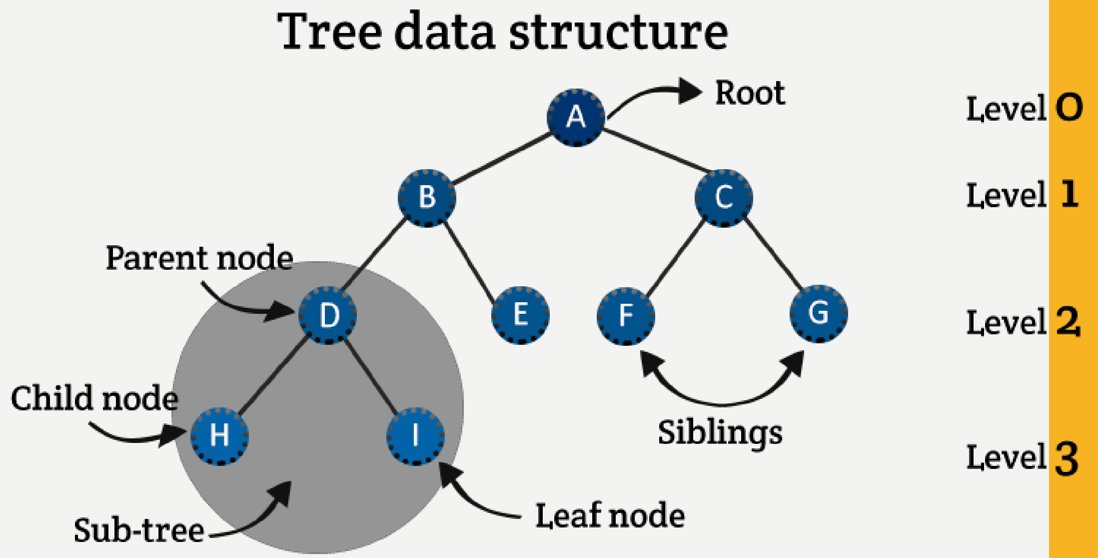


TREE :

Example – Phylogenetic tree, pedigree

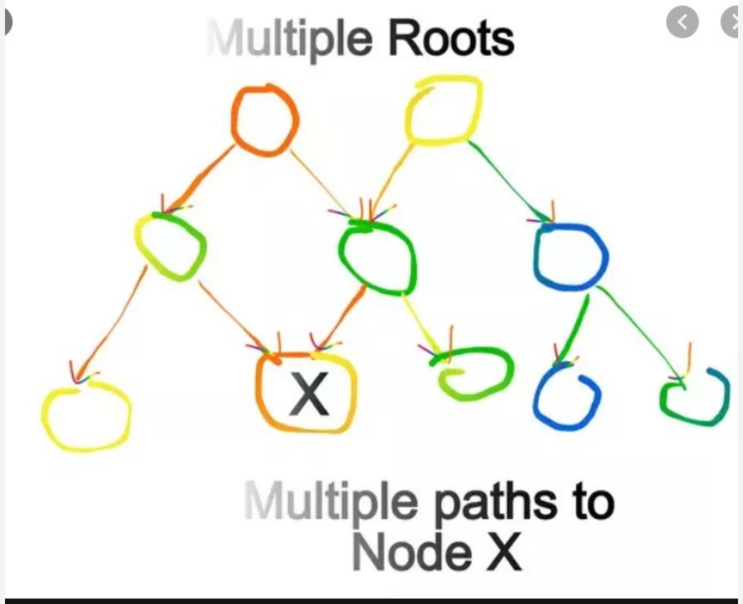
- A **tree** is an undirected graph in which any two vertices are connected by only one path. A tree is an **acyclic graph** and has N - 1 edges where N is the number of vertices. Each node in a graph may have one or multiple parent nodes. However, in a tree, each node (except the root node) comprises exactly one parent node. A root node has no parent.
- A tree cannot contain any cycles or self loops.

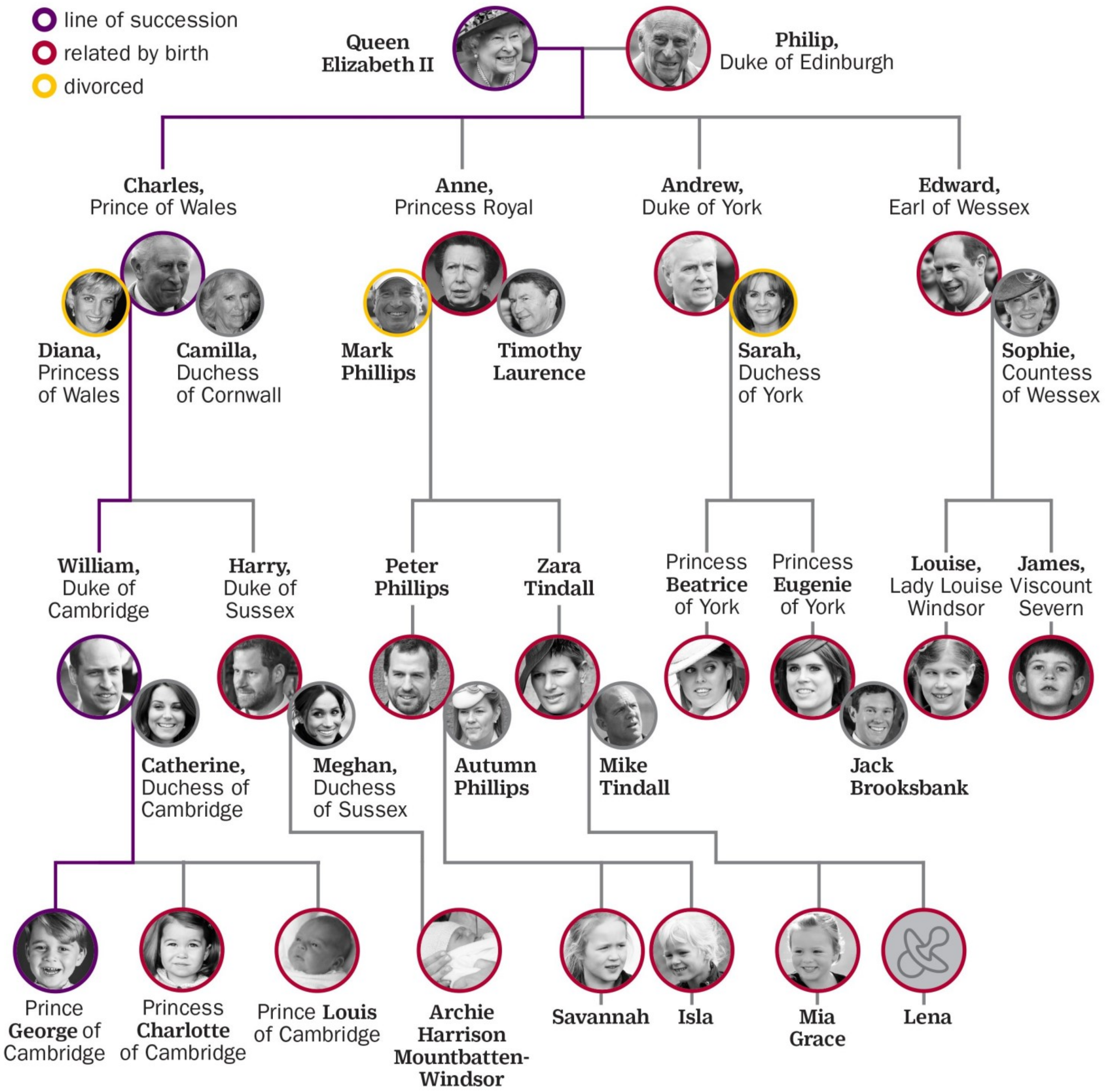
- Tree data structures have terminology that is worth becoming familiar with:
- **Root**: the top (initial) node of the tree, where all the operations start. The root node is the ancestor of all other nodes in
- **Node**: each item in the tree, usually a key-value
- **Edge**: a tree has n-1 edges (where n is the number of nodes) representing the connection between two nodes
- **Parent**: a node which is a predecessor of any node
- **Child**: a node which is descendant of any node
- **Siblings**: a group of nodes which have the same parent
- **Leaf (terminal) node**: a node without children
- **Level(**generation) it is defined as 1 + the number of edges between the node and the root
- **Height**: the number of edges from its root to the furthest leaf
- **Depth**: the number of edges from the node to the tree's root
- **Sub-tree**: a portion of a tree data structure that can be viewed as a complete tree in itself
- There are different types of trees, like Binary Tree, Binary Search Tree, Red-Black tree, AVL tree, Heap, etc.



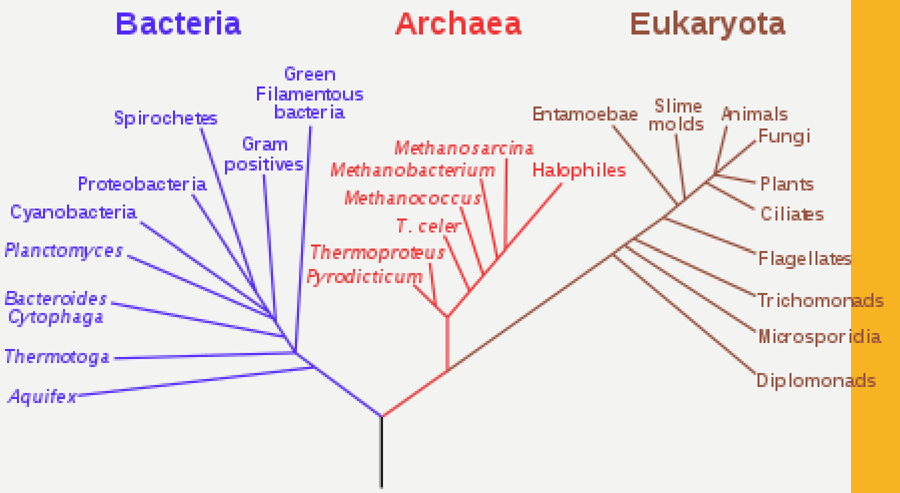
GENERAL TREE EXAMPLE- PEDIGREE OF BRITISH QUEEN ELIZABETH

- Example - where each node in a graph may have multiple parent nodes. In real-life situation this happens in Chemical industries, pipeline, Sewage network



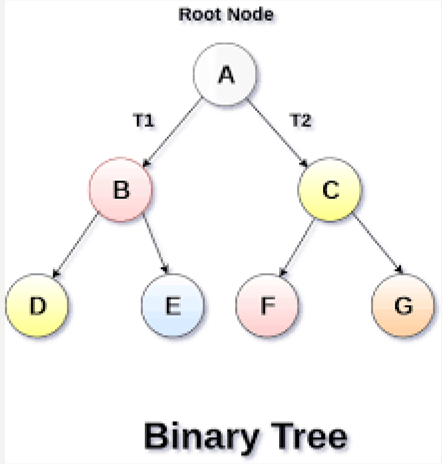


BINARY VS ROOTED TREE



- a **rooted tree** is a tree wherein one node is designated as root and every edge is directed away from it

- A **binary tree is** a tree in which there is exactly (ONLY) one vertex of degree two
- a **binary tree** is a tree data structure in which each node has at most two children,



- Spanning tree

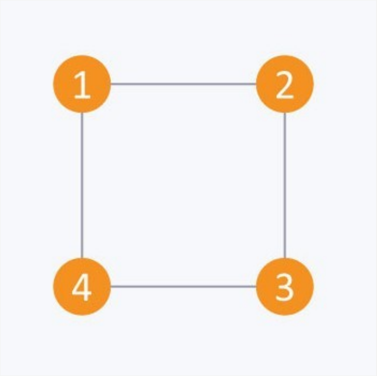
PROPERTIES OF TREES

- Tree is a connected graph AND tree is an acyclic graph
- There is one and only one path between every pair of vertices in a tree .
- A tree with n-vertices has n-1 edges or in reverse
- Any connected graph with n vertices and n-1 edges is a tree
- A graph is tree if and only if it is minimally connected
- **Distance-**In a connected graph, G the distance d(vi,vj) between two of its vertices vi and vj is the length of the shortest path between them (i.e. no of edges in the shortest path)

GRAPH REPRESENTATION

- **Adjacency matrix**
- An adjacency matrix is a **m x n** binary matrix **A**.
- Its elements - Ai,j is **1** if there is an edge from vertex i to vertex j else Ai,j is 0.
- The adjacency matrix can also be modified for the weighted graph in which instead of storing 0 or 1 in Ai,j, the weight or cost of the edge will be stored.
- In an undirected graph, if Ai,j = 1, then Aj,i = 1.
- In a directed graph, if Ai,j = 1, then Aj,i may or may not be 1.

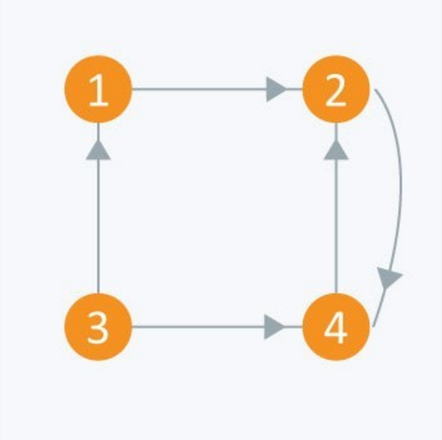
Undirected graph



Adjacency matrix

i/j	:	1	2	3	4
1	:	0	1	0	1
2	:	1	0	1	0
3	:	0	1	0	1
4	:	1	0	1	0

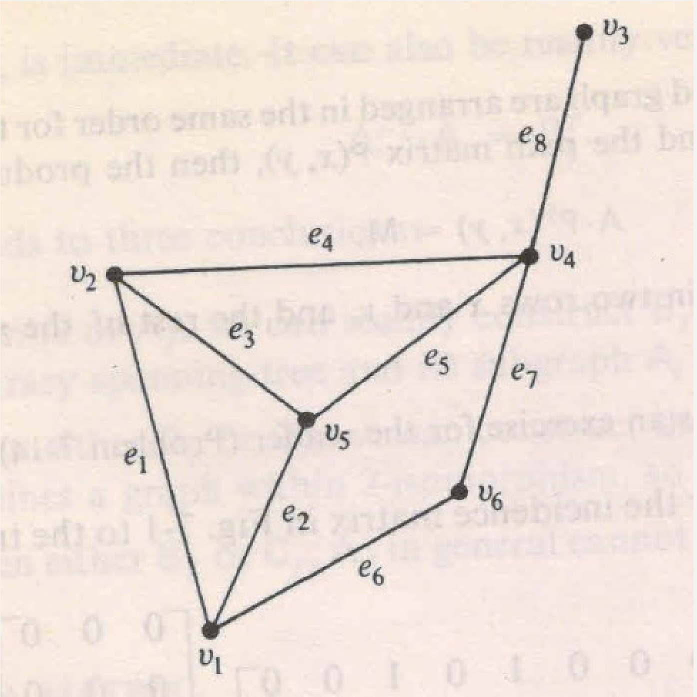
directed graph



Adjacency matrix

i/j:		1	2	3	4
1	:	0	1	0	0
2	:	0	0	0	1
3	:	1	0	0	1
4	:	0	1	0	0

- Q.Write down the adjacency matrix of the opposite graph



- Ans:

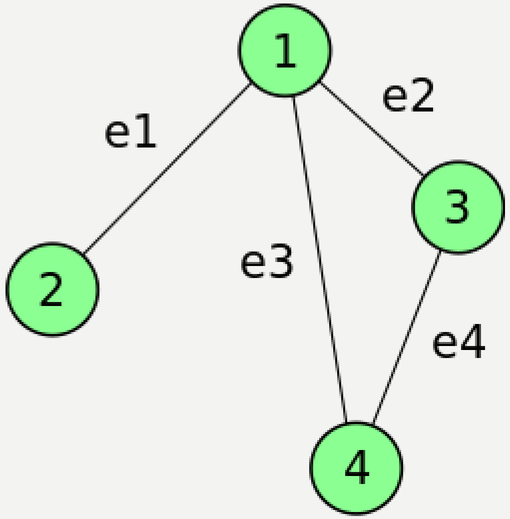
$$
X = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array}
\begin{array}{c} \begin{array}{cccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0
\end{bmatrix} \end{array}
$$

INCIDENCE (CONMECTION) MATRIX

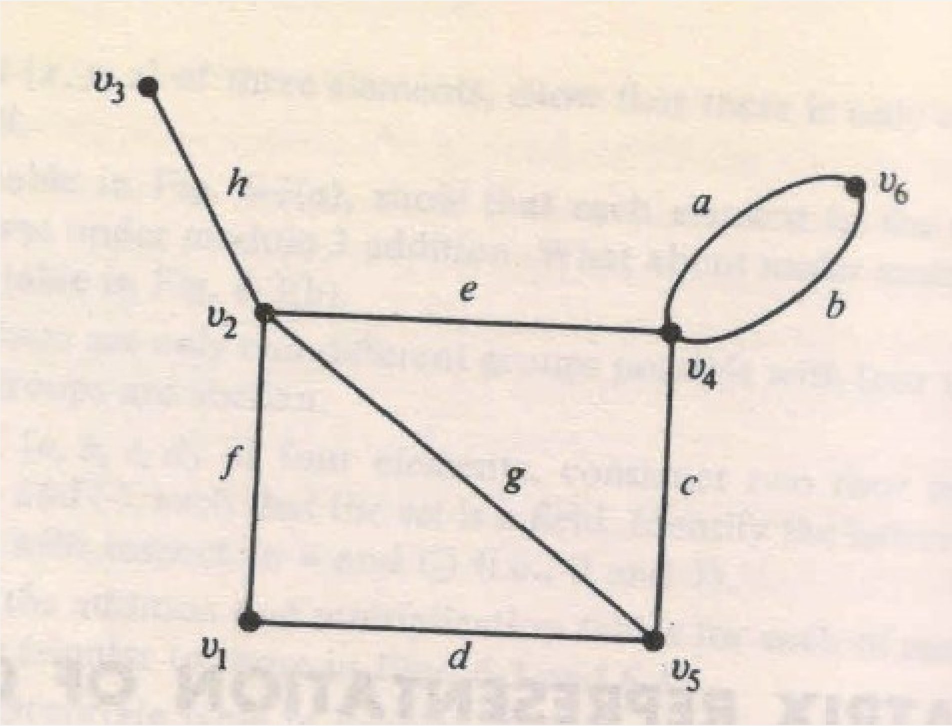
- Let there be a matrix A=[aij]
- Which has n rows (corresponding to n-vertices) and e-columns (corresponding to e-edges)
- The matrix element
- aij=1 if jth edge ej is incident on the ith vertex vi
- =0 otherwise

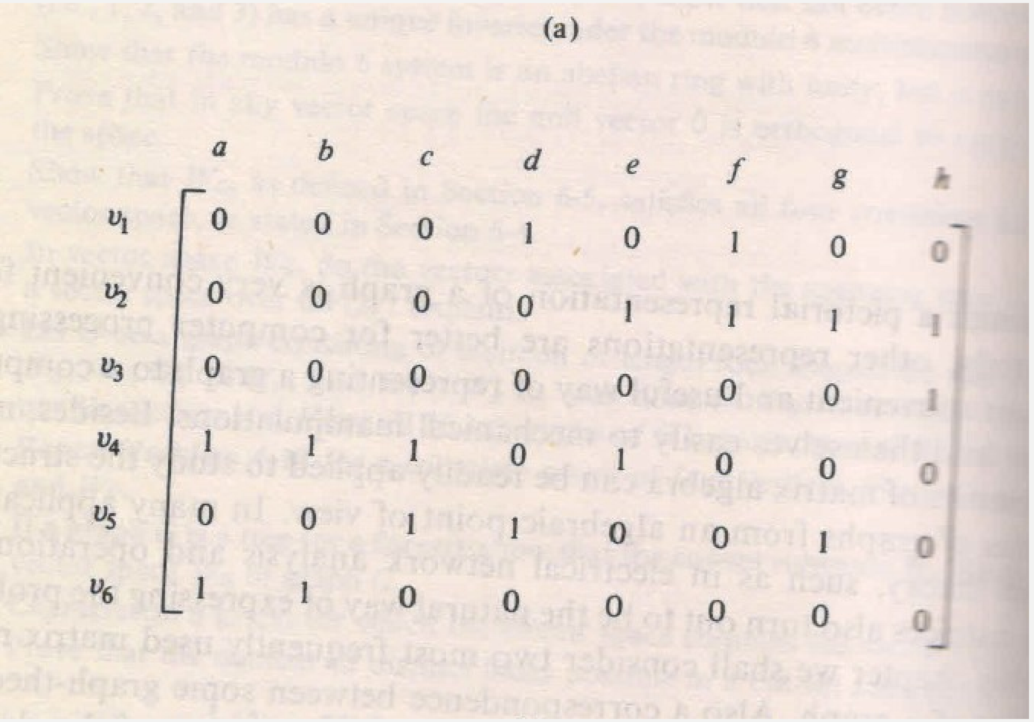
	e_1	e_2	e_3	e_4
1	1	1	1	0
2	1	0	0	0
3	0	1	0	1
4	0	0	1	1

$$= \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



WHAT IS THE INCIDENCE MATRIX OF THE FOLLOWING GRAPH?





(a)

	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

SUBGRAPH

- A graph g is said to be a subgraph of a graph G if all the vertices and all the edges of g are in G.
- And each edge of g has the same end vertices in g as in G

- Theorem: Every graph is its own subgraph

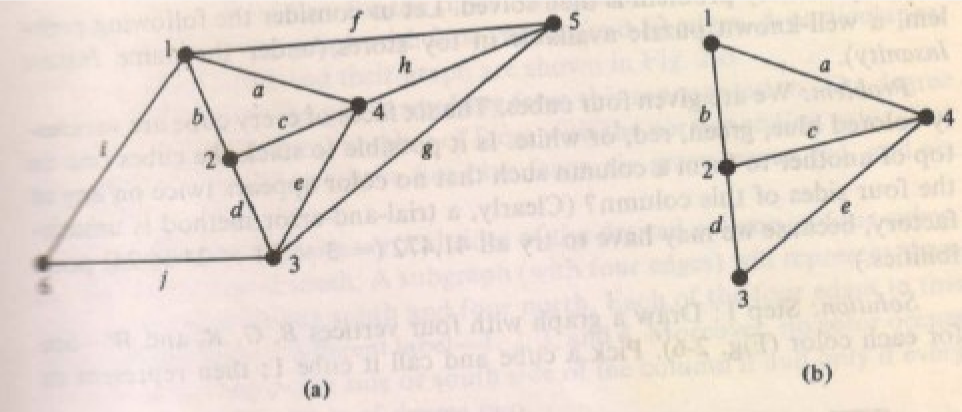


Fig. 2-5 Graph (a) and one of its subgraphs (b).