Graph:

Graph Theory

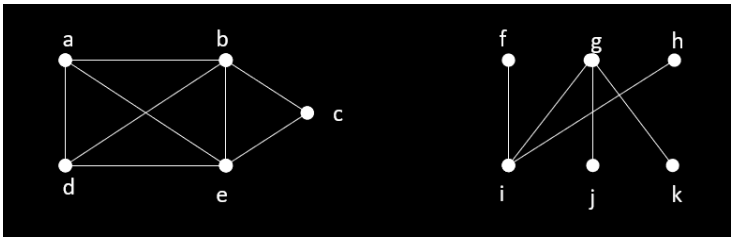
- branch of mathematics concerned with network of points connected by lines.

Graph consists of the following:

•finite set of vertices (nodes)
•finite unordered pairs of vertices called edges (edge family)
•vertex set V(G)
•edge set E(G)
•order – number of vertices
•size – number of edges
•adjacent (neighbors)
•incident
•multiple edges (parallel edge) – A pair of vertices consists of more than one edge.
•Loop – An edge whose endpoints are the same vertex
•loop-free – graph with no loop
•degree of a vertex – is the number of edges with the vertex
•degree sequence – graph consists of the degrees written in increasing orders.
•Isolated – does not belong to any edge and has a degree of 0.
•end-vertex (pendant) – is a vertex with

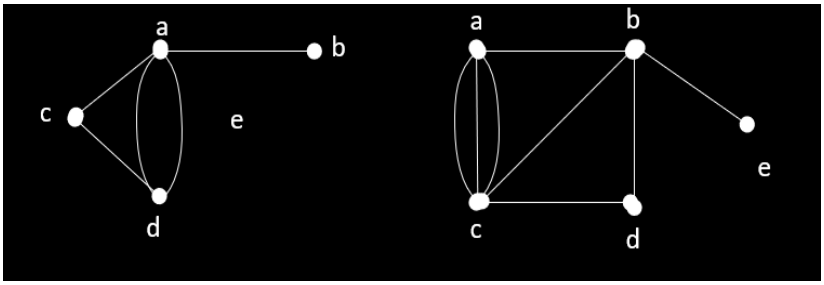
Simple Graph

• consists of non empty finite set of elements and finite of distinct elements.
• it is a graph with no loops and no multiple edges.



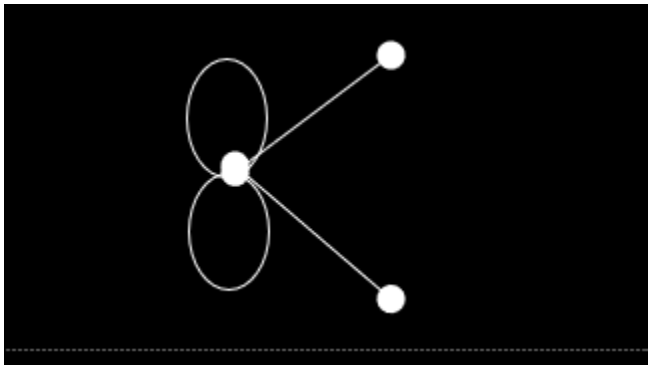
Multigraph

•A graph in which multiple edges may connect to the same pair of vertices.



Pseudograph

- Is a graph with at least one loop.



Trivial Graph

- A graph with one vertex and no edge.

Null Graph

- Is a graph whose edge - set is empty.

Empty Graph

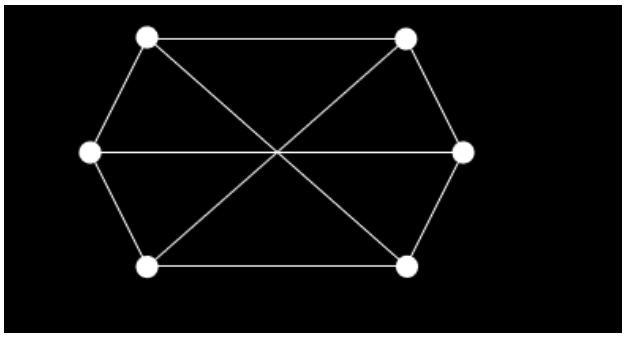
- Has no vertices and no edges.

Regular Graph

- A graph in which each vertex has the degree.

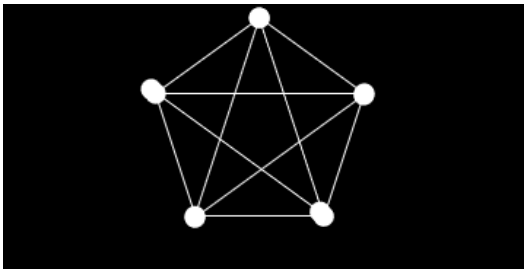
Subgraph

•Is a whose vertices and edges are subset of another graph



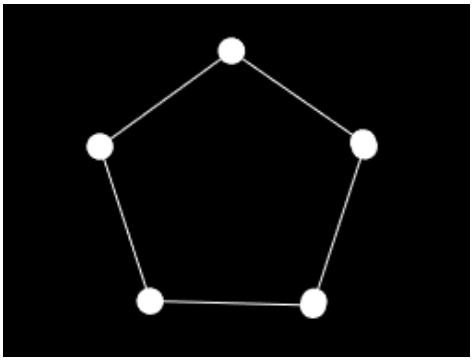
Complete Graph

- Is a simple graph in which each pair of distinct vertices are adjacent (there is exactly one edge joining every pair of vertices.



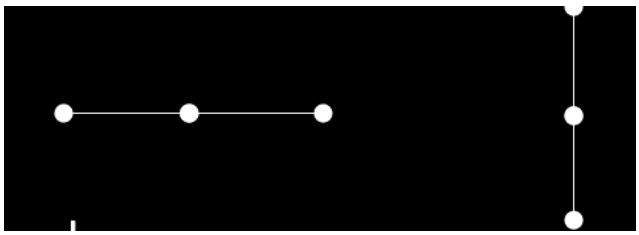
Cycle Graph

- Is a connected graph that is regular of degree of 2.



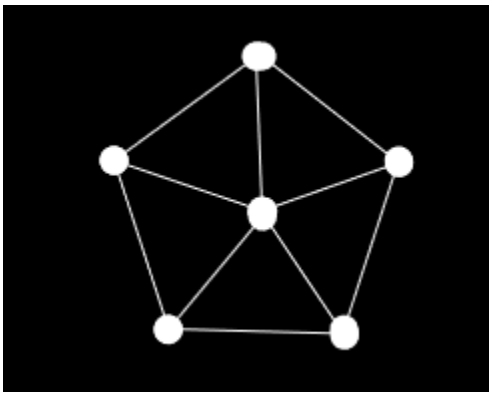
Path Graph

- Is a tree with two nodes of vertex degree 1 and the other nodes of vertex degree
- A graph that can be drawn so that all of its vertices and edges are lie in on a single straight line.



Wheel Graph

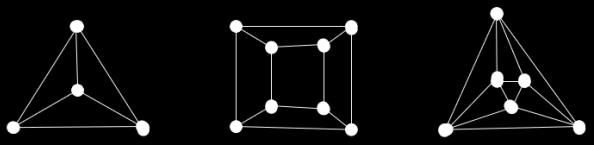
- Is a graph formed by connecting a single universal vertex to all vertices of a cycle.



Platonic Graph

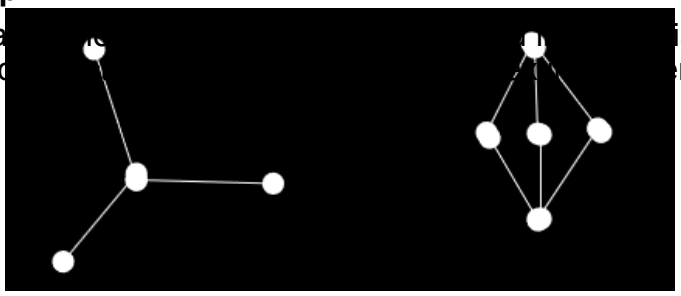
- Are graphs which formed from the vertices and edges of the five regular (platonic) solid.
- Tetrahedron, octahedron, cube, icosahedron and dodecahedron

Solid	Faces / Regions	Vertices	Edges
Tetrahedron	4	4	6
Octahedron	8	6	12
Cube (Hexahedron)	6	8	12
Icosahedron	20	12	30
Dodecahedron	12	20	30



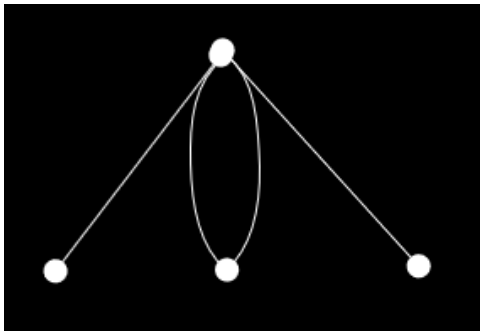
Bipartite Graph

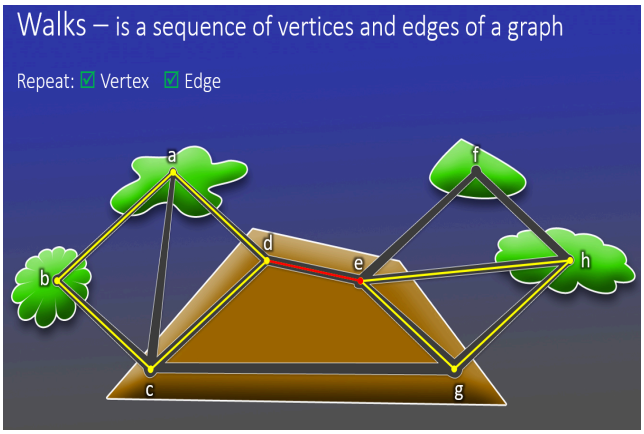
- Is a gra isjoint sets such as all edges c r set.

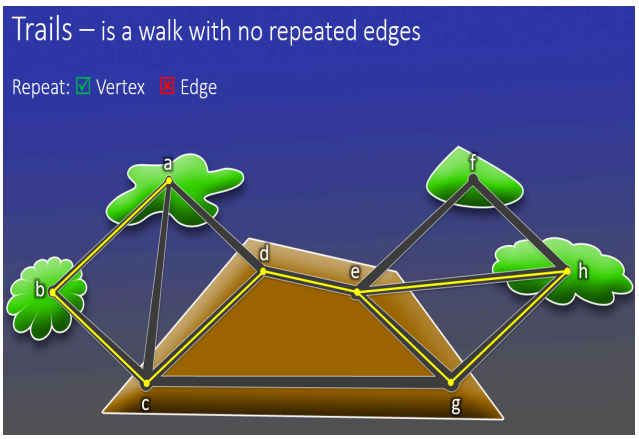


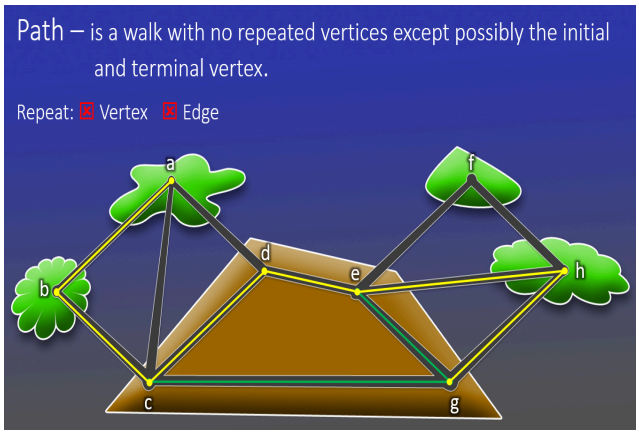
Complete Bipartite Graph

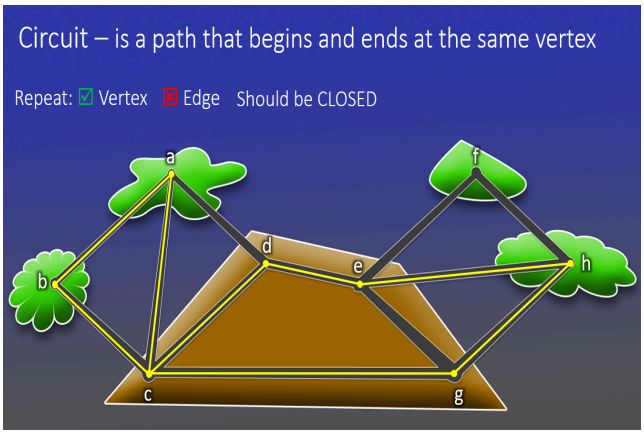
- is a special type of bipartite graph where every vertex of one set is connected to every vertex of other set.

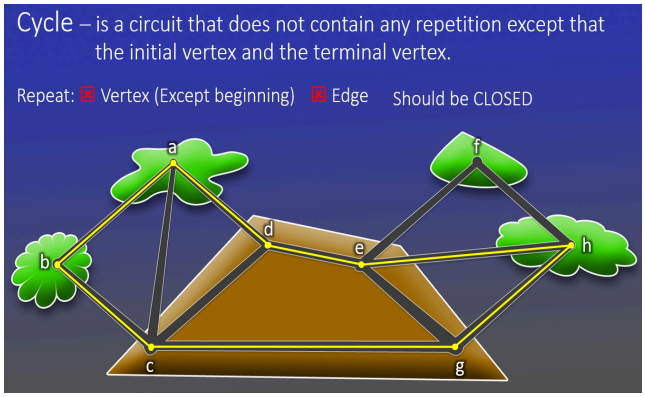


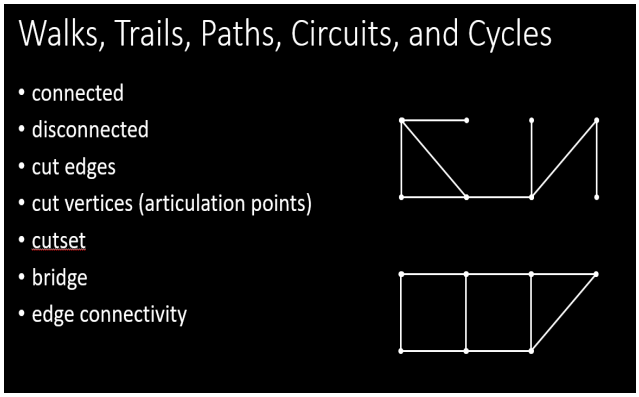




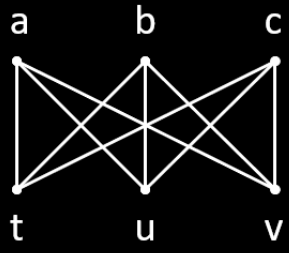


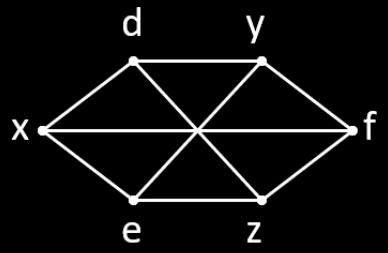






Isomorphism



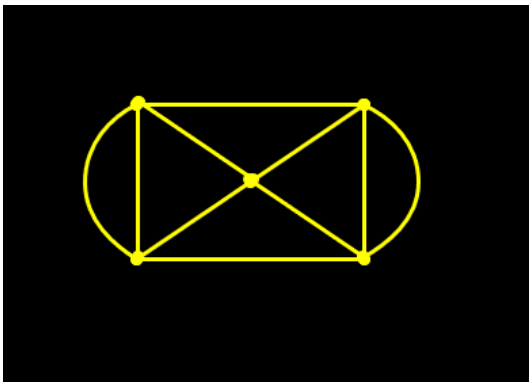


	a	b	c	t	u	v
a	0	0	0	1	1	1
b	0	0	0	1	1	1
c	0	0	0	1	1	1
t	1	1	1	0	0	0
u	1	1	1	0	0	0
v	1	1	1	0	0	0

	d	e	f	x	y	z
d	0	0	0	1	1	1
e	0	0	0	1	1	1
f	0	0	0	1	1	1
x	1	1	1	0	0	0
y	1	1	1	0	0	0
z	1	1	1	0	0	0

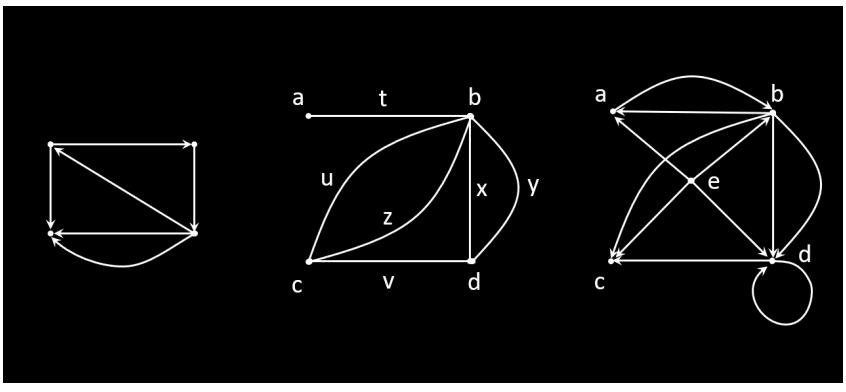
Eulerian

- Is a graph in which the vertices represent points in the plane, and each edge is assigned the length equal to the Euclidean distance between its endpoints.



Hamiltonian

- is a graph which has a closed path (cycle) that visits each vertex exactly once, ending on the same vertex as it began. This closed path is also called a Hamiltonian cycle.

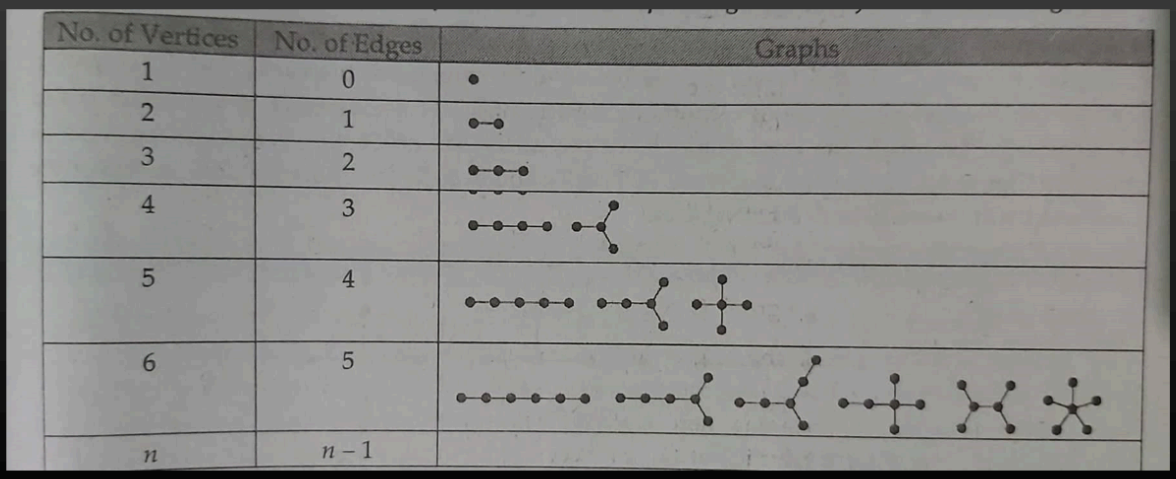


TREES:

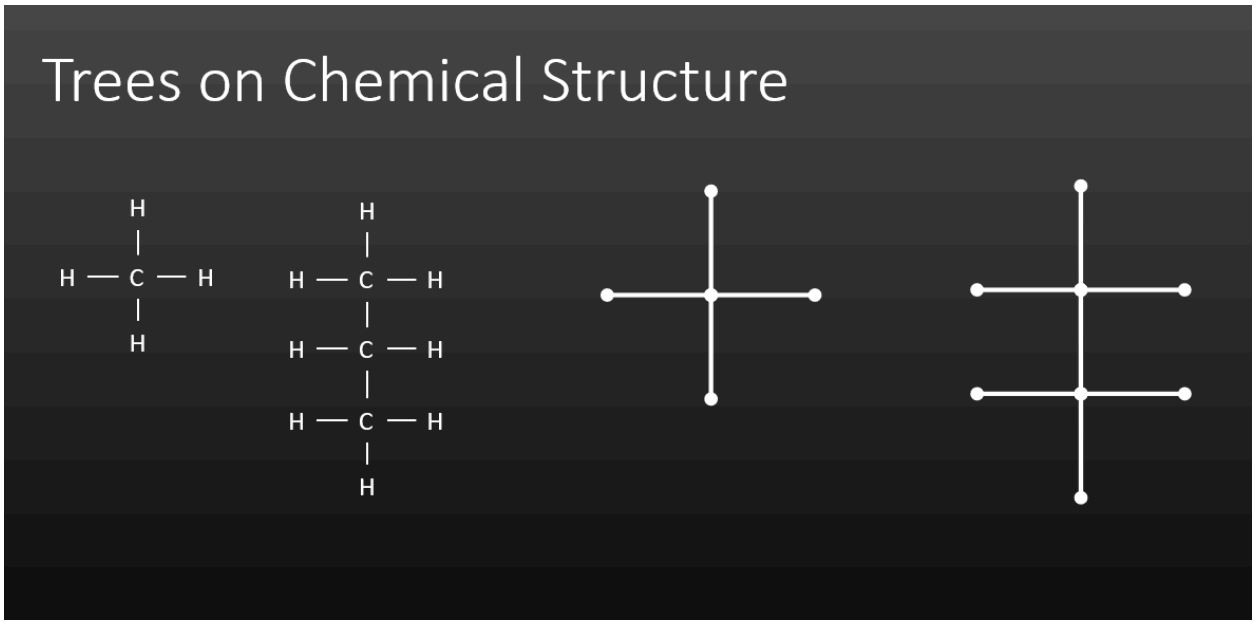
Trees

- Is a mathematical structure that is a special kind of graph which has the following properties:
    1. undirected
    2. Connected
    3. acyclic
    4. bridge
    5. Components
    6. Forest

Examples of Trees with Corresponding Number of Vertices and Edges



No. of Vertices	No. of Edges	Graphs
1	0	
2	1	
3	2	
4	3	
5	4	
6	5	
n	$n-1$	



Rooted Tree

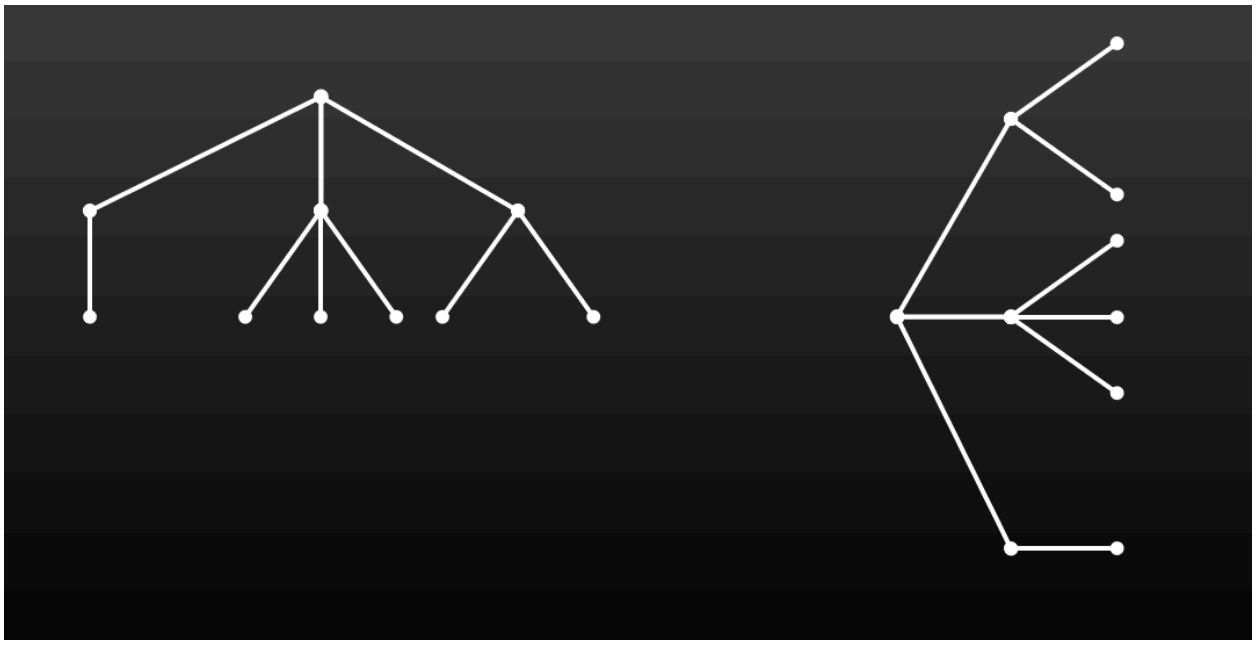
- A tree is called a rooted tree if one vertex has been designated the root.

ROOT

- usually refers to a top – most vertex that extends downward meaning each edge is simply directed away from the root.

FREE ROOT

- a tree without any designated root.



Level (or depth)

- in a rooted tree, the vertex is its distance from the root.

Height

- height of a tree is the number of vertices or edges traversed from the root to get the most distant vertex.

Parent

- is the vertex connected to it on the path to the root.

Child (or offspring)

- vertex *d* is a vertex of which *a* is a parent.

Sibling

- is a vertex which shares the same parent.

Descendant

- is a vertex which is further away from the root than some other vertex.

Ancestors

- is any vertex between a given vertex and the root.

Terminal vertex (or leaf)

- is a vertex with no children.

Internal vertex (or branch vertex)

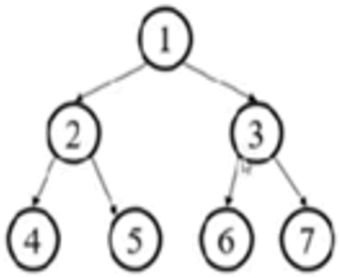
- is a vertex that has at least one child.

Sub – tree

- a smaller portion of a tree starting at some specifies vertex.

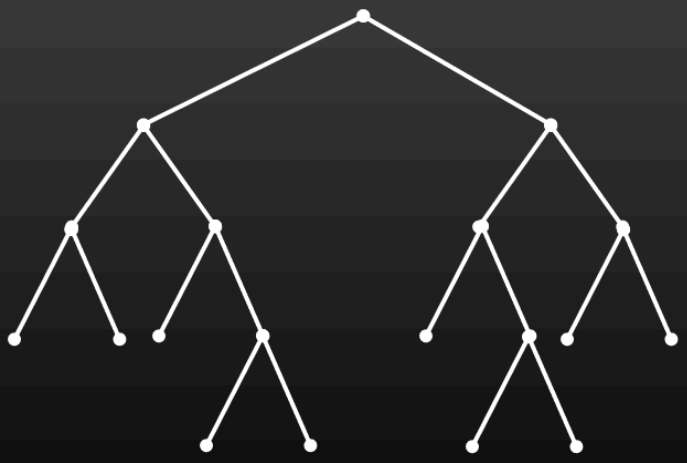
BASIC TERMINOLOGY: BINARY TREE

Root : it is the starting node of the tree
Parent: it is the above node of the present node
Sibling: it is the other node but with the same parent
Children: it is the node below the present node
Ancestor: it is the list of node from parent to the root
Descendant: it is the list of node from the present node to children up to the leaves
Internal Vertices: is a vertex that has parent and at least one child
External Vertices: list of nodes without parent or nodes without children
List of Leaves: list of nodes without children
Height of tree: the length of the longest path from a leaf to the root
Dept of tree: is the length of the path from it to the root
Size of tree: total number of nodes

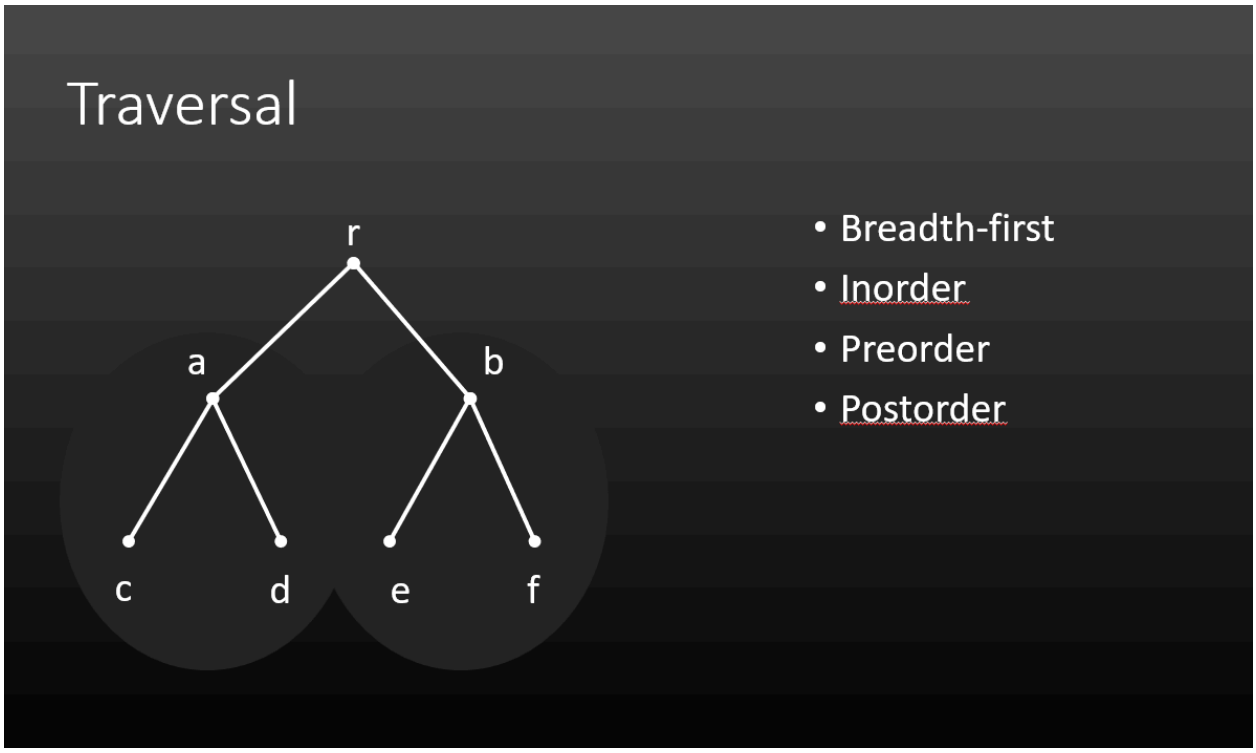


Root : {1}
Parent of 7: {3}
Sibling of 5 :{4}
Children of 2 : {4,5}
Ancestor of 6 : {3,1}
Descendant of 1: {2,3,4,5,6,7}
Internal Vertices: {2,3}
External Vertices: {1,4,5,6,7}
List of Leaves: {4,5,6,7}
Height of tree: 3
Dept of tree: 2
Size of tree: 7

M-ary



- Complete
- Full
- Balanced



Spanning Tree

- is a subgraph which is tree and contains all vertices of the graph.

Prim's algorithm

- developed by Robert Clay in 1957 which is a greedy algorithm that determines a minimum spanning tree for a connected weighted undirected graph.

Kruskal's algorithm

- developed by Joseph Bernard Kruskal (1928-2010).

Dijkstra's algorithm

- developed by Edsger W. Dijkstra (1930 – 2002). The algorithm is a graph search algorithm that determines the single – source shortest – path problem for a graph producing a shortest path tree.