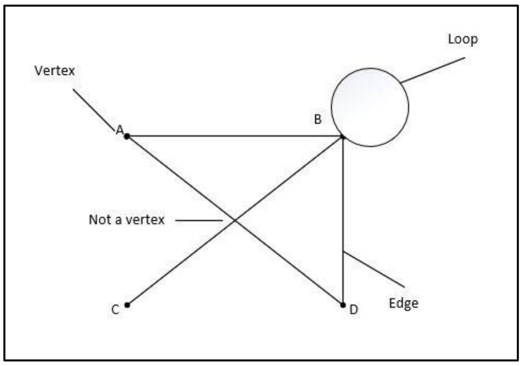
GRAPH TERMINOLOGY



SUMMARY OF GRAPH TERMINOLOGY IN GRAPH THEORY

The **size** of a graph is the number of its edges.

The **degree of a vertex** written $deg(v)$ is equal to the ***number of edges*** which are incident on v

The **sum of the degrees** of the vertices of a graph is equal to ***twice the number of edges***

The vertex v is said to be **even or odd (parity)** according as $deg(v)$ is even or odd

A vertex v is **isolated** if it is does not belong to any edge

A vertex with degree 1 is called a **leaf vertex**

The incident edge of vertex with degree 1 is referred as a **pendant edge**

A **path** is the sequence of connected vertices.

A **simple path** is a path where the vertices are only passed through once

A **trail** is a path where each edge is traveled once, meaning that there are no repeated edges (all edges are distinct)

The **length** of the path is the number n of edges that it contains

The **distance** between two vertices is described by the length of the shortest path that joins them

A **cycle / simple cycle** is a closed path with at least 3 edges, and no repeated vertices and

An **acyclic** is a graph that has no cycles in it

A **closed path or circuit** is a path that starts and ends at the same vertex

A graph is called **planar** if it can be drawn in the plane without any edges crossing each other

EULER PATH, EULER CIRCUIT, HAMILTON PATH, HAMILTON CIRCUIT

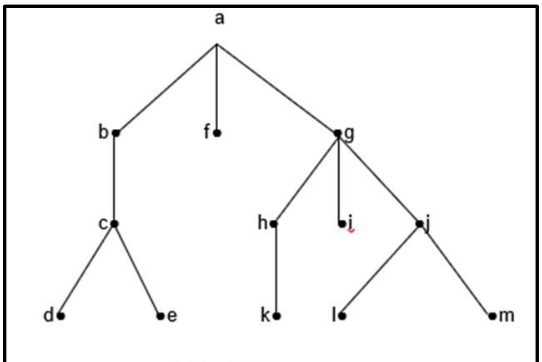
Euler path -A connected multigraph has an **Euler path** if and only if it has **exactly two vertices of odd degree**.

Euler Circuit- A connected multigraph has an **Euler circuit** if and only if **every vertex have even degree.**

Hamilton path-A **Hamilton path** is a simple path in a graph G that **passes through every vertex exactly once.**

Hamilton Circuit-A **Hamilton circuit** is a simple circuit in a graph G that **passes through every vertex exactly once.**

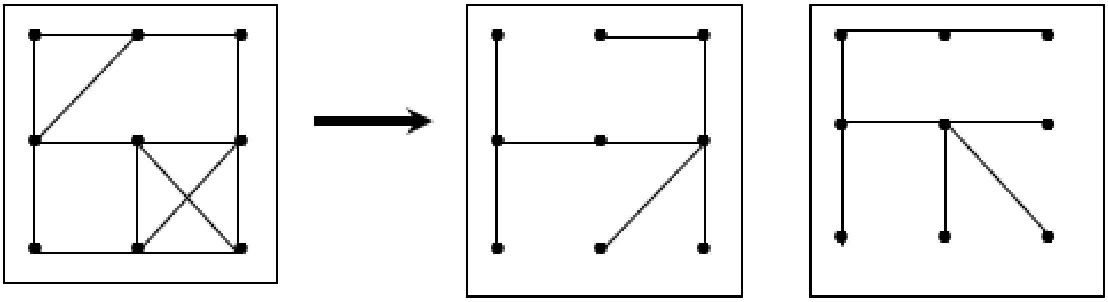
EXAMPLE OF TREES



A Rooted Tree

- The **root** is **a**.
- The **parent of h,i** and **j** is **g**.
- The **children of b** is **c.** The **children of j** are **l** and **m.**
- **h, i** and **j** are a **sibling**.
- The **ancestor of e** are **c, b** and **a**.
- The **descendants** of **b** are **c, d** and **e.**
- The **internal vertices** are **a, b, c, g, h** and **j.** (vertices that have children)
- The **leaves** are **d, e, f, i, k, l** and **m.** (vertices that have no children)

SPANNING TREES



MINIMAL SPANNING TREES

PRIM'S ALGORITHM

KRUSKAL ALGORITHM