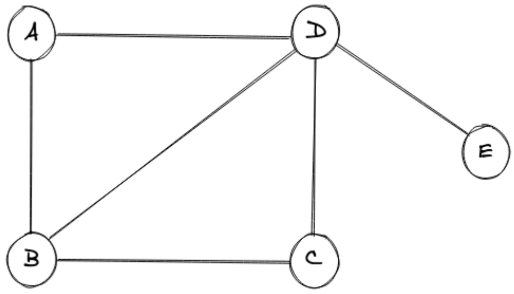
Introduction to Graphs

A **graph is an advanced data structure** that is used to organize items in an **interconnected network**. Each item in a graph is known as a **node**(or **vertex**) and these nodes are connected by **edges**.

In the figure below, we have a simple graph where there are five nodes in total and six edges.

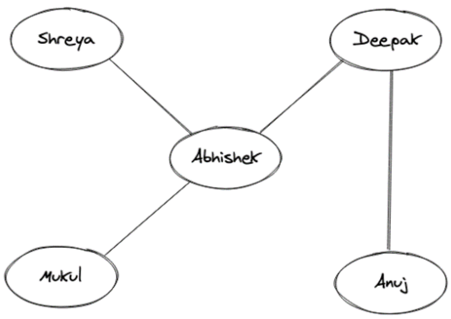


The nodes in any graph can be referred to as **entities** and the edges that connect different nodes define the **relationships between these entities**. In the above graph we have a set of nodes **{V} = {A, B, C, D, E}** and a set of edges, **{E} = {A-B, A-D, B-C, B-D, C-D, D-E}** respectively.

Real-World Example

A very good example of graphs is a **network of socially connected people**, connected by a simple connection which is whether they know each other or not.

Consider the figure below, where a pictorial representation of a social network is shown, in which there are five people in total.

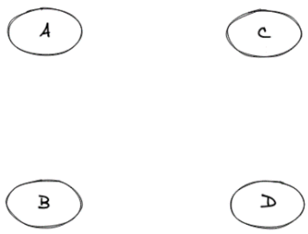


A line in the above representation between two people mean that they know each other. If there's no line in between the names, then they simply don't know each other. The names here are equivalent to the nodes of a graph and the lines that define the relationship of "knowing each other" is simply the equivalent of an edge of a graph. It should also be noted that the relationship of knowing each other goes both ways like "Abhishek" knows "Mukul" and "Mukul" knows "Abhishek".

Types of Graphs

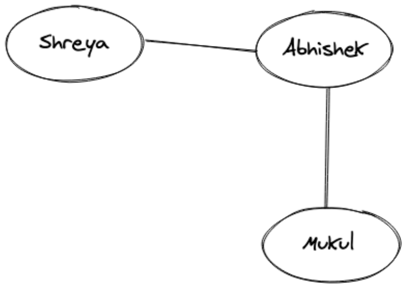
1. Null Graphs

A graph is said to be null if there are no edges in that graph.



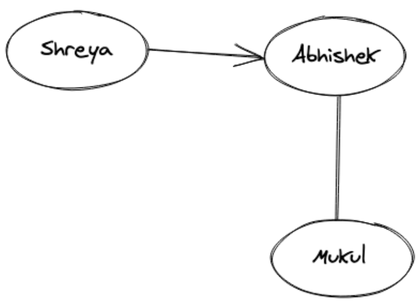
2. Undirected Graphs

If we take a look at the pictorial representation that we had in the Real-world example above, we can clearly see that different nodes are connected by a link (i.e. edge) and that edge doesn't have any kind of direction associated with it. For example, the edge between "Anuj" and "Deepak" is bi-directional and hence the relationship between them is two ways, which turns out to be that "Anuj" knows "Deepak" and "Deepak" also knows about "Anuj". These types of graphs where the relation is bi-directional or there is not a single direction, are known as Undirected graphs.



3. Directed Graphs

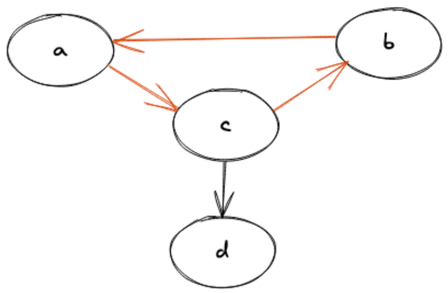
What if the relation between the two people is something like, "Shreya" know "Abhishek" but "Abhishek" doesn't know "Shreya". This type of relationship is one-way, and it does include a direction. The edges with arrows basically denote the direction of the relationship and such graphs are known as directed graphs.



It can also be noted that the edge from "Shreya" to "Abhishek" is an outgoing edge for "Shreya" and an incoming edge for "Abhishek".

4. Cyclic Graph

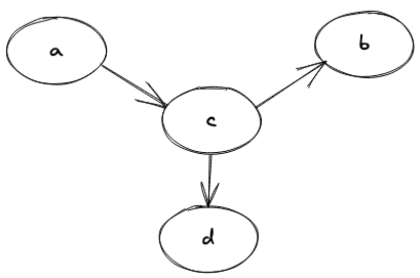
A graph that contains at least one node that traverses back to itself is known as a cyclic graph. In simple words, a graph should have at least one cycle formation for it to be called a cyclic graph.



It can be easily seen that there exists a cycle between the nodes (a, b, c) and hence it is a cyclic graph.

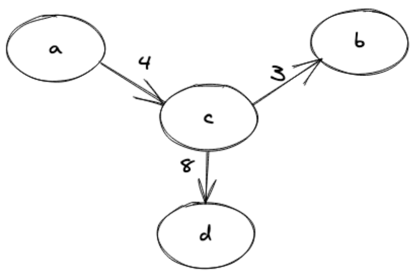
5. Acyclic Graph

A graph where there's no way we can start from one node and can traverse back to the same one, or simply doesn't have a single cycle is known as an acyclic graph.



6. Weighted Graph

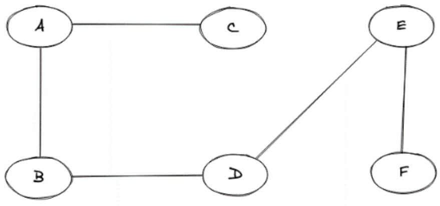
When the edge in a graph has some weight associated with it, we call that graph as a weighted graph. The weight is generally a number that could mean anything, totally dependent on the relationship between the nodes of that graph.



It can also be noted that if any graph doesn't have any weight associated with it, we simply call it an unweighted graph.

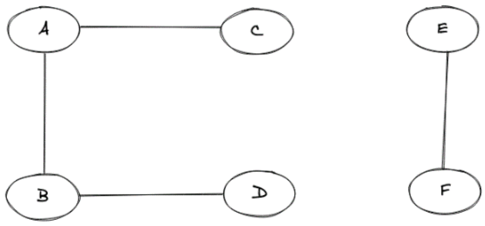
7. Connected Graph

A graph where we have a path between every two nodes of the graph is known as a connected graph. A path here means that we are able to traverse from a node "A" to say any node "B". In simple terms, we can say that if we start from one node of the graph we will always be able to traverse to all the other nodes of the graph from that node, hence the connectivity.



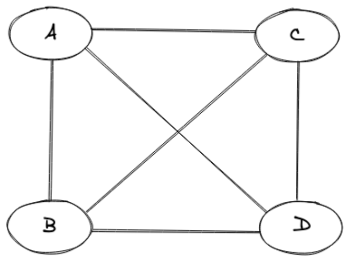
8. Disconnected Graph

A graph that is not connected is simply known as a disconnected graph. In a disconnected graph, we will not be able to find a path from between every two nodes of the graph.



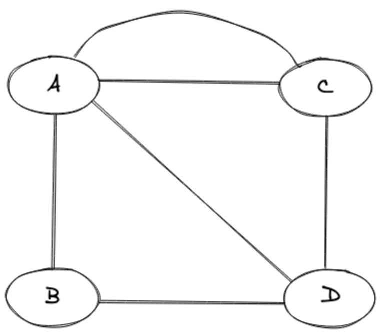
9. Complete Graph

A graph is said to be a complete graph if there exists an edge for every pair of vertices(nodes) of that graph.



10. Multigraph

A graph is said to be a multigraph if there exist two or more than two edges between any pair of nodes in the graph.



Commonly Used Graph Terminologies

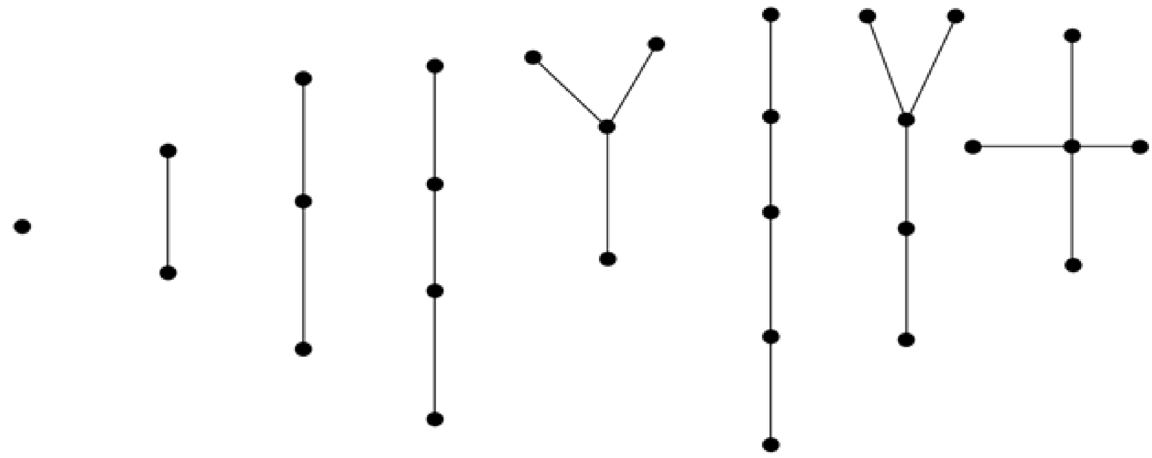
- Path - A sequence of alternating nodes and edges such that each of the successive nodes are connected by the edge.
- Cycle - A path where the starting and the ending node is the same.
- Simple Path - A path where we do not encounter a vertex again.
- Bridge - An edge whose removal will simply make the graph disconnected.
- Forest - a **forest** is **an undirected, disconnected, acyclic graph**.
- Tree - a **tree** is an **undirected, connected and acyclic graph.**
- Degree - The degree in a graph is the number of edges that are incident on a particular node.
- Neighbour - We say vertex "A" and "B" are neighbours if there exists an edge between them.

What is Tree and Forest?

Tree

- In graph theory, a tree is an undirected, connected and acyclic graph. In other words, a connected graph that does not contain even a single cycle is called a tree.
- A tree represents hierarchical structure in a graphical form.
- The elements of trees are called their nodes and the edges of the tree are called branches.
- A tree with n vertices has (n-1) edges.
- Trees provide many useful applications as simple as a family tree to as complex as trees in data structures of computer science.
- A leaf in a tree is a vertex of degree 1 or any vertex having no children is called a leaf.

Example

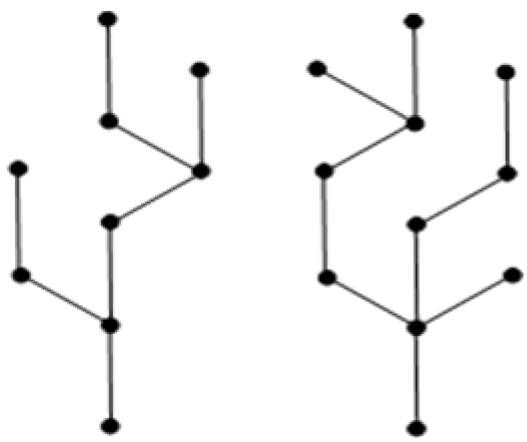


In the above example, all are trees with fewer than 6 vertices.

Forest

In graph theory, a **forest** is **an undirected, disconnected, acyclic graph**. In other words, a disjoint collection of trees is known as forest. Each component of a forest is tree.

Example



The above graph looks like a two sub-graphs but it is a single disconnected graph. There are no cycles in the above graph. Therefore it is a forest.