DATA STRUCTURE - GRAPH DATA STRUCTURE

A graph is a pictorial representation of a set of objects where some pairs of objects are connected by links. The interconnected objects are represented by points termed as **vertices,** and the links that connect the vertices are called **edges**.

Formally, a graph is a pair of sets $V, E,$ where **V** is the set of vertices and **E** is the set of edges, connecting the pairs of vertices. Take a look at the following graph −
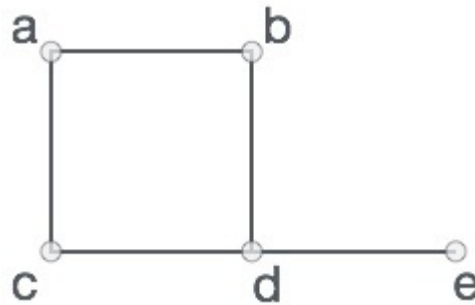

In the above graph,

V = {a, b, c, d, e}

E = {ab, ac, bd, cd, de}

Graph Data Structure

Mathematical graphs can be represented in data-structure. We can represent a graph using an array of vertices and a two dimensional array of edges. Before we proceed further, let's familiarize ourselves with some important terms −

- **Vertex** − Each node of the graph is represented as a vertex. In example given below, labeled circle represents vertices. So A to G are vertices. We can represent them using an array as shown in image below. Here A can be identified by index 0. B can be identified using index 1 and so on.

- **Edge** − Edge represents a path between two vertices or a line between two vertices. In example given below, lines from A to B, B to C and so on represents edges. We can use a two dimensional array to represent array as shown in image below. Here AB can be represented as 1 at row 0, column 1, BC as 1 at row 1, column 2 and so on, keeping other combinations as 0.

- **Adjacency** − Two node or vertices are adjacent if they are connected to each other through an edge. In example given below, B is adjacent to A, C is adjacent to B and so on.

- **Path** − Path represents a sequence of edges between two vertices. In example given below, ABCD represents a path from A to D.
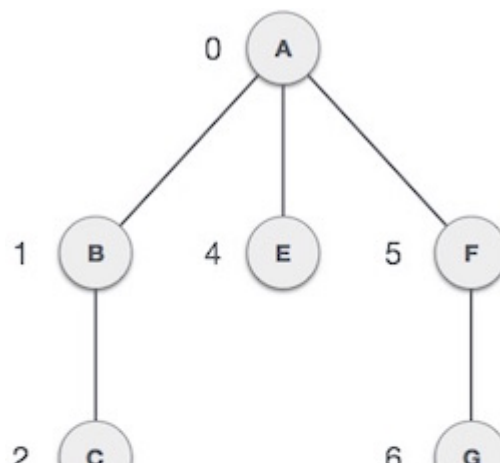
Basic Operations

Following are basic primary operations of a Graph which are following.

- **Add Vertex** − add a vertex to a graph.

- **Add Edge** − add an edge between two vertices of a graph.

- **Display Vertex** − display a vertex of a graph.

To know more about Graph, please read [Graph Theory Tutorial](#). We shall learn traversing a graph in coming chapters.

Loading [MathJax]/jax/output/HTML-CSS/jax.js