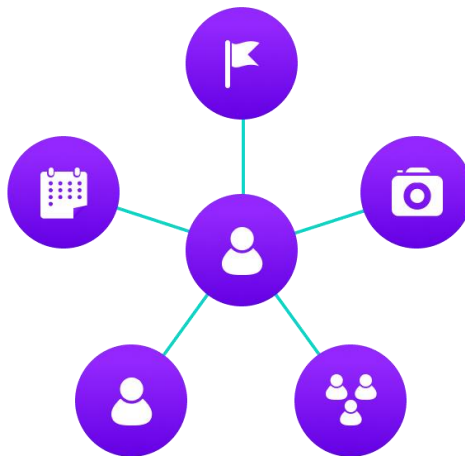


Graph Data Structure

A graph data structure is a collection of nodes that have data and are connected to other nodes.

Let's try to understand this through an example. On facebook, everything is a node. **That includes User, Photo, Album, Event, Group, Page, Comment, Story, Video, Link, Note...**anything that has data is a node.

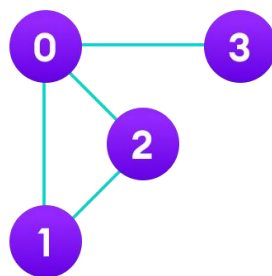
Every relationship is an edge from one node to another. Whether you post a photo, join a group, like a page, etc., a new edge is created for that relationship.



All of facebook is then a collection of these nodes and edges. This is because facebook uses a graph data structure to store its data.

More precisely, a graph is a data structure (V, E) that consists of

- **A collection of vertices V**
- **A collection of edges E , represented as ordered pairs of vertices (u,v)**



In the graph,

$$V = \{0, 1, 2, 3\}$$

$$E = \{(0,1), (0,2), (0,3), (1,2)\}$$

$$G = \{V, E\}$$

Graph Terminology

- **Adjacency:** A vertex is said to be adjacent to another vertex if there is an edge connecting them. Vertices 2 and 3 are not adjacent because there is no edge between them.
- **Path:** A sequence of edges that allows you to go from vertex A to vertex B is called a path. 0-1, 1-2 and 0-2 are paths from vertex 0 to vertex 2.
- **Directed Graph:** A graph in which an edge (u,v) doesn't necessarily mean that there is an edge (v, u) as well. The edges in such a graph are represented by arrows to show the direction of the edge.

Graph Representation

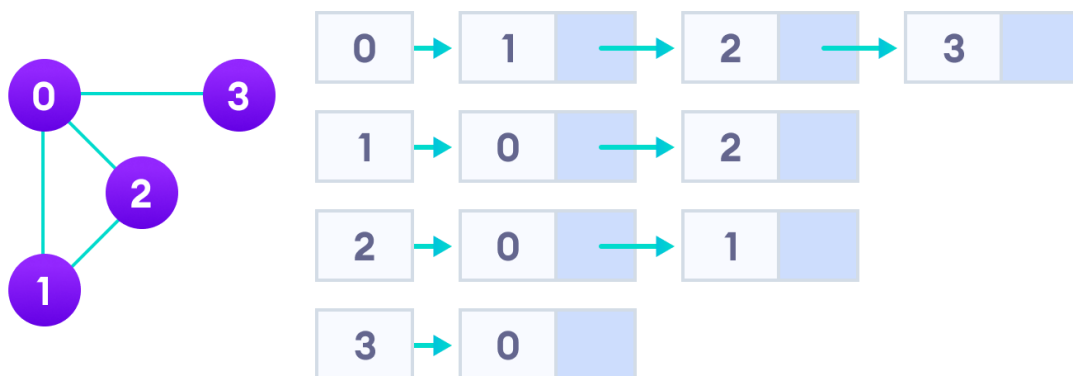
1. Adjacency Matrix

An adjacency matrix is a 2D array of $V \times V$ vertices. Each row and column represent a vertex.



2. Adjacency List

An adjacency list represents a graph as an array of linked lists.



Graph Operations

The most common graph operations are:

- Check if the element is present in the graph
- Graph Traversal

- Add elements(vertex, edges) to graph
- Finding the path from one vertex to another