PART-A

1. List any four terminologies in graph data structures.

Node (Vertex): A node, also known as a vertex, is a fundamental element in a graph. It represents a distinct entity or point in the graph. Nodes can contain data and are connected to other nodes through edges.

Edge: An edge is a connection between two nodes in a graph. It defines a relationship or interaction between the connected nodes. Edges can be directed or undirected and may have associated attributes, such as weights or labels.

Directed Graph (Digraph): A directed graph, or digraph, is a type of graph in which edges have a direction. Each edge has an origin (tail) node and a destination (head) node. Directed graphs are often used to model asymmetric relationships, like in road networks or information flow.

Weighted Graph: In a weighted graph, each edge is assigned a numerical value, referred to as a weight. These weights represent the cost, distance, or some other measure associated with the edge.

2. Mention the types of graphs with example

Undirected Graphs: A graph in which edges have no direction, i.e., the edges do not have arrows indicating the direction of traversal. Example: A social network graph where friendships are not directional.

Directed Graphs: A graph in which edges have a direction, i.e., the edges have arrows indicating the direction of traversal. Example: A web page graph where links between pages are directional.

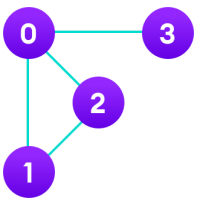
Weighted Graphs: A graph in which edges have weights or costs associated with them. Example: A road network graph where the weights can represent the distance between two cities.

Unweighted Graphs: A graph in which edges have no weights or costs associated with them. Example: A social network graph where the edges represent friendships.

Complete Graphs: A graph in which each vertex is connected to every other vertex. Example: A tournament graph where every player plays against every other player.

3. Show the adjacency matrix of the given graph

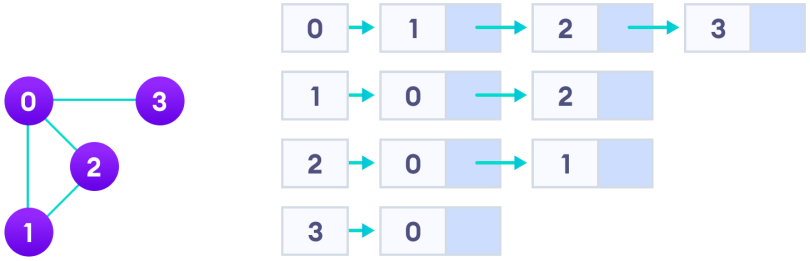
example:



i → / j ↓	0	1	2	3
0	0	1	1	1
1	1	0	1	0
2	1	1	0	0
3	1	0	0	0

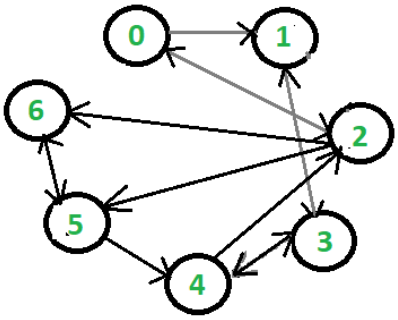
4. Demonstrate the adjacency list of the specified graph

Example



5. Find the In degree and Out degree of the particular given.

example



Output:

Vertex	In	Out
0	1	2
1	2	1
2	2	3
3	2	2
4	2	2
5	2	2
6	2	1