Graphs – Introduction to Graphs

1. Introduction

A **Graph** is a **non-linear data structure** used to represent relationships between objects.

Unlike trees, graphs can have **cycles**, **multiple paths**, and **no fixed root**.

Graphs are widely used in **networks, maps, social media, and real-world relationship modeling**.

---

2. What is a Graph?

A graph is made up of:

- **Vertices (Nodes)** – represent entities
- **Edges** – represent connections between vertices

Formally, a graph is represented as:

```
G = (V, E)
```

Where:

- V = set of vertices
- E = set of edges

---

3. Basic Structure of a Graph

Example:

```
A ─── B
|   |
C ─── D
```

- A, B, C, D are vertices

- Lines between them are edges

4. Key Terminology in Graphs

Term	Description
Vertex	A node in the graph
Edge	Connection between two vertices
Adjacent Vertices	Vertices connected by an edge
Degree	Number of edges connected to a vertex
Path	Sequence of vertices connected by edges
Cycle	Path that starts and ends at the same vertex
Connected Graph	Graph where all vertices are reachable
Disconnected Graph	Graph with unreachable vertices

5. Types of Graphs

1️⃣ Undirected Graph

- Edges have no direction

```
A — B
```

2️⃣ Directed Graph (Digraph)

- Edges have direction

```
A → B
```

3️⃣ Weighted Graph

- Edges have weights (cost, distance, etc.)

4️⃣ Unweighted Graph

- Edges have no weights

6. Directed vs Undirected Graph

Feature	Directed	Undirected
Direction	Yes	No
Edge Representation	A → B	A — B
Use Case	Web links	Social networks

7. Degree of a Vertex

- **Undirected Graph:**

  Degree = number of edges connected
- **Directed Graph:**
  - In-degree → incoming edges
  - Out-degree → outgoing edges

8. Graph Representation Methods

Graphs can be represented using:

1️⃣ Adjacency Matrix

- 2D matrix representation
- Uses more memory

2️⃣ Adjacency List

- List of connected vertices
- Memory efficient
- Most commonly used

9. Graph vs Tree

Feature	Tree	Graph
Cycles	No	Yes
Root	One root	No fixed root
Paths	One unique path	Multiple paths
Structure	Hierarchical	Network-like

10. Why Graphs Are Important?

Graphs are used to:

- Model real-world networks
- Represent relationships
- Solve shortest path problems
- Analyze connections and dependencies

11. Applications of Graphs

- Social media networks
- Road and map navigation
- Computer networks
- Web page linking
- Recommendation systems
- AI and machine learning
- Operating systems (process scheduling)

12. Advantages of Graphs

- Flexible structure
- Represents complex relationships
- Efficient for network-based problems

- Powerful modeling tool

13. Limitations of Graphs

- Complex implementation
- Higher memory usage
- Traversal logic is difficult
- Requires careful cycle handling

14. Summary

- Graph is a non-linear data structure
- Consists of vertices and edges
- Can be directed or undirected
- Can have cycles and multiple paths
- Used extensively in real-world problems