1. Terminology & Representations of Graph:

Graph: A collection of nodes (vertices) connected by edges.

Terminology: Nodes, edges, degree, adjacency, path, cycle, etc.

Representations:

Adjacency Matrix: A 2D array where each element represents the presence or absence of an edge between two vertices.

Adjacency List: A collection of lists where each list represents the neighbors of a vertex.

Edge List: A list of all edges in the graph, each containing the vertices it connects.

2. Graphs & Multi-Graphs, Directed Graphs:

Graphs: Collections of nodes and edges where edges may or may not have directionality.

Multi-Graphs: Graphs that allow multiple edges between the same pair of vertices.

Directed Graphs (Digraphs): Graphs where edges have a direction, indicating a one-way relationship between vertices.

3. Sequential Representations of Graphs:

Adjacency Matrices: Representing graphs using 2D arrays, with rows and columns corresponding to vertices and matrix elements indicating edge presence.

Adjacency Lists: Storing each vertex's neighbors in linked lists or arrays, providing a more memory-efficient representation for sparse graphs.

4. Traversal, Connected Components, and Spanning Trees:

Traversal: Techniques like Depth-First Search (DFS) and Breadth-First Search (BFS) to visit all nodes in a graph.

Connected Components: Subgraphs where every pair of vertices is connected by a path.

Spanning Trees: Subgraphs that are trees and include all vertices of the original graph, useful for ensuring connectivity and efficient data transmission.

5. Minimum Cost Spanning Trees:

Minimum Spanning Tree (MST): A spanning tree of a weighted graph with the minimum possible total edge weight.

Algorithms: Prim's Algorithm and Kruskal's Algorithm are commonly used to find MSTs.

Applications: Network design, clustering, and optimization problems.

6. File Handling:

Physical Storage Media File Organization: Storing and organizing data on physical storage devices like disks.

Organization of Records into Blocks: Grouping data records into blocks for efficient storage and retrieval.

Sequential Files: Storing records in sequential order, facilitating sequential access but limiting random access.

Indexing and Hashing: Techniques for optimizing search operations by creating indexes or hash tables for fast retrieval.

Primary and Secondary Indices: Index structures that provide access paths to data records based on primary or secondary keys.