Course: Advanced Algorithm and Problem Solving

WEEK 7 Graph Data Structures and Basic Algorithms

Lemlem Kassa (Ph.D.)

Addis Ababa Science and Technology University, Ethiopia

May, 2025

Week 7: Graph Data Structures and Basic Algorithms

Content

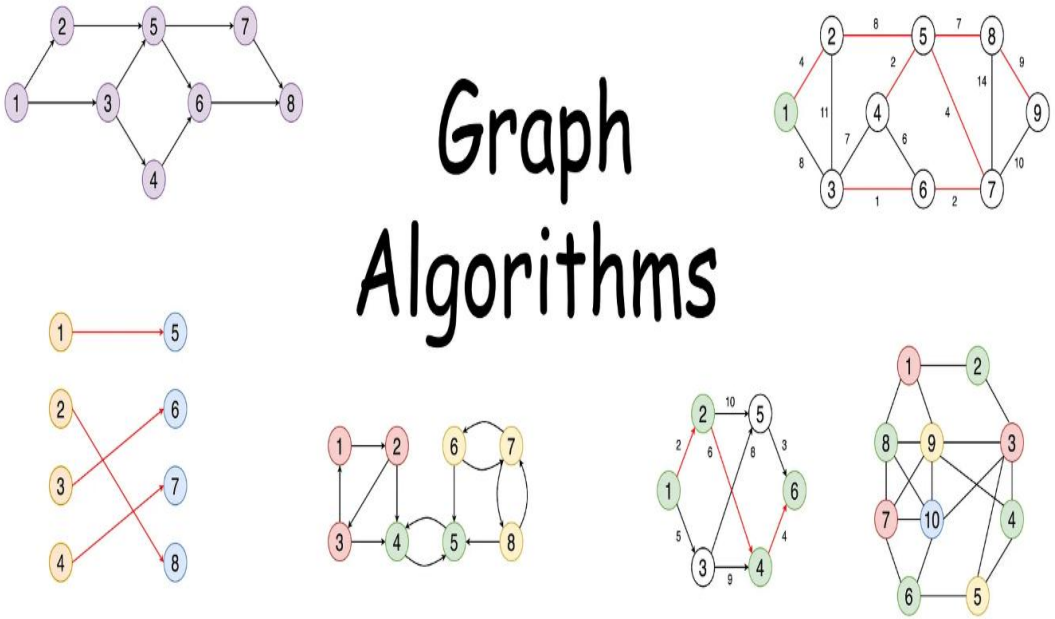
- Introduction to graph
- Types of graph data structure
- Graph algorithms
  - Beneath first search
  - Depth first Search
- Representations of Graph Data Structure

Lecture Learning Outcome

- Understand graph algorithms and the tools in data structures
- Differentiate types of graph data structure
- Understand different types of graph algorithms
- Understand representations of graph data structure
- Understand the difference between tree and graph algorithms
- Comprehend applications of graph data structure

Introduction to Graph

- Graphs have become a powerful means of modelling and capturing data in real-world scenarios such as social media networks, web pages and links, and locations and routes in GPS.
- If we have a set of objects that are related to each other, then we can represent them using a graph.

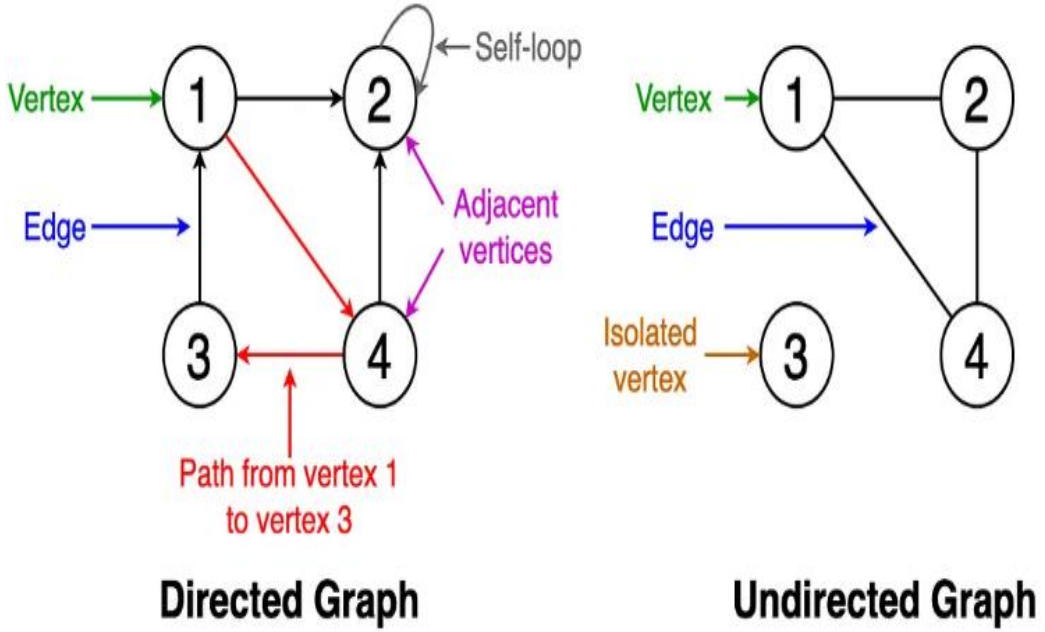


[1] Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Introduction to Graph …Cont'd

Basic definitions related to graphs:-

- **Order:** The number of vertices in the graph
- **Size:** The number of edges in the graph
- **Vertex degree:** The number of edges that are incident to a vertex
- **Isolated vertex:** A vertex that is not connected to any other vertices in the graph
- **Self-loop**: An edge from a vertex to itself



[1] Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Introduction to Graph …Cont'd

…….Basic definitions related to graphs:-

- **Directed graph:** A graph where all the edges have a direction indicating what is the start vertex and what is the end vertex
- **Undirected graph:** A graph with edges that have no direction
- **Weighted graph:** Edges of the graph has weights
- **Unweighted graph:** Edges of the graph has no weights

[1] Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Introduction to Graph …Cont'd

- Graph algorithms are essential tools in data structures, used to solve problems including connections, paths, and relationships between different entities.
- Understanding graph algorithms is crucial for solving complex problems such as
  - finding the shortest path,
  - detecting cycles, and
  - determining the flow in a network.

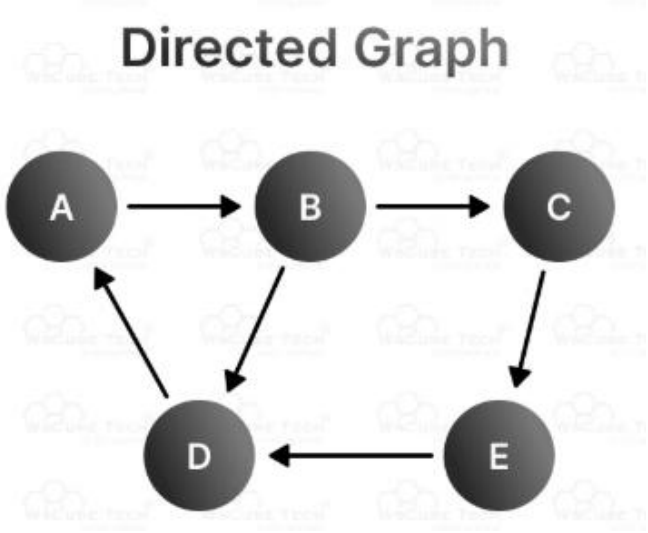
Types of Graph Data Structure

Types of Graph Data Structure

Graphs can be categorized based on their characteristics and properties:

1. Directed Graph

- In a directed graph, edges have a direction, meaning they go from one node to another in a specific way.



Example:

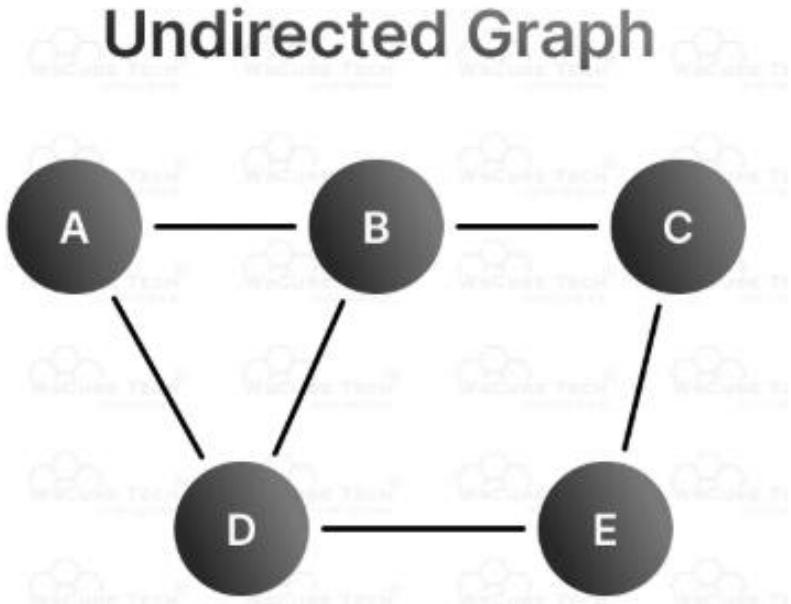
- If Alice follows Bob in Twitter network , there is an edge from Alice to Bob but not necessarily from Bob to Alice.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Types of Graph Data Structure …Cont'd

2. Undirected Graph

- In an undirected graph, edges do not have a direction.
- They simply connect two nodes without any particular order.



Example:

- Think of a Facebook friendship where if Alice is friends with Bob, then Bob is also friends with Alice. The edge goes both ways.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

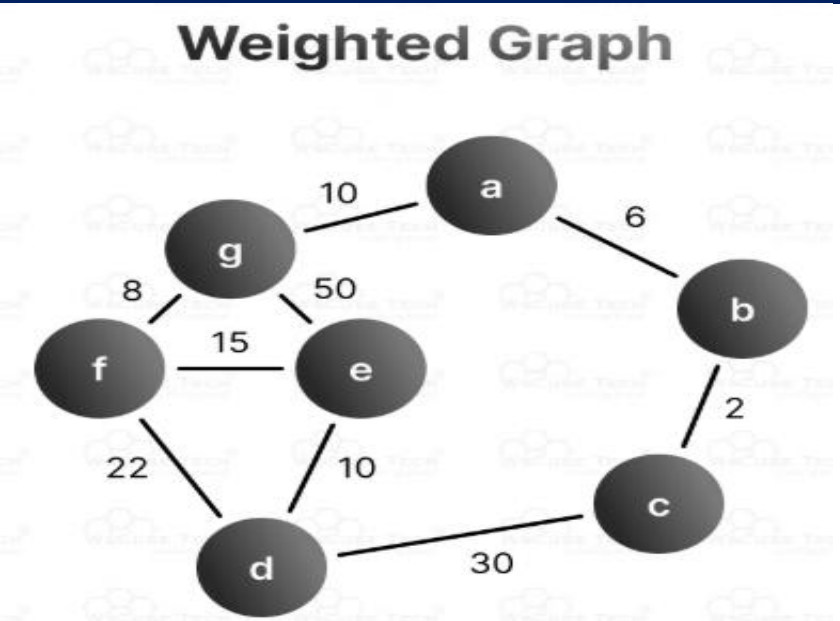
Types of Graph Data Structure …Cont'd

3. Weighted Graph

- In a weighted graph, edges have weights or costs associated with them.
- These weights can represent distances, costs, or any other metric.

Example:

- A road map where the weights on the edges represent the distance between cities.

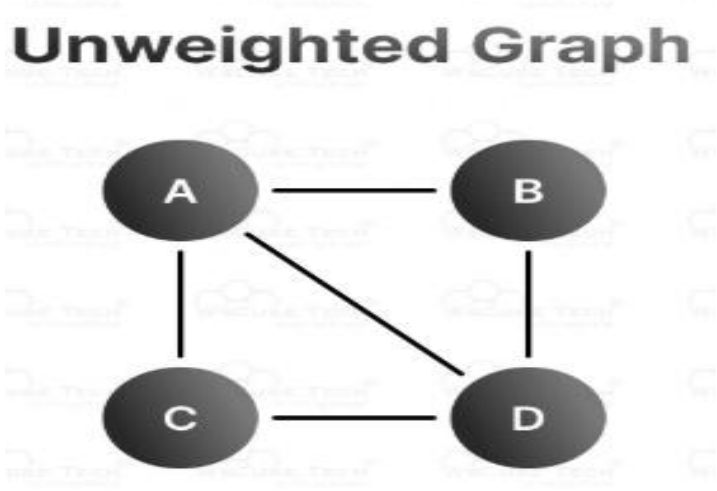


[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Types of Graph Data Structure …Cont'd

4. Unweighted Graph

- In an unweighted graph, all edges have the same weight, typically considered as 1.

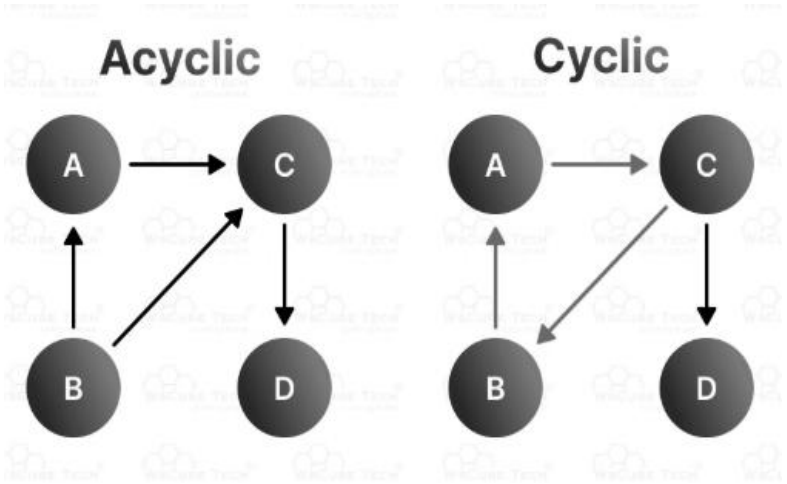


Example:

- A simple social network where each friendship has the same importance.

5. Cyclic Graph

- A cyclic graph contains at least one cycle, meaning we can start at a node and follow a path that leads back to the same node
- An acyclic graph does not contain any cycles.



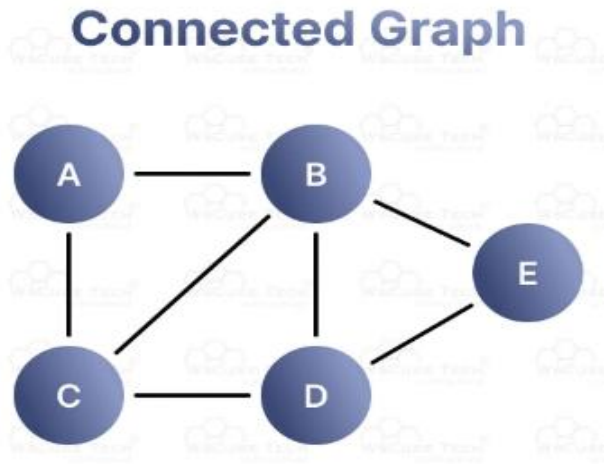
Example:

- A graph representing routes between cities where some routes form a loop.

Types of Graph Data Structure …Cont'd

6. Connected Graph

- A connected graph has a path between every pair of nodes.

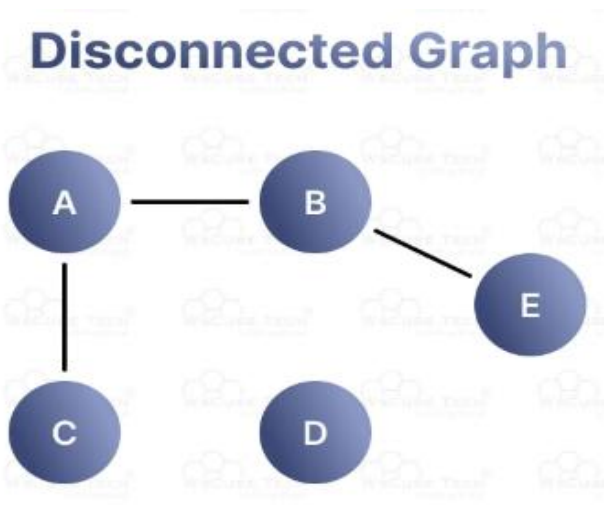


Example:

- A small network where every computer can reach every other computer directly or indirectly.

7. Disconnected Graph

- A disconnected graph has at least one pair of nodes with no path between them.



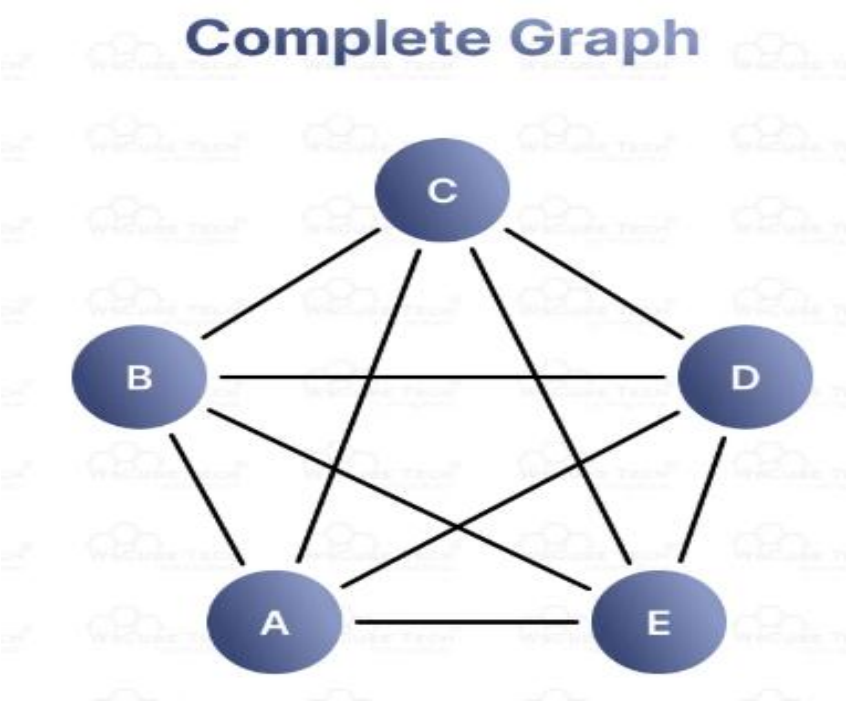
Example:
A network of isolated groups of people.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Types of Graph Data Structure …Cont'd

8. Complete Graph

- In a complete graph, there is an edge between every pair of nodes.



Example:

- A small social network where everyone is friends with everyone else.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Graph algorithms

- Graph algorithms help us understand and find information about the connections between the nodes of a graph.
- Used to figure out things like the shortest path between two points, how to connect all points with the least number of connections, or how to organize tasks in order.
- Graph algorithms are like the tools that help to plan the best route, making sure we use the shortest path or visit every spot in the most efficient way.
- They are like guides that help us solve problems involving connections, routes, or relationships between different points, making sure we do so in the best possible way.

[1]. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Graph algorithms …Cont'd

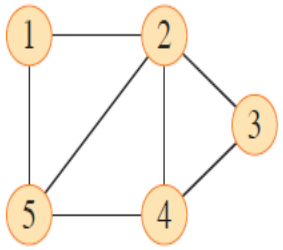
Graph in Data Structure

- **It consists of two main components:**
  - **Vertices (or Nodes):** These are the individual objects or points in the graph.
    - Each vertex represents an entity, like a city in a map, a user in a social network, or a computer in a network.
  - **Edges:** These are the connections between the vertices. An edge represents a relationship between two vertices, like a road connecting two cities, a friendship between two users, or a cable connecting two computers.

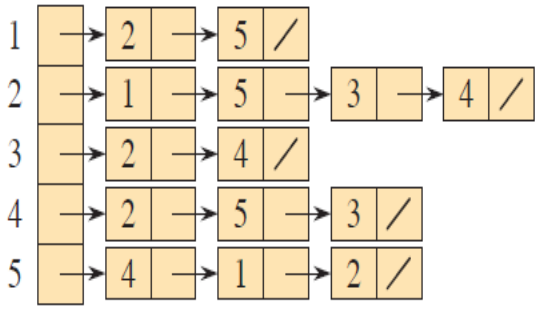
- Graphs are used to model complex relationships and connections in a simple and visual way, making them very powerful in solving many real-world problems.

Graph algorithms …Cont'd

Example : Representations of graphs



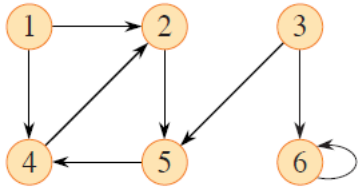
(a)



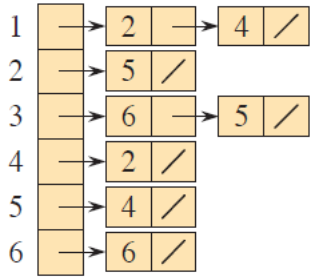
(b)

	1	2	3	4	5
1	0	1	0	0	1
2	1	0	1	1	1
3	0	1	0	1	0
4	0	1	1	0	1
5	1	1	0	1	0

(c)



(a)



(b)

	1	2	3	4	5	6
1	0	1	0	1	0	0
2	0	0	0	0	1	0
3	0	0	0	0	1	1
4	0	1	0	0	0	0
5	0	0	0	1	0	0
6	0	0	0	0	0	1

(c)

- Two representations of an undirected graph. **(a)** An undirected graph G with 5 vertices and 7 edges. **(b)** An adjacency-list representation of G. **(c)** The adjacency-matrix representation of G

- Two representations of a directed graph. **(a)** A directed graph G with 6 vertices and 8 edges. **(b)** An adjacency-list representation of G. **(c)** The adjacency-matrix representation of G.

[3] . Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press, 2009.

Classification of Graph Algorithms

Traversal Algorithms

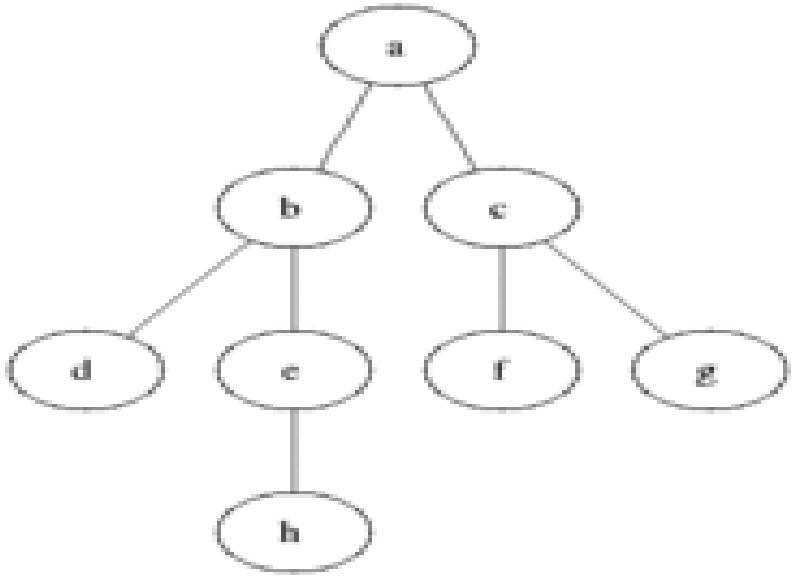
- *Traversing* or *searching* is one of the fundamental operations which can be performed on graphs.
  - Depth-First Search (DFS)
  - Breadth-First Search (BFS)

[1]. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Graph algorithms …Cont'd

Breadth-first search (BFS)

- We start at a particular vertex and explore all of its neighbours at the present depth before moving on to the vertices in the next level.
- Unlike trees, graphs can contain cycles (a path where the first and last vertices are the same). Hence, we have to keep track of the visited vertices.
- When implementing BFS, we use a queue data structure.



https://www.freecodecamp.org/news/graph-algorithms-and-data-structures-explained-with-java-and-c-examples/

- The Figure denotes the animation of a BFS traversal of an example graph.

[1]. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Graph algorithms …Cont'd

Applications of Breadth-first search

- Used to determine the shortest paths and minimum spanning trees.
- Used by search engine crawlers to build indexes of web pages.
- Used to search on social networks.
- Used to find available neighbour nodes in peer-to-peer networks such as BitTorrent.

[1]. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Graph algorithms …cont'd

Depth First Search algorithm

- The fundamental graph traversal technique.
- It works like an essential tool for solving problems like finding connected components, detecting cycles, and performing topological sorting.
- The DFS algorithm is a way to explore all the nodes (or vertices) in a graph or a tree.
- In the depth-first search algorithm, we start at a specific node, and then we go as far as possible along each branch before going back (this is called backtracking).
- We keep doing this until we have visited all the nodes.

Graph algorithms …cont'd

- ***Depth-first search*** (DFS) starts from a particular vertex and explore as far as possible along each branch before retracing back (backtracking).
- DFS also have to keep track of the visited vertices. When implementing DFS, we use a stack data structure to support backtracking.
- The Figure denotes the animation of a DFS traversal how it traverses to the depths and backtracks.

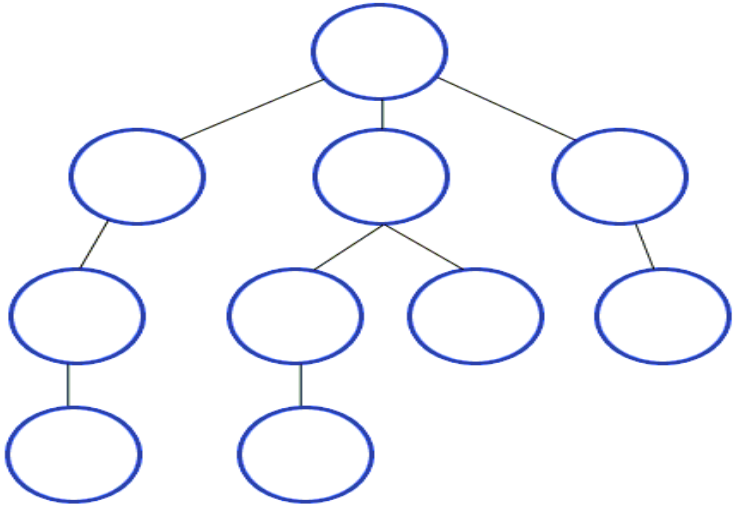


Fig. https://www.freecodecamp.org/news/graph-algorithms-and-data-structures-explained-with-java-and-c-examples/

[1]. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3

Graph algorithms …cont'd

Depth-first search (DFS)

- Imagine a maze with many paths. We start at the entrance, and decide to follow one path until we can't go any further.
- If we reach a dead end, we go back to the last point where we have another choice and try a different path.
- Keep doing this until we have explored the entire maze.
- This is exactly how the DFS algorithm in data structure works—it explores all possible paths in a graph one by one, backtracking when necessary, until all nodes are visited.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Graph algorithms …cont'd

Applications of DFS (Depth-First Search)

- **Pathfinding in Mazes**: DFS is used to find a path from the start to the end of a maze by exploring all possible routes.
- **Detecting Cycles in Graphs**: DFS helps in detecting cycles in both directed and undirected graphs by tracking back edges.
- **Topological Sorting**: DFS is used in topological sorting of a Directed Acyclic Graph (DAG) to order tasks or jobs with dependencies.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Graph algorithms …cont'd

,….. Applications of DFS (Depth-First Search)

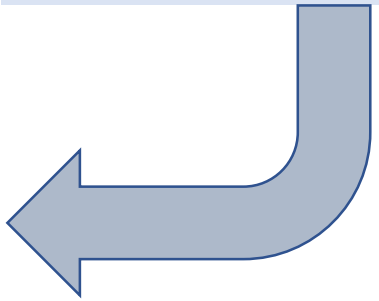
- **Finding Connected Components**: In an undirected graph, DFS can identify all connected components by exploring each vertex and its neighbors.
- **Tree Traversals:** DFS is the basis for tree traversals like pre-order, in-order, and post-order traversals.
- **Strongly Connected Components (SCCs):** In directed graphs, DFS is used to find all SCCs, which are subsets of vertices where each vertex is reachable from every other vertex in the subset.

[2]. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Graph algorithms …Cont'd

Feature	Tree	Graph
Definition	A tree is a hierarchical data structure with a root node and child nodes forming a parent-child relationship.	A graph is a collection of nodes (vertices) connected by edges, representing relationships between the nodes.
Structure	Hierarchical (parent-child relationship)	Network-like (arbitrary connections between nodes)
Root Node	Has a single root node	Does not necessarily have a root node
Path	There is exactly one path between any two nodes.	There can be multiple paths between any two nodes.
Cycles	Trees do not contain cycles (acyclic).	Graphs can contain cycles (cyclic) or be acyclic.
Connectivity	Trees are connected (all nodes are reachable from the root).	Graphs can be connected or disconnected.
Parent-Child Relationship	Each node (except the root) has exactly one parent.	Nodes can have multiple parents.
Leaf Nodes	Nodes with no children are called leaf nodes.	Graphs do not have the concept of leaf nodes.
Types	Binary Tree, AVL Tree, B-Tree, etc.	Directed Graph, Undirected Graph, Weighted Graph, etc.
Usage	Used to represent hierarchical data, such as file systems, organization structures, and XML documents.	Used to represent network structures, such as social networks, computer networks, and transportation systems.
Degree	In a binary tree, each node has a maximum degree of 2.	Nodes in a graph can have any number of edges.
Traversal Methods	In-order, Pre-order, Post-order, Level-order	Depth-First Search (DFS), Breadth-First Search (BFS)
Example	Family Tree, Binary Search Tree	Road Network, Social Network
Applications	Efficient searching, sorting, and hierarchical data storage	Network analysis, shortest path algorithms, and connectivity checks

Tree Vs Graph Data structures



Representations of Graph Data Structure

- Graphs can be represented in several ways.
- The three most common methods are
  - ***Adjacency matrix, Adjacency list, and Edge list.***

1. Adjacency Matrix

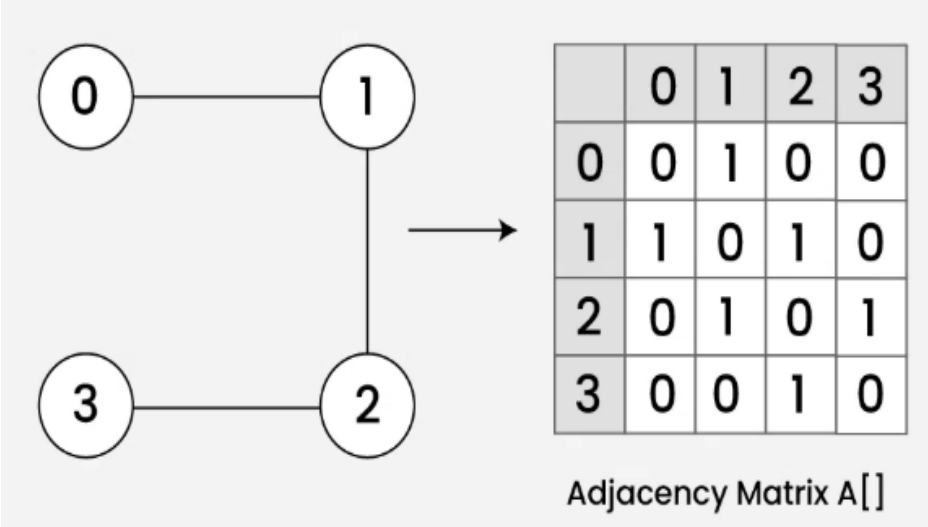
- An **adjacency matrix** is a 2D array used to represent a graph.
- Each cell in the matrix indicates whether an edge exists between a pair of vertices.
- If an edge exists, the cell contains a value (typically 1 for unweighted graphs or the weight of the edge for weighted graphs); otherwise, it contains 0.

[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa/

Representations of Graph Data Structure …Cont'd

1. Adjacency Matrix for Undirected and Unweighted graph:

- Consider an Undirected and Unweighted graph **G** with **4 vertices** and **3 edges**.
- For the graph G, the adjacency matrix would look like:



	0	1	2	3
0	0	1	0	0
1	1	0	1	0
2	0	1	0	1
3	0	0	1	0

Adjacency Matrix A[]

Here's how to interpret the matrix:

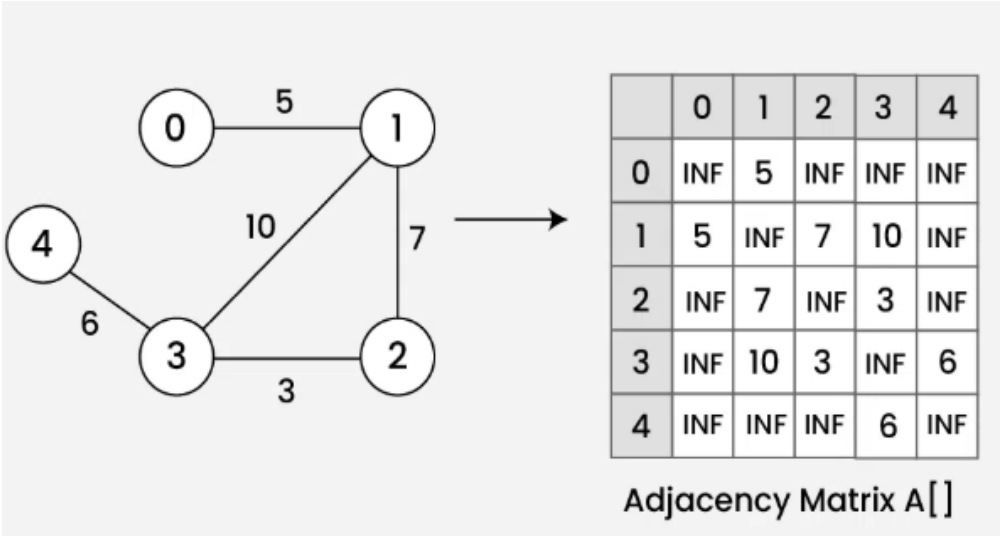
•A[i][j] = 1, there is an edge between vertex i and vertex j.

•A[i][j] = **0,** there is NO edge between vertex i and vertex j.

Representations of Graph Data Structure …Cont'd

2. Adjacency Matrix for Undirected and Weighted graph:

- Consider an Undirected and Weighted graph **G** with **5 vertices** and **5 edges**.
- For the graph G, the adjacency matrix would look like:

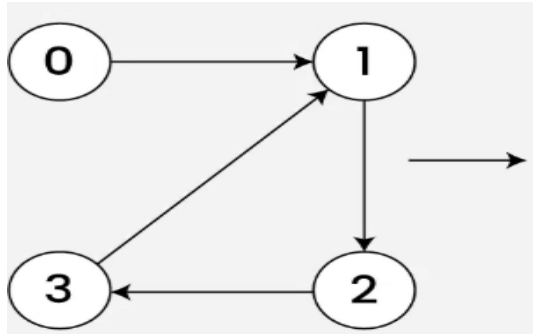


	0	1	2	3	4
0	INF	5	INF	INF	INF
1	5	INF	7	10	INF
2	INF	7	INF	3	INF
3	INF	10	3	INF	6
4	INF	INF	INF	6	INF

Adjacency Matrix A[]

3. Adjacency Matrix for Directed and Unweighted graph:

- Consider an Directed and Unweighted graph G with 4 vertices and 4 edges.
- For the graph G, the adjacency matrix would look like:



	0	1	2	3
0	0	1	0	0
1	0	0	1	0
2	0	0	0	1
3	0	1	0	0

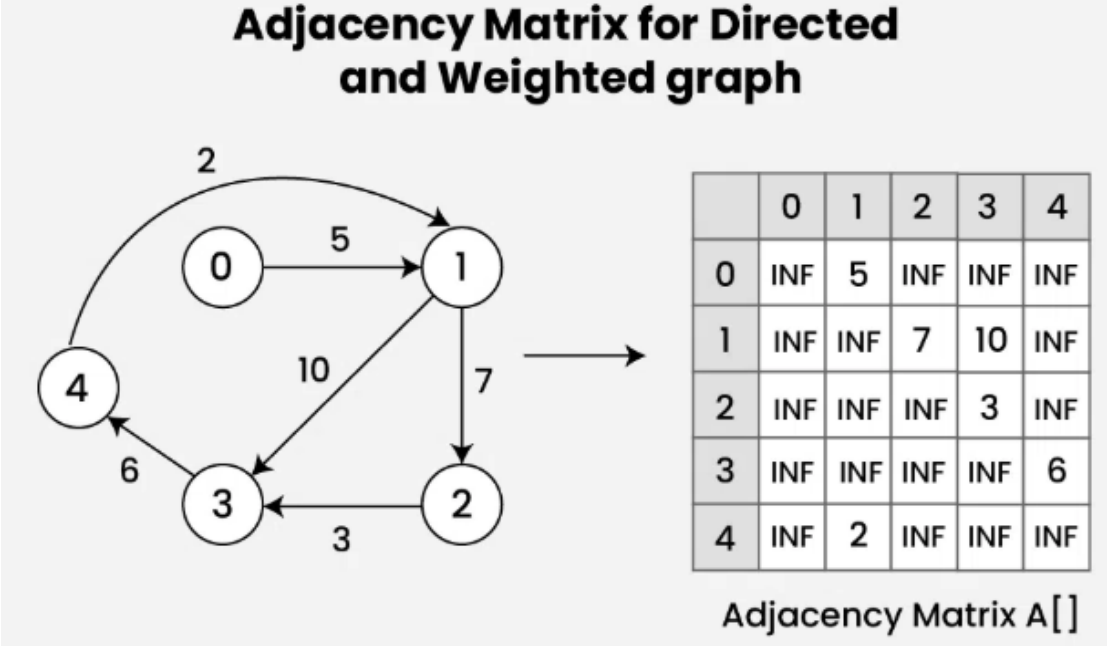
Adjacency Matrix A[]

[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

Representations of Graph Data Structure …Cont'd

4. Adjacency Matrix for Directed and Weighted graph:

- Consider an Directed and Weighted graph G with 5 vertices and 6 edges.
- For the graph G, the adjacency matrix would look like:



	0	1	2	3	4
0	INF	5	INF	INF	INF
1	INF	INF	7	10	INF
2	INF	INF	INF	3	INF
3	INF	INF	INF	INF	6
4	INF	2	INF	INF	INF

[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

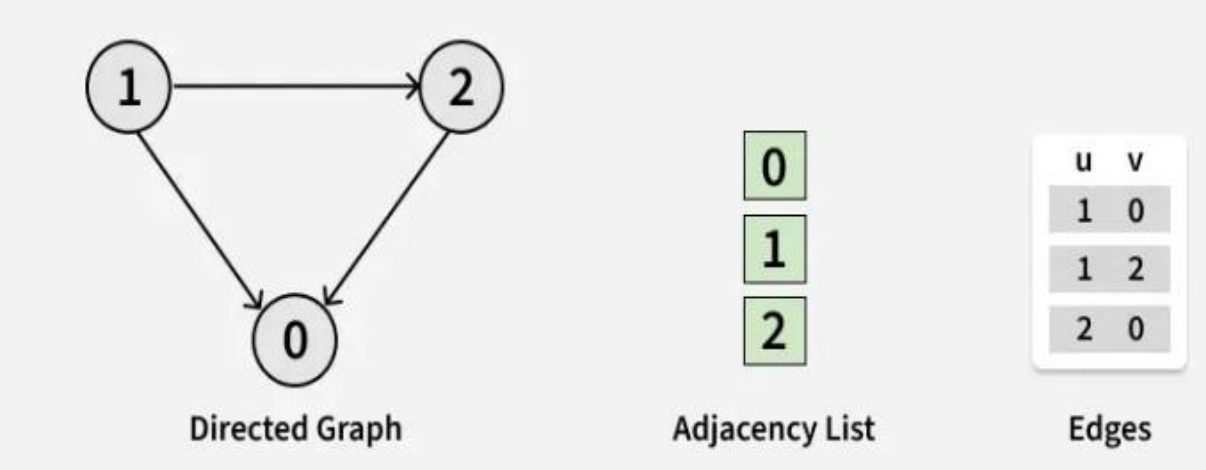
Representations of Graph Data Structure …Cont'd

Adjacency List Representation

- It is a data structure used to represent a graph where each node in the graph stores a list of its neighboring vertices.

1. Adjacency List for Directed graph:

- Consider an Directed and Unweighted graph **G** with 3 **vertices** and **3 edges**

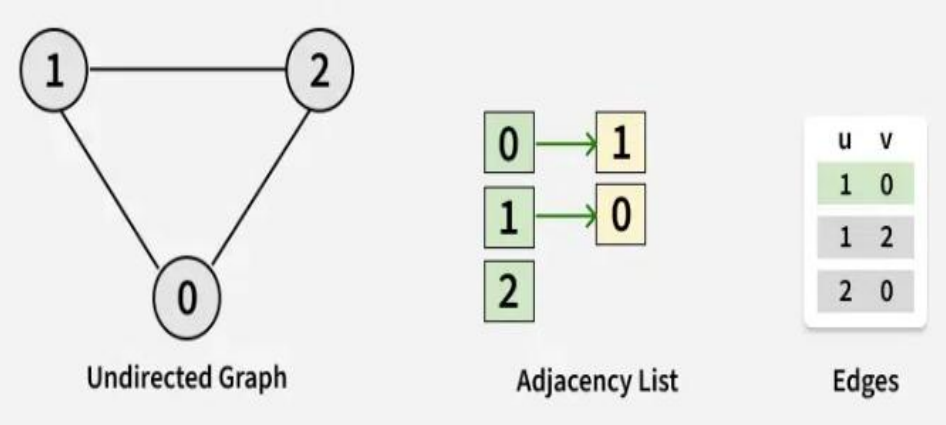


[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

Representations of Graph Data Structure …Cont'd

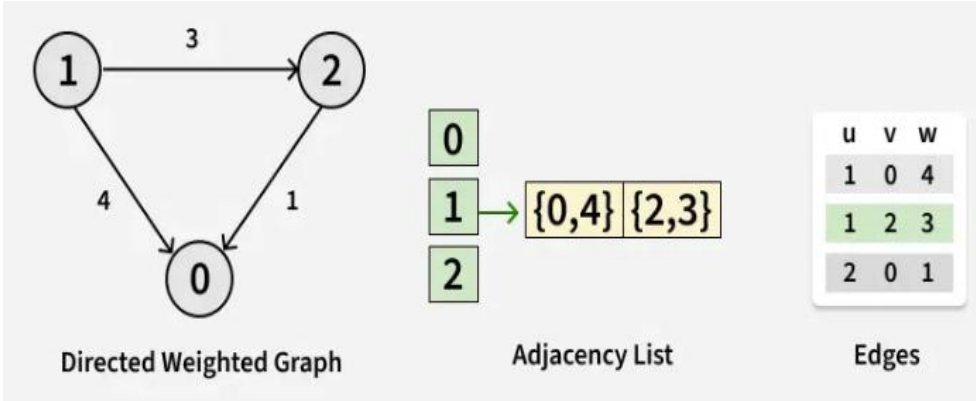
2. Adjacency List for Undirected graph:

- Consider an Undirected and Unweighted graph **G** with 3 **vertices** and **3 edges**.



3. Adjacency List for Directed and Weighted graph:

- Consider an Directed and Weighted graph **G** with 3 **vertices** and **3 edges**.

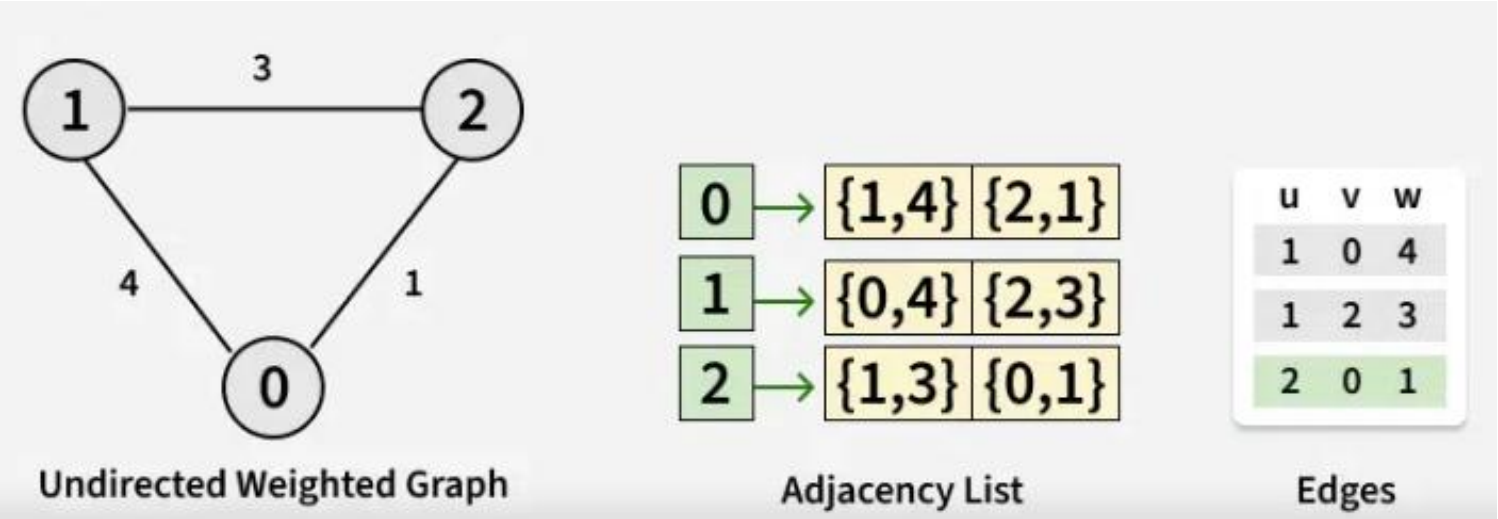


[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

Representations of Graph Data Structure …Cont'd

4. Adjacency List for Undirected and Weighted graph:

- Consider an Undirected and Weighted graph **G** with 3 **vertices** and **3 edges**

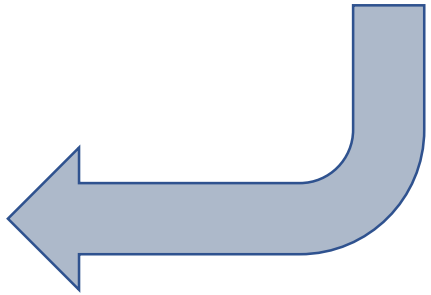


[4]. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

Representations of Graph Data Structure …Cont'd

Application	Explanation	Example
Social Networks	Representing user interactions and connections.	Facebook, LinkedIn, Twitter
Computer Networks	Optimizing data transfer and network structure.	Internet routers and switches
Web Page Ranking	Ranking web pages based on importance.	Google PageRank
Transportation	Finding shortest paths and optimizing travel routes.	GPS navigation systems
Scheduling	Planning and optimizing task execution order.	Gantt charts, PERT charts
Biology	Modeling protein interactions and genetic networks.	Protein-protein interaction networks
Game Development	Designing game levels, AI, and player interactions.	Pathfinding algorithms in game AI
Electrical Circuits	Modeling and analyzing electrical circuits.	Circuit design software like SPICE
Dependency Resolution	Managing software project dependencies.	Package managers like npm, pip, Maven

Uses of Graph Data Structure (Applications)



2. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure

Summary

- Understanding graph algorithms is crucial for solving complex problems such as finding the shortest path, detecting cycles, and determining the flow in a network.
- Graphs have become a powerful means of modelling and capturing data in real-world scenarios such as social media networks, web pages and links, and locations and routes in GPS.
- The DFS algorithm is a way to explore all the nodes (or vertices) in a graph or a tree. In the depth-first search algorithm, we start at a specific node, and then go as far as possible along each branch before going back (this is called backtracking).
- The BFS algorithm, or Breadth-First Search algorithm, is a way to explore or search through a graph or a tree in data structures. It works by starting at one node, and then visiting all its neighbors first before moving on to their neighbors. This process continues until all nodes are visited.
- Graphs can be represented in several ways. The three most common methods are Adjacency matrix, Adjacency list, and Edge list.
- A tree is a hierarchical data structure with a root node and child nodes forming a parent-child relationship. Whereas graph is a collection of nodes (vertices) connected by edges, representing relationships between the nodes.

References

1. Graph Algorithms Visually Explained, Vijini Mallawaarachchi, Medium (TDS Archive),2020. https://medium.com/data-science/10-graph-algorithms-visually-explained-e57faa1336f3
2. Graph Data Structure: Types, Uses, Examples, Algorithms, WsCube Tech. https://www.wscubetech.com/resources/dsa/graph-data-structure.
3. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press, 2009.
4. Adjacency List Representation, GeeksforGeeks, Nov.2024. https://www.geeksforgeeks.org/adjacency-list-meaning-definition-in-dsa

Thank You!

For your attention