Course: Data Structures (CSE CS203A, 114-1)

Study Note: Graph

Student ID: 1133317 Student Name: 吳杰恩

Definition

A graph G(V, E) consists of:

1. V = set of vertices (nodes)
2. E = set of edges connecting vertices

Real-world Motivation: Google Maps

Concept	Google Maps Analogy
Node (Vertex)	Intersection or landmark (e.g., Neili Station, Yuan Ze University)
Edge (Link)	Road connecting two intersections
Weight	Distance, travel time, or cost
Path	Sequence of connected roads from start to destination
Graph traversal / search	Finding all reachable locations or the best route
Cycle	Round trip that ends at the starting point
Directed edge	One-way street
Undirected edge	Two-way road

Classification

Type	Description	Example
Undirected Graph	Edges have no direction	Friendship network
Directed Graph (Digraph)	Edges have direction	Instagram “following”
Weighted Graph	Each edge has a cost	Google Maps distance
Unweighted Graph	All edges equal	Board game map
Cyclic Graph	Has loops	City ring road
Acyclic Graph	No loops	Family tree
Connected Graph	Every node reachable	Road network
Disconnected Graph	Some nodes isolated	Islands without bridge

Graph Representation in Memory

Adjacency List / Adjacency Matrix

Adjacency List: Linked list 存放點和點之間鄰近資訊，似 Hash

Adjacency Matrix: 二維 array，可紀錄連線（誰指向誰）

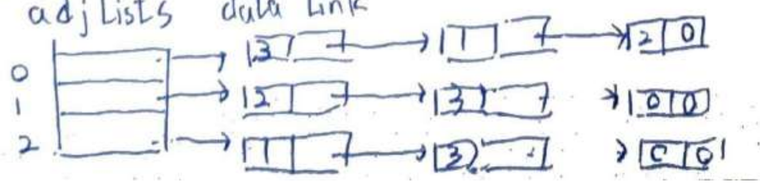
1. Adjacency List

(E)

2. Adjacency Matrix

pros & Cons

(n) 有



Graph Traversal

1. Graph vs. Tree

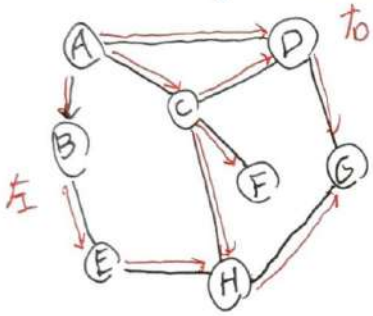
	Tree	Graph
Graph:	Always connected	May be disconnected
Cycles	No	Yes
Direction	Not directed	Directed / undirected
Hierarchy	Yes (rooted)	No inherent hierarchy

The Same:
A visited strategy
DFS
BFS
Systematic exploration of nodes
Difference:

2. Depth-First Search (DFS)

LIFO
stack

string node A



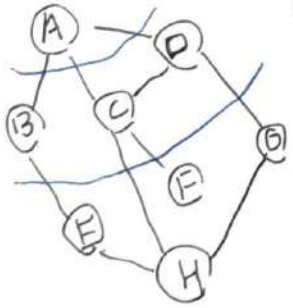
左 to

round 0
round 1
round 2
round 3

pop A → print

3. Bread-First Search (BFS)

By level ⇒ degree



starting node ⇒ A

A → B C D
↓ ↓ ↓
E H F G

FIFO
Quene (暫存輸出點)

round 0 [A]
round 1 [B | C | D]
round 2 [C | D | E]
round 3 [D | E | H | F]
round 4 [E | H | F | G]
round 5 [H | F | G | X] ↑輸出

★ visited[ ] 判斷 是否已輸出過
存放已輸出字母