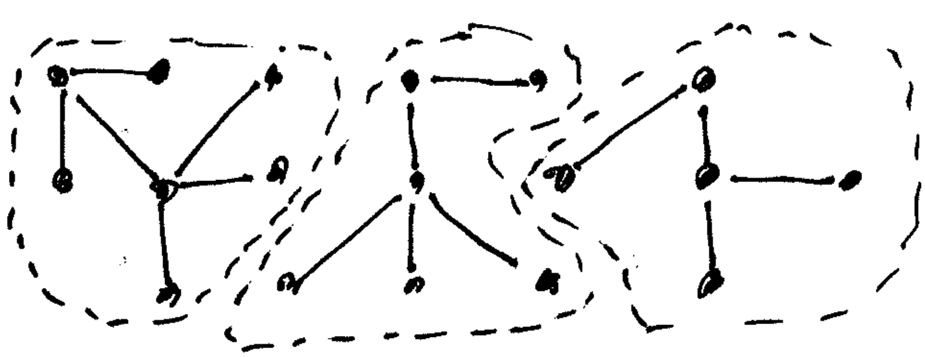
Defn

A graph G is called *acyclic* (or a *forest*) iff it contains no cycles. A graph ~~that~~ is both acyclic & connected is called a *Tree*

Ex.



#			
Vert.:	7	6	5
Edges:	6	5	4

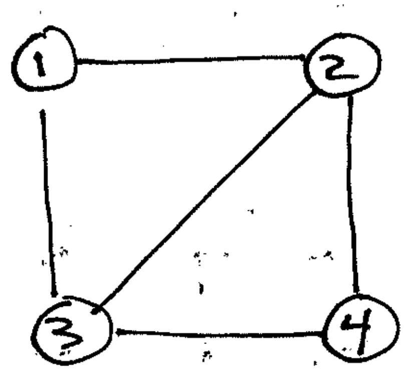
Representations of Graphs

$n = |V(G)|$, $m = |E(G)|$

- Incidence Matrix : $n \times m$
- Adjacency Matrix : $n \times n$
- Adjacency List : array of n lists

Ex graph $n=4$

G

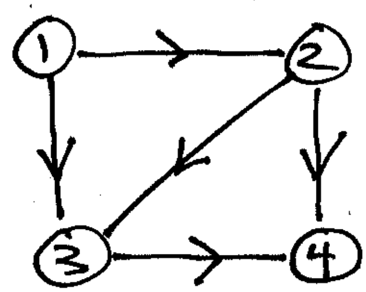


adj(G)

1 : 2 3
2 : 1 3 4
3 : 1 2 4
4 : 2 3

Ex digraph $n = 4$

D



adj(D)

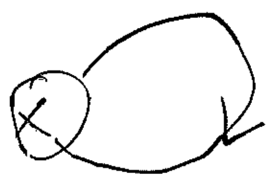
1 : 2 3
2 : 3 4
3 : 4
4 :

Note $D = (V, E)$ is a **digraph** iff

$$V \neq \phi$$

$$E \subseteq V \times V$$

(x, x)



(x, y)
(y, x)

} allowed in **digraphs**.

Graph Algorithms

Defn. We say $y \in V(G)$ is *reachable* from $x \in V(G)$ iff there exists

(1) an x-y Path (undirected)

(2) a directed x-y Path (directed)

Defn

$$\delta(x,y) = \begin{cases} \text{min len. of an } x\text{-}y \text{ Path} \\ \text{if } y \text{ is reachable from } x \\ \infty \quad \text{otherwise} \end{cases}$$

↑ distance from x to y.

Single source shortest Path (SSSP)

Problem :

Given a graph (digraph) G, and $s \in V(G)$ called the source, determine

(1) $\delta(s,x)$ for all $x \in V(G)$.

(2) if $\delta(s,x) < \infty$, find a shortest s–x path.

Breadth First Search (BFS)

solves SSSP.

It uses vertex attributes

- $\text{Color}[x] = \begin{cases} \text{white} \\ \text{gray} \\ \text{black} \end{cases}$

- distance[x] = estimate of $\delta(s, x)$

- Parent[x] encodes shortest paths.

Also uses FIFO queue: Q

Uses adj-list rep.

- adj[x]: list of vertices adjacent to x