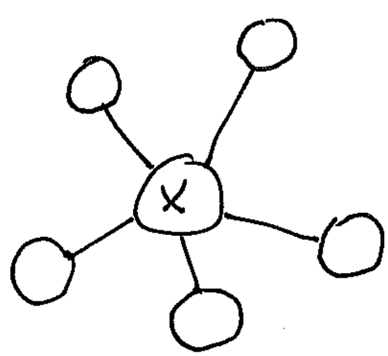
cse 101-01 4-13-23

Defn

if $x \in V(G)$, the degree of x is

$\deg(x) = \#$ of edges incident with x.

Ex.



$\deg(x) = 5$

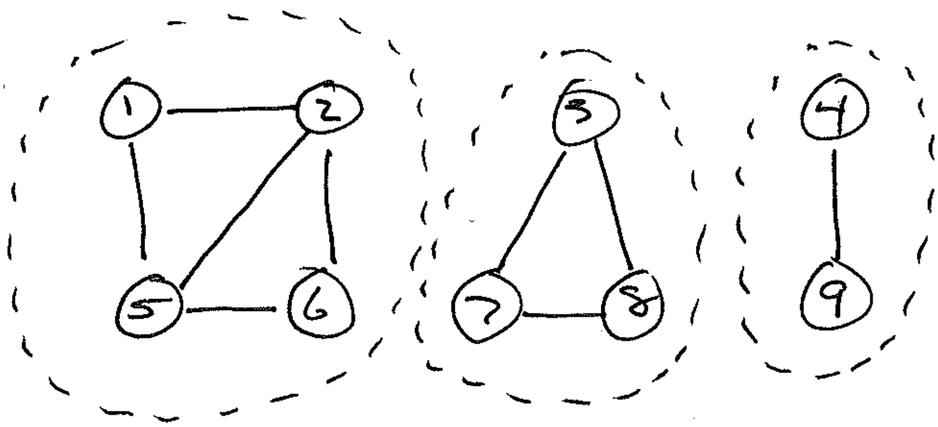
Handshake lemma:

$$\sum_{x \in V(G)} \deg(x) = 2|E(G)|$$

Defn

a graph G is called <u>connected</u> iff for any $x, y \in V(G)$ there exists an x-y Path in G.

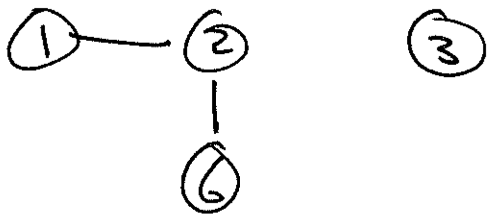
Ex.



disconnected

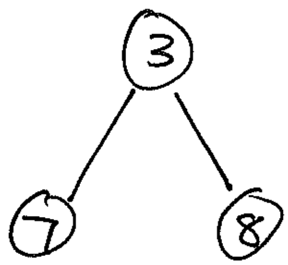
Defn A <u>Subgraph</u> H of G is a graph H with $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$

Ex. $(\{1,2,6,3\},\{12,26\})$



disconnected

Ex. $(\{3,7,8\},\{37,38\})$



connected

Ex. $(\{1,2,3,4\},\{49,12,37\})$

not a graph, so not a subgraph

Defn

A subgraph H of G is called a connected component of G iff

(1) H is connected.

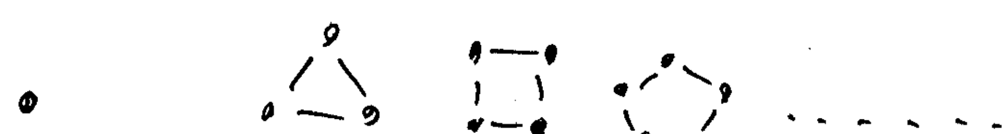
and

(2) H is maximal w.r.t. (1).

Defn

a graph G is acyclic (also a forest) iff it contains no cycles.

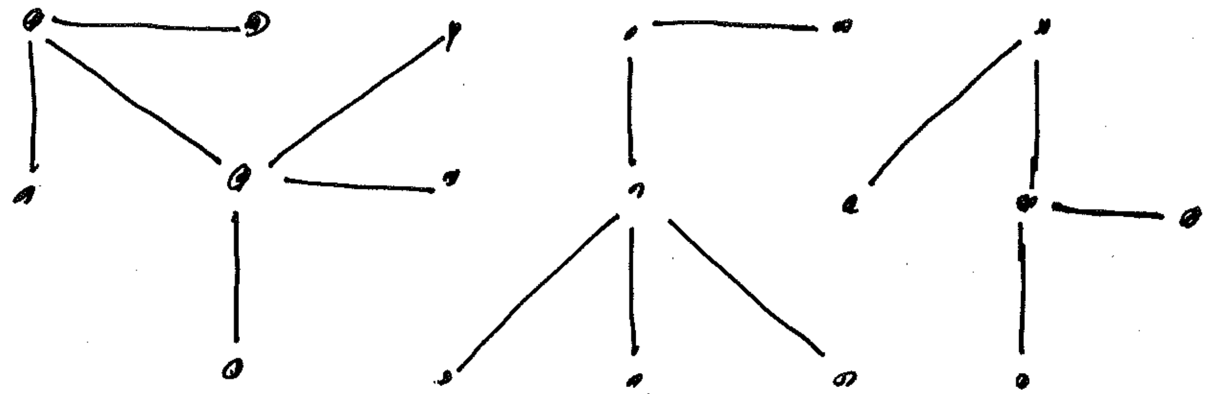
non-trivial closed Path



trivial Path

cycles - - - - - -

Ex.



#V:	7	6	5
#E:	6	5	4

Defn

A graph T is called a tree iff it is both acyclic and connected.

Lemma If T is a tree with n vertices, then T has $n-1$ edges.

Representations of graphs

- Incidence matrix: $n \times m$ matrix

$$V = \{x_1, x_2, \ldots, x_n\}$$

$$E = \{e_1, e_2, \ldots, e_m\}$$

<u>undirected</u>

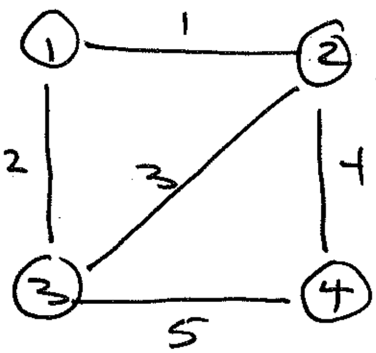
$$I_{ij} = \begin{cases} 1 & \text{if } x_i \text{ incident with } e_j \\ 0 & \text{otherwise} \end{cases}$$

<u>directed</u>

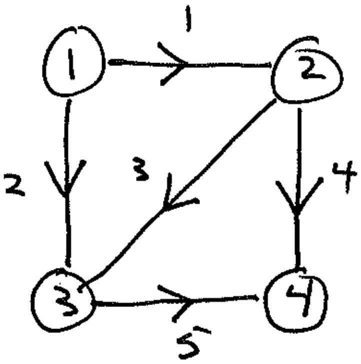
$$I_{ij} = \begin{cases} 1 & \text{if } e_j : \bigcirc \longrightarrow x_i \\ -1 & \text{if } e_j : x_i \longrightarrow \bigcirc \\ 0 & \text{otherwise} \end{cases}$$

Ex.

G



D



$$I = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad I = \begin{pmatrix} -1 & -1 & 0 & 0 & 0 \\ +1 & 0 & -1 & -1 & 0 \\ 0 & +1 & +1 & 0 & -1 \\ 0 & 0 & 0 & +1 & +1 \end{pmatrix}$$

- Adjacency matrix: $n \times n$ matrix

$V = \{x_1, x_2, \ldots, x_n\}$

undirected

$$A_{ij} = \begin{cases} 1 & \text{if } x_i \text{ adj. to } x_j \\ 0 & \text{otherwise} \end{cases}$$

directed

$$A_{ij} = \begin{cases} 1 & \text{if } (x_i) \rightarrow (x_j) \\ 0 & \text{otherwise} \end{cases}$$

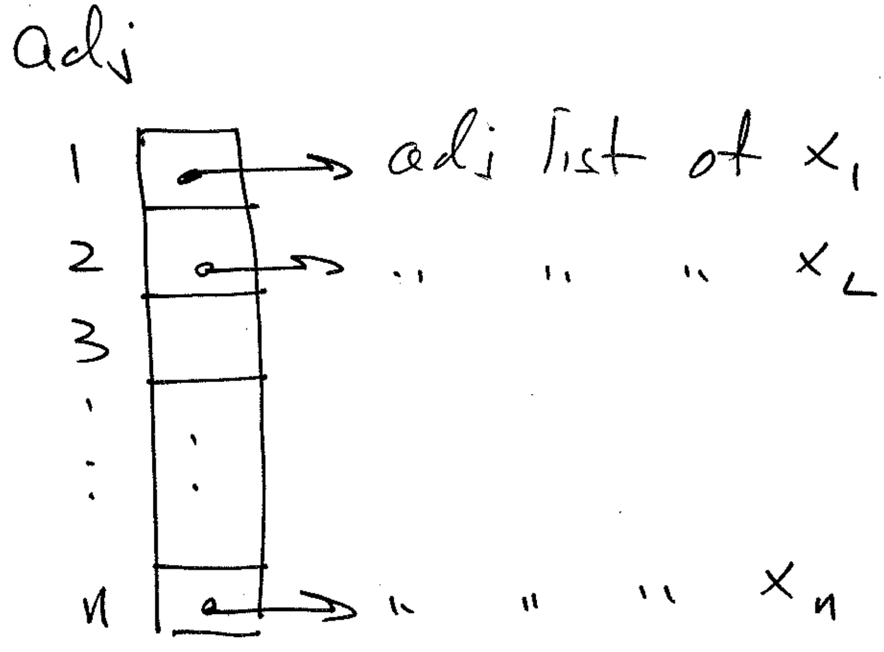
Ex.

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \qquad A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

- Adjacency List: array of n lists

undirected

$adj[i]$ = list of neighbors of x_i



directed

$adj[i]$ = list of terminii of edges having origin x_i

Ex.

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

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

Handout: Graph algorithms.

Defn Let G be a graph (or digraph), let $x, y \in V(G)$. The distance from x to y is

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

SSSP (single source shortest path)

Problem:

Given a graph G, and a source vertex $s \in V(G)$, determine

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

(2) for all x reachable from s, find an s-x path of length $\delta(s, x)$.

Bredth First Search (BFS)

Vertex attributes: for each $x \in V(G)$

- color[x]: white, gray, black
- distance[x]: $\delta(s, x)$, when complete.
- Parent[x]: Predecessor of x in a shortest s-x path.

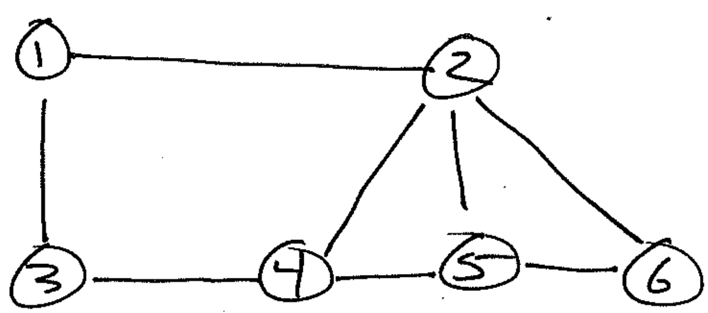
Also use a fifo queue: Q.

See: pseudocode

<u>Convention:</u>

Process adj lists in increasing vertex label order.

<u>Ex.</u>



$\boxed{s=3}$

	adj	color	d	p
1	2 3	w	∞	n
2	1 4 5 6	w	∞	n
3	1 4	g	0	n
4	2 3 5	w	∞	n
5	2 4 6	w	∞	n
6	2 5	w	∞	n