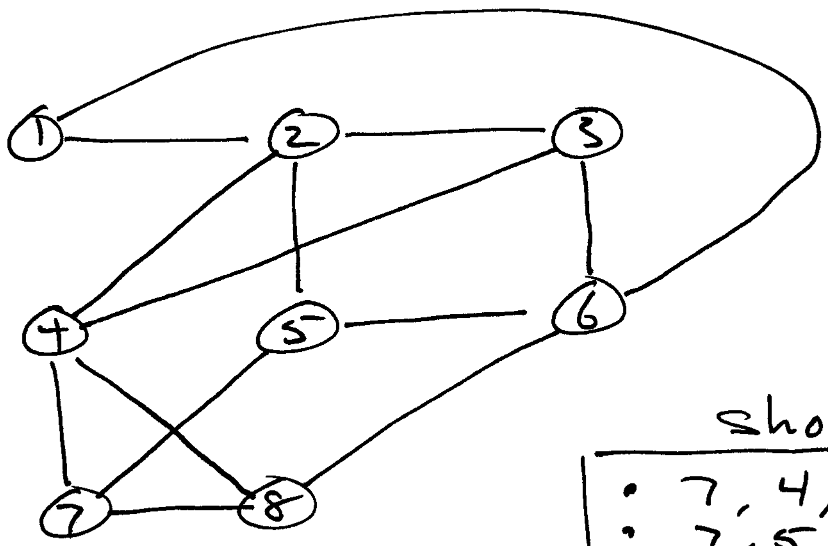
cse 101 1-14-21

Pa1 : ext. 1 day (last)

Pa2 : Posted. BFS

Previous example :

G



distance

$\delta(1,7) = 3$

shortest 7-1 Path

- 7, 4, 2, 1 ← found by BFS
- 7, 5, 2, 1
- 7, 8, 6, 1

<u>adj[G]</u> : adjacency list representation

1 : 2, 6
2 : 1, 3, 5
3 : 2, 4, 6
4 : 2, 3, 7, 8
5 : 2, 6, 7
6 : 1, 3, 5, 8
7 : 4, 5, 8
8 : 4, 6, 7

Breadth First Search (BFS)

- solves SSSP
- uses adj list rep.
- uses vertex attributes:
  - color[x] : white, gray, black
  - d[x] : estimate of $\delta(s,x)$
  - p[x] : predecessor of x along a shortest s-x path.
- uses a FIFO queue of vertices: Q

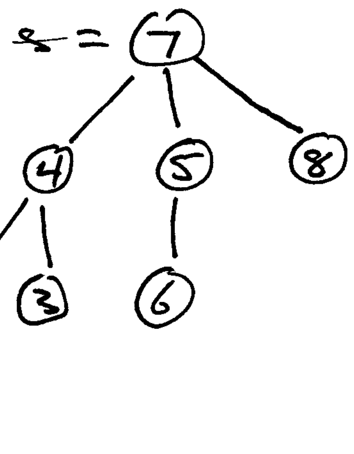
Ex.

	adj	color	d	p
1	2 6	~~w~~ ~~g~~ b	~~∞~~ 3	~~n~~ 2
2	1 3 5	~~w~~ ~~g~~ b	~~∞~~ 2	~~n~~ 4
3	2 4 6	~~w~~ ~~g~~ b	~~∞~~ 2	~~n~~ 4
4	2 3 7 8	~~w~~ ~~g~~ b	~~∞~~ 1	~~n~~ 7
5	2 6 7	~~w~~ ~~g~~ b	~~∞~~ 1	~~n~~ 7
6	1 3 5 8	~~w~~ ~~g~~ b	~~∞~~ 2	~~n~~ 5
7	4 5 8	~~g~~ b	0	n
8	4 6 7	~~w~~ ~~g~~ b	~~∞~~ 1	~~n~~ 7

let s = 7

Q : ~~7~~ ~~4~~ ~~5~~ 8 ~~2~~ ~~3~~ ~~6~~ ~~1~~

BFS Tree:



dist = depth

0

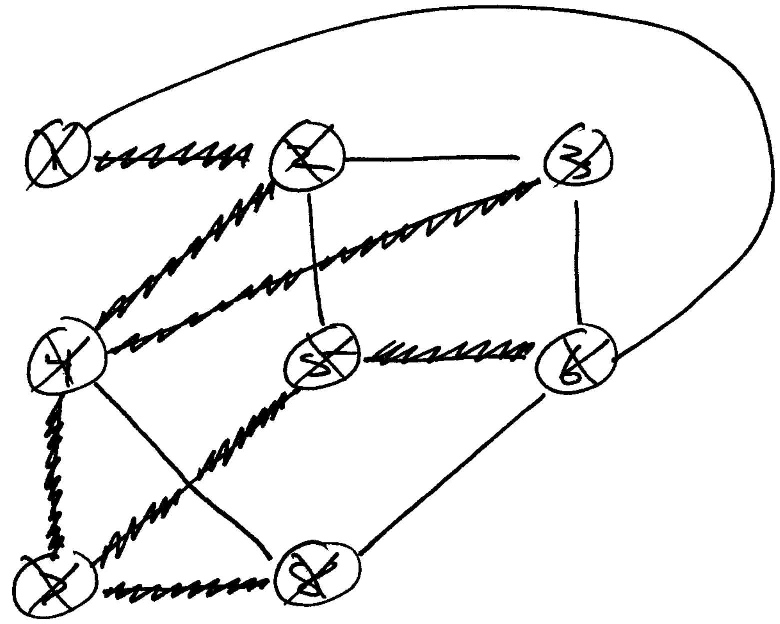
1

2

3

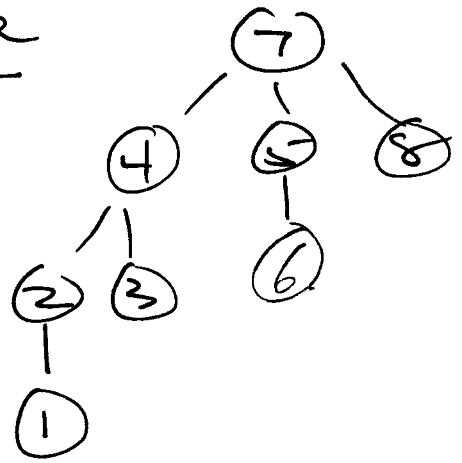
Visual aproach:

s =



$Q: \underbrace{\cancel{7}}_{0}\ \underbrace{\cancel{4}\ \cancel{5}\ \cancel{8}}_{1}\ \underbrace{\cancel{2}\ \cancel{3}\ \cancel{6}}_{2}\ \underbrace{\cancel{1}}_{3}$

Tree



Predecessor subgraph.

Defn

H is a subgraph of G iff

- H is a graph
- $V(H) \subseteq V(G)$
- $E(H) \subseteq E(G)$

Predecessor subgraph of G

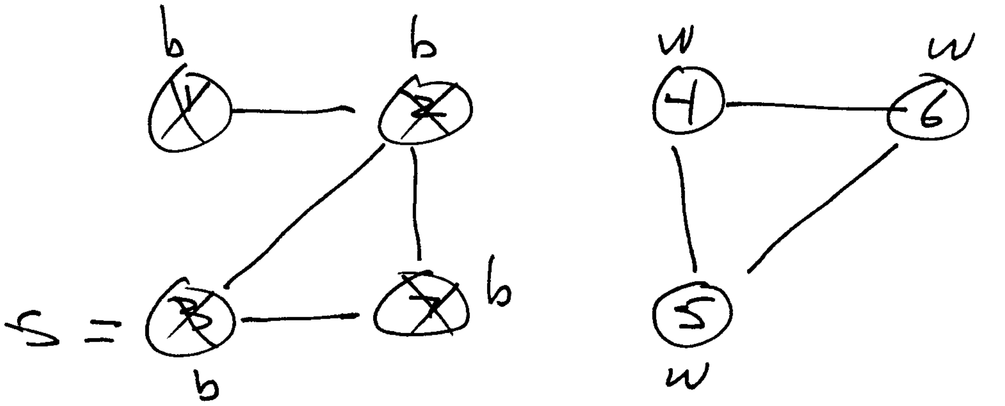
(V_p, E_p) where

$$V_p = \{ x \in V(G) \mid p[x] \neq nil \} \cup \{s\}$$

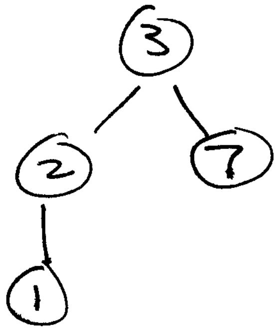
$$E_p = \{ \{x, y\} \mid p[y] = x \}$$

observe: $|E_p| = |V_p| - 1$.

Ex. disconnected graph



Q : $\cancel{3}$ $\cancel{2}$ $\cancel{7}$ $\cancel{1}$



dist.

0

1

2

Directed graphs

$G = (V, E)$

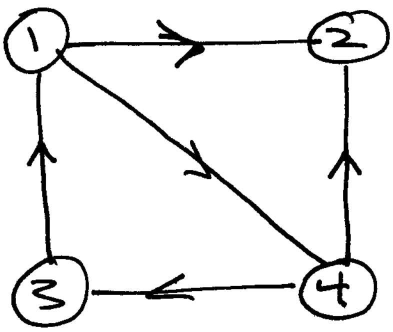
$V \neq \phi$

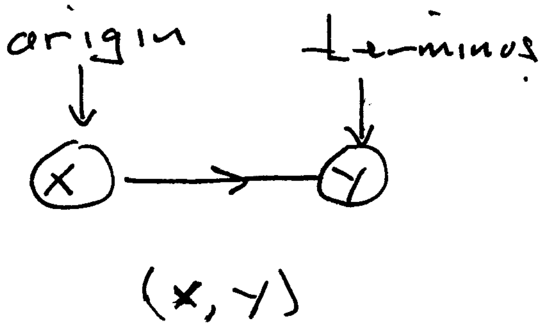
$E \subseteq V \times V$

Define directed

- walks
- trails
- paths

Ex.





$V = \{1, 2, 3, 4\}$

$E = \{(1,2), (1,4), (3,1), (4,2), (4,3)\}$

adj list Representation

1 : 2 4 ← list of terminii of edges having 1 as origin

2 :

3 : 1

4 : 2 3

	adj	color	d	p
1	2 4			
2				
3	1			
4	2 3			

⋮

Quiz Practice Problems

```
void swap(List L, int i, int j){
    int k, x, y, temp;
    if(i>j){temp=i; i=j; j=temp}
    moveFront(L);
    for(k=0; k<i; k++) moveNext(L);
    x=get(L);
    for( ; k<j; k++) moveNext(L);
    y=get(L)
    set(L,x);
    for( ; k>i; k--) movePrev(L);
    set(L,y);
}
```

```
void reverse(List L, int a, int b){
    int i, j, temp;
    if(a>b){temp=a; a=b; b=temp;}
    i=a; j=b;
    while(i< j){
        swap(L, i, j)
        i++;
        j--;
    }
}
```