Last assignment:

"Six Degrees of Kevin Bacon"

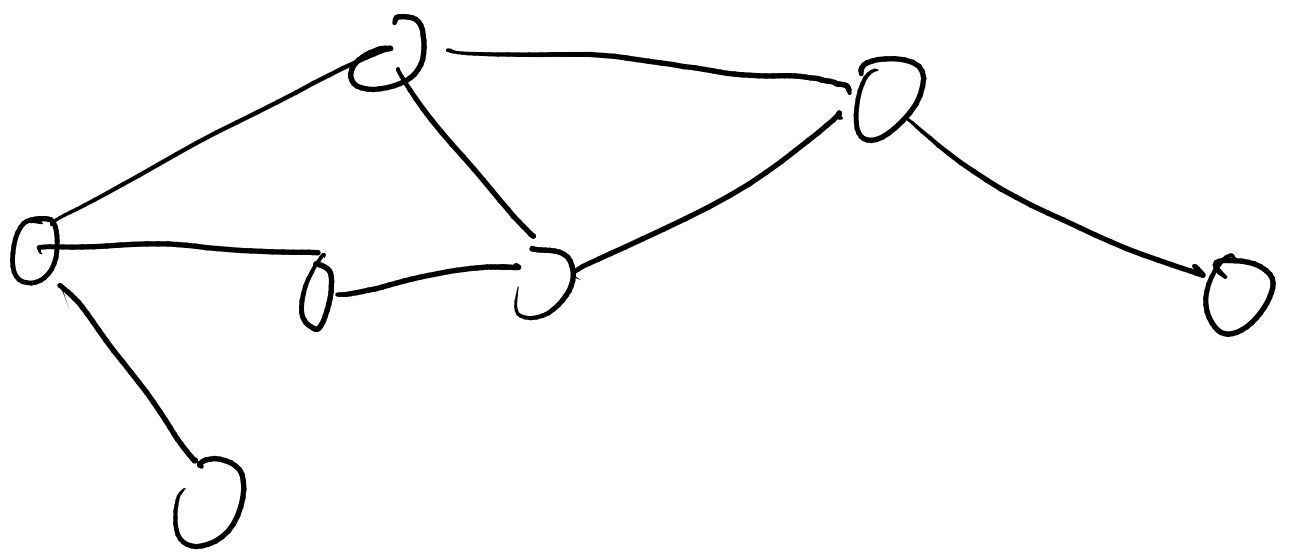
- based on IMDb data

Deadline (one part)

- due Tues (next week)
- late deadline Wed (next week)

Graph

a set of vertices (or nodes) and edges connecting them



Ex: Social networks

vertices: individuals

edges: likes / friends / follows

Wikipedia (or websites in general)

vertices: pages

edges: links

Courses at Carleton

vertices: courses

edges: prerequisites

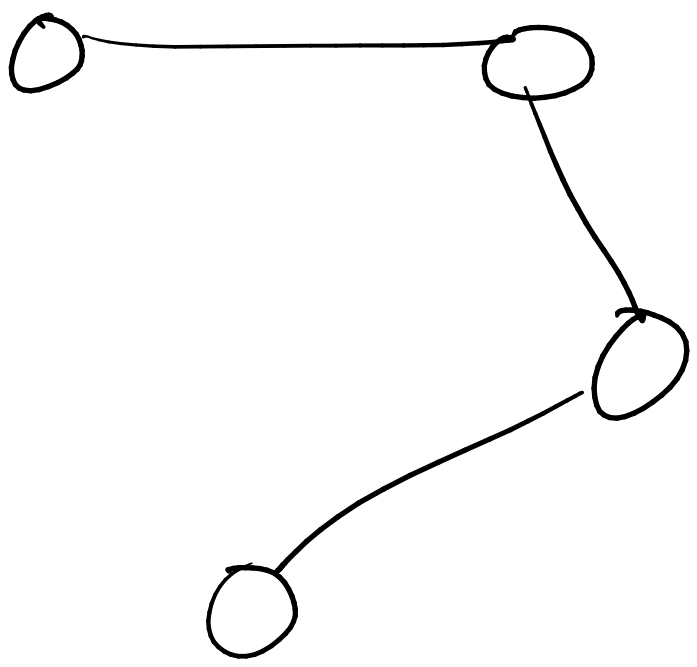
Map (geographic)

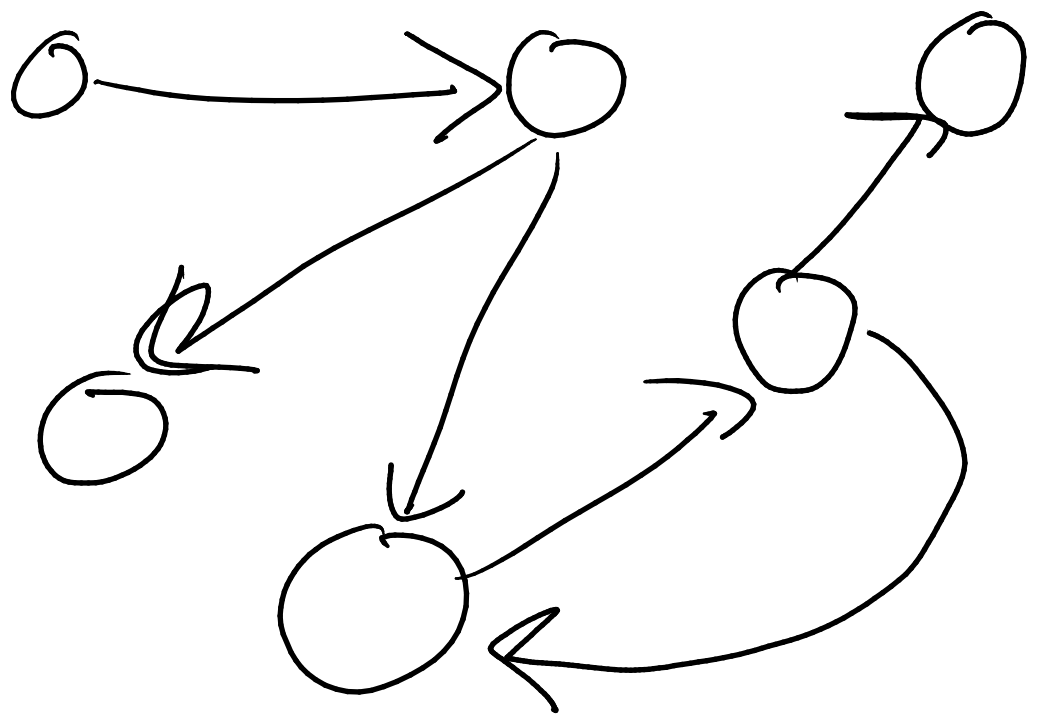
vertices: locations

edges: connections

Undirected graph (edges have no direction)

Directed graph(- - - a - - -)





map (?)

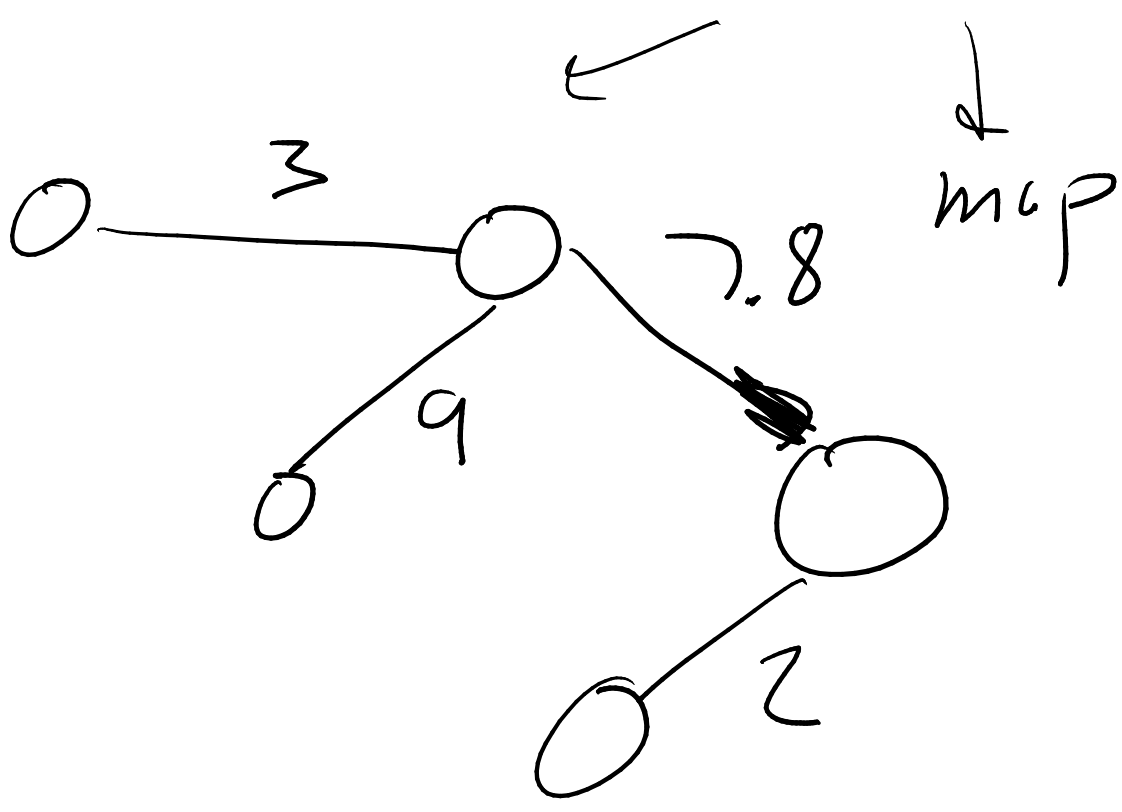
Facebook friends (?)

Instagram mutuals

Courses

Wikipedia

Edges can be weighted or unweighted



map

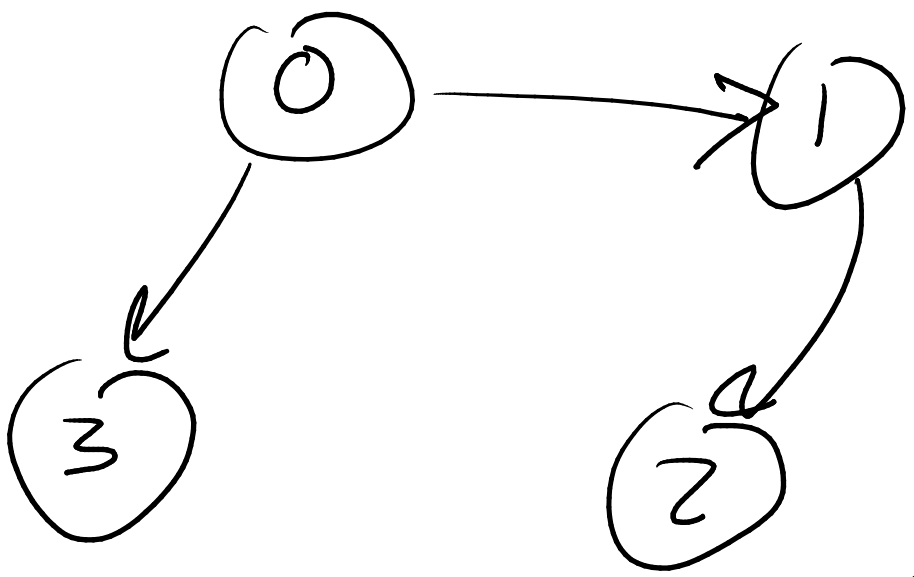
friends

web pages

Graph is another ADT

ADTs	Implem strategies
Stack	List / -----
Map / Dict	
Priority Queue	Heap
Graph	Adjacency matrix, adjacency lists

Adjacency matrix

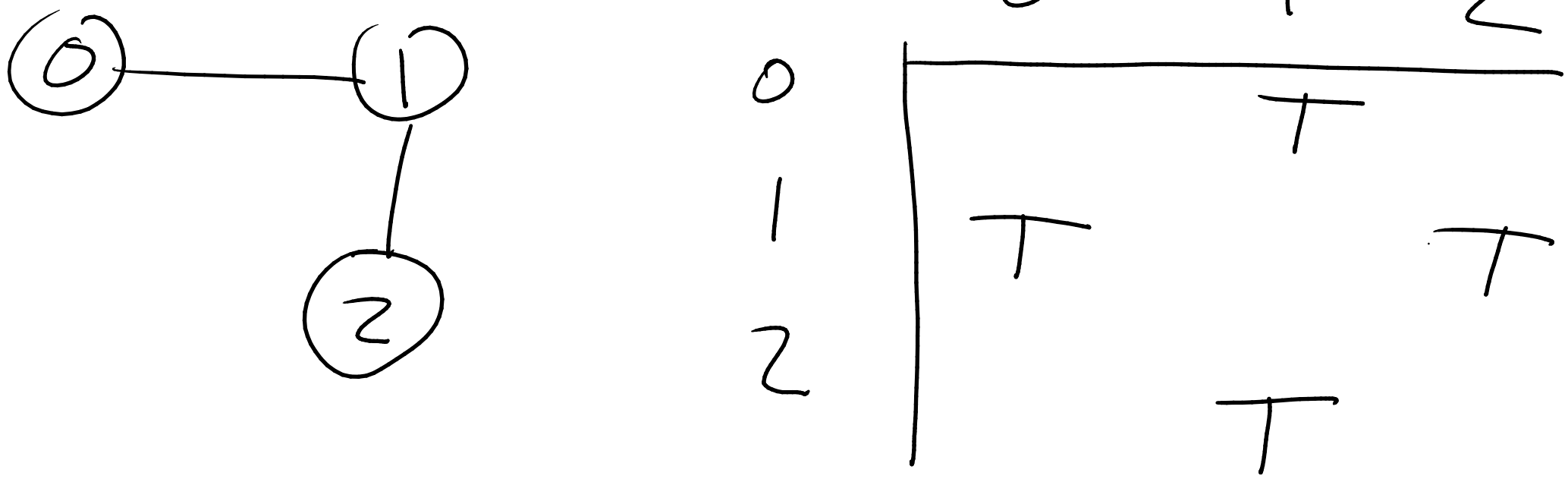


source \ dest	0	1	2	3
0		T		T
1			T	
2				
3				

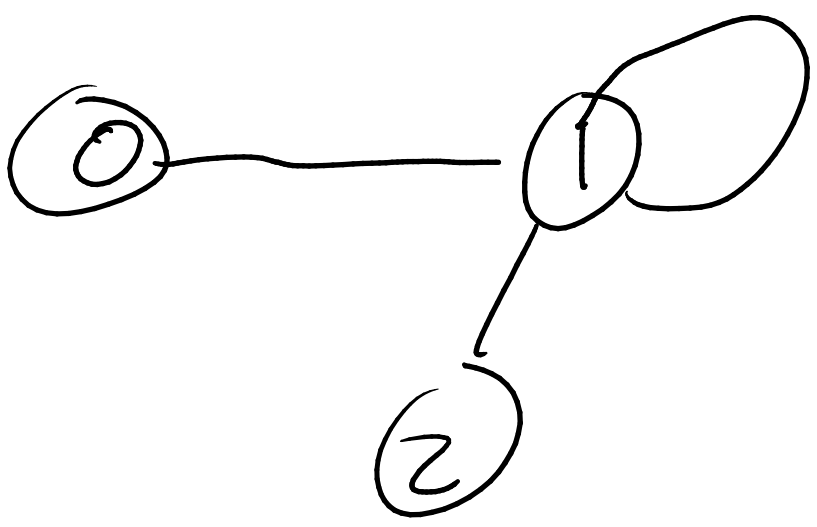
all else F

if weighted, use numeric values
instead of Booleans
(null for no edge)

if undirected, just put both directions



vertices do not connect to themselves
by default



← explicit

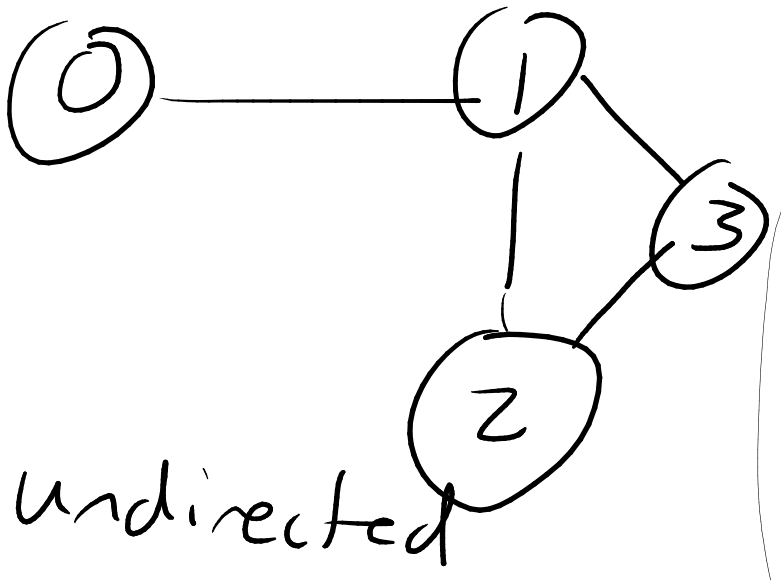
Good / bad about adj matrices:

- fast lookups
- waste of memory if sparse graph ← not many edges
- easy

Alt: adjacency lists

- a list for each vertex of its connections

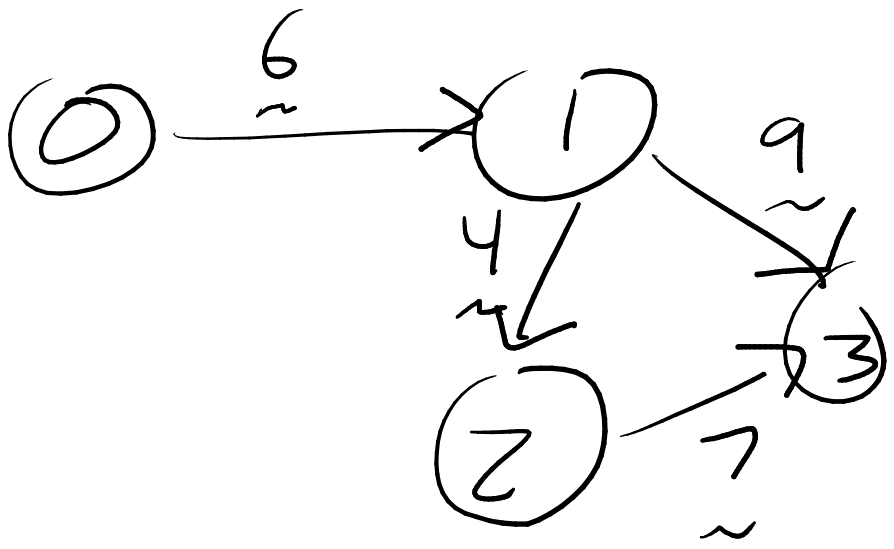
A



vertices	connections
0	1
1	0, 2, 3
2	1, 3
3	2, 1

directed - just store connections that are there

weighted



vertices	Connections (object w/ both pieces)
0	1 (6)
1	2(4), 3(9)
2	3(7)
3	—

Adj list pros/cons:

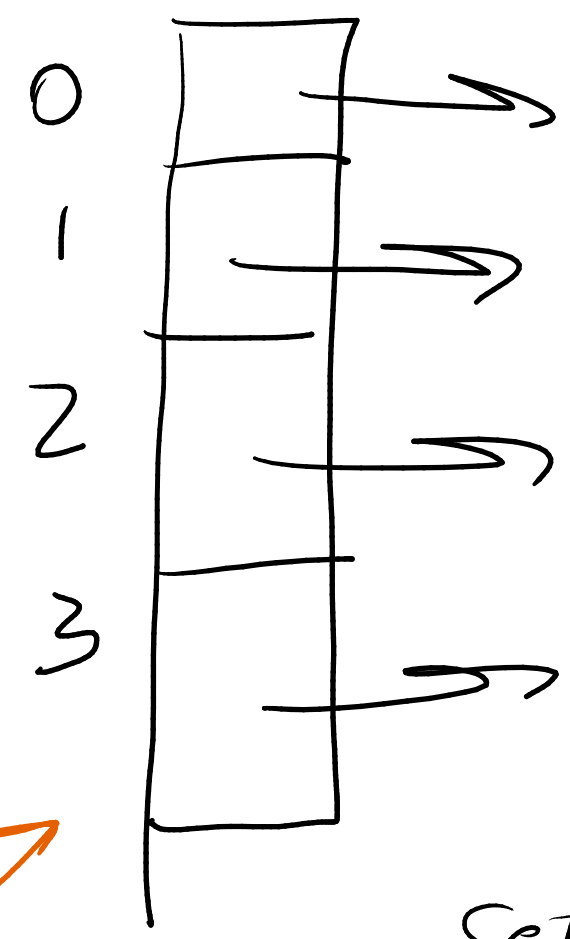
- betta for spase (just storing conections)
  - less memory
- slower lookup (search through list)

More specifically, what kinds of lists are these

-no single answer, one approach

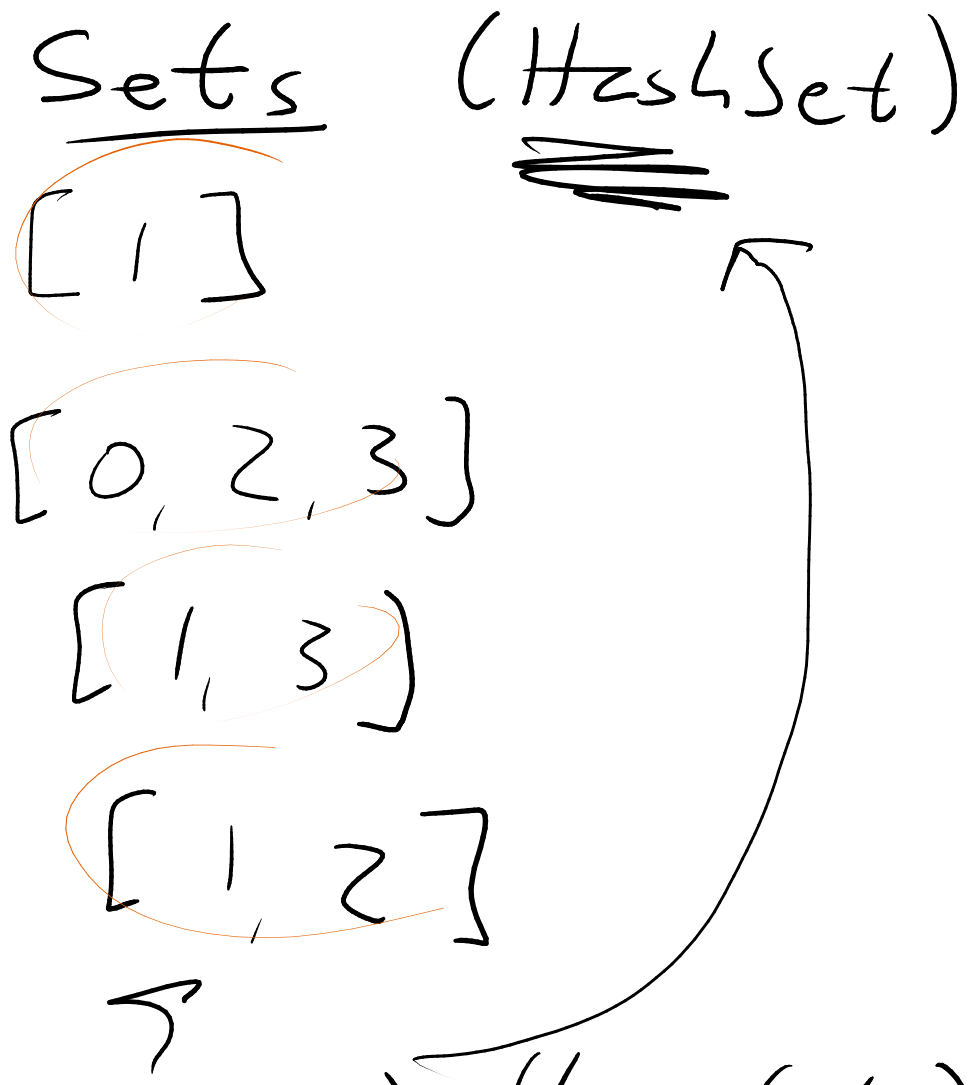
(A)

Kotlin list (array)



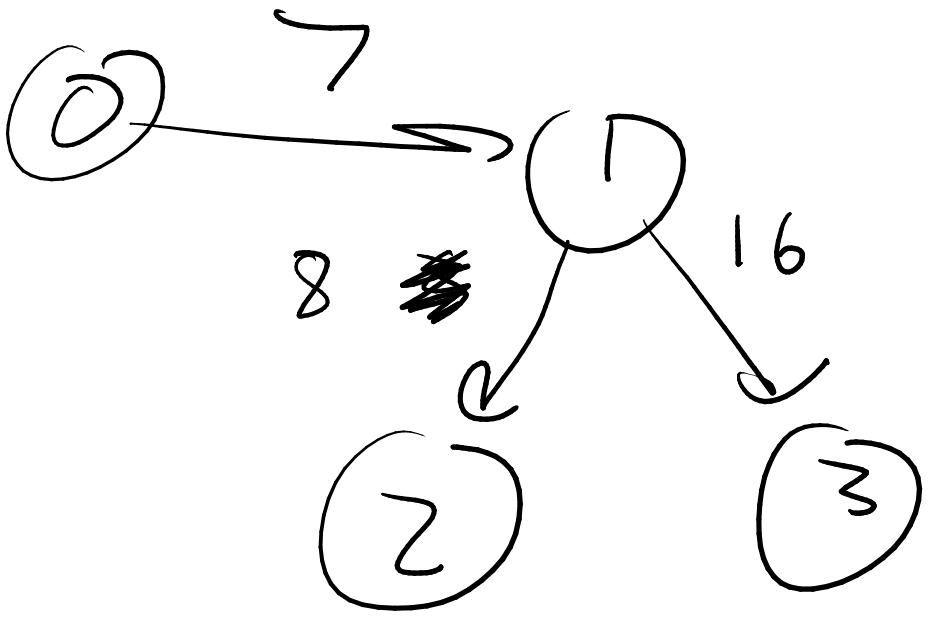
list

Sets (HashSet)



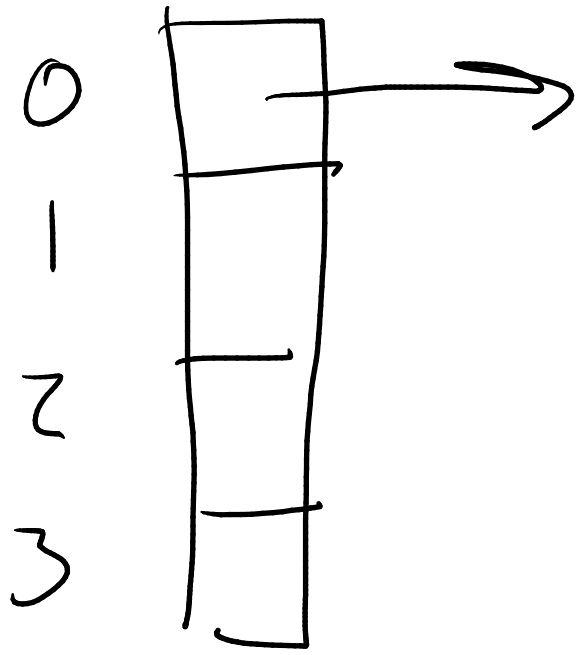
sets, so O(1) (hopefully) to see if something is there or not.

weighted



vertex → weight

Maps (HashMaps)

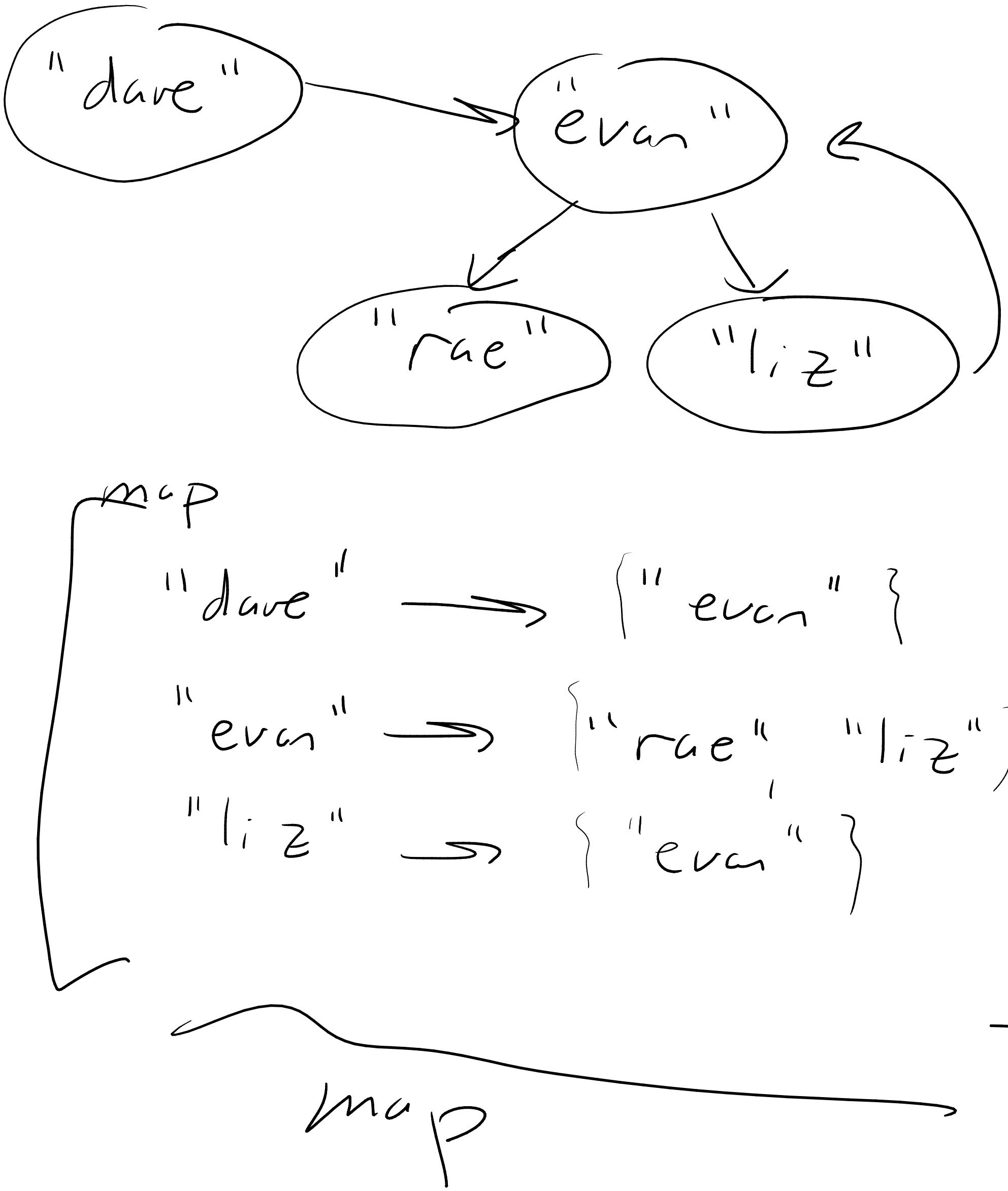


{1 → 7}

{2 → 8, 3 → 16}

list

What if vertices are not integers
(string ids, or otherwise)


"dave"
"evan"
"rae"
"liz"
map
"dave" → {"evan"}
"evan" → {"rae", "liz"}
"liz" → {"evan"}
map

Adj matrix

vs

Adj lists

secret: lists are a
dumb idea here
sets/maps are what is
done in practice