13 Oct 2021

* FRIDAY: Fall Workshop on CG
  → attend at least one talk! (20-min only)
  → check email tonight for how to register
  → no class meeting on Friday for AA.
  → opt: write-up for Misc-credit!

* H-04G will be posted by tomorrow (due: Oct 25)
  → must submit as a project group.
  → Group sizes: 3-4 people (ideally)
    ... ask for exceptions
  → send email to Prof. Fusy once group finalized.
  → 2 options:
    1 → ~~Same~~ research-oriented option: make a video about a "recent" algorithm, must include some +1 component.
    2 → ~~compa~~ implementation option: implement at least one algo + compare to at least 2 other algo for the same problem.

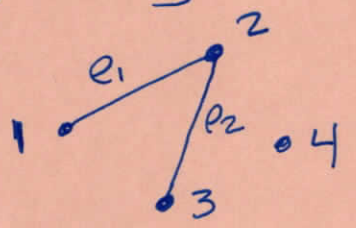
Data Structures for Graphs

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

① Adjacency Matrix

→ rows / columns represent vertices
→ (i,j) entry indicates if $\{i,j\} \in E$

e.g.:



	1	2	3	4
1	0	1	1	0
2	1	0	0	0
3	1	0	0	0
4	0	0	0	0

} easy to make these weighted edges!

Note: $\{1,2\} = \{2,1\}$ since undirected
In general, this matrix is symmetric

Q: What is the space complexity? $\Theta(n^2)$
⇒ if $m \ll n^2$, might not be the best option

But, if $m \sim n^2$, probably ok to use.

Queries: (A) check if $\{i,j\} \in E$ $\Theta(1)$

(B) List edges incident to a given vertex $\Theta(n)$

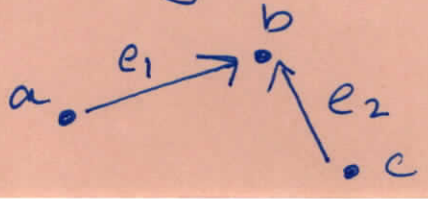
Boundary ~~Incidence~~ Matrix

"the boundary matrix"

→ rows rep. edges
→ cols rep. verts

	1	2	3	4
e_1	1	1	0	0
e_2	0	1	1	0

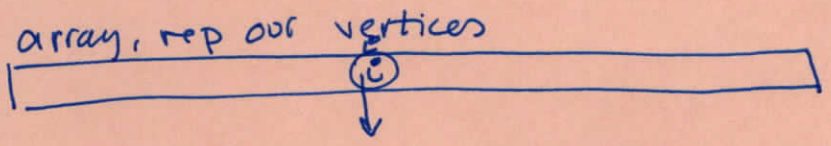
Directed graphs:

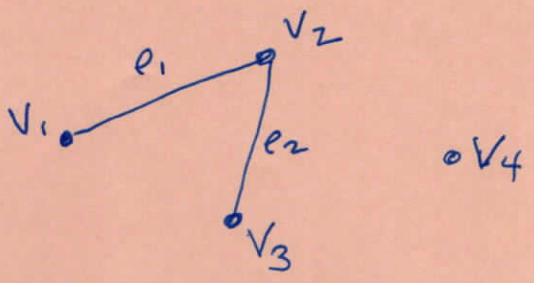


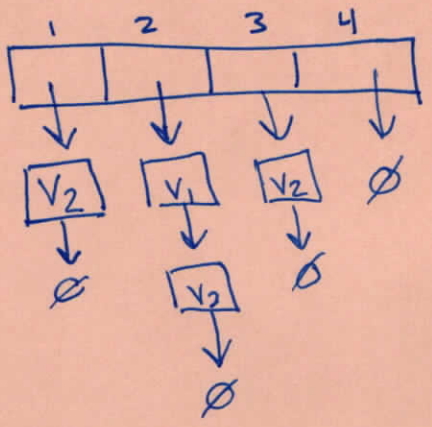
	a	b	c
e_1	-1	1	0
e_2	0	1	-1

space: $\Theta(nm)$
Q-A: $O(m)$
Q-B: $O(m)$

Adjacency Lists ← the data structure used by JE, unless otherwise stated.

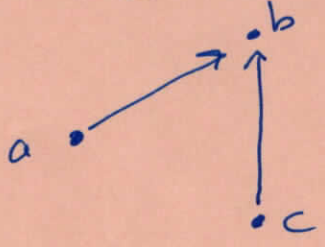


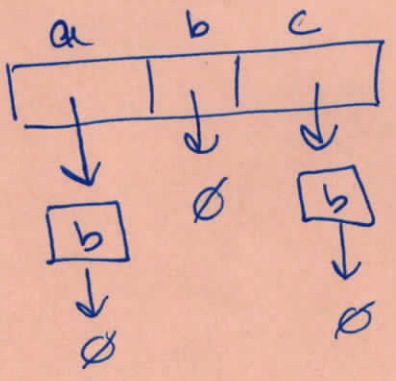




→ note: each edge is stored twice, but that's ok.

If digraph, just list edges leaving the vertex





up here, we have
$n = \Theta(n)$ pointers
+
$2m = \Theta(m)$ elts of list
+
n empty list elts

$2n + 2m = \Theta(n+m)$

Space: $\Theta(n+m)$

note: in K_n, $m = n^2$

Q-A: Is $\{i,j\} \in E$?

Looking at neighbors of i: $O(n)$...
more concisely $\Theta(\deg(v_i)+1)$

Q-B: Edges incident to v_i can be given in $\Theta(\deg(v_i)+1)$

Graphs Data Structures

① Adjacency Lists ← assumption going forward

- dir/undir ex
- $\Theta(V+E)$ space
- search edge: $O(1+\deg(v))$
  - → w/ bal: $O(1+\log(\deg(v)))$
  - → hash: $O(1)$ expected
- find neighbors: $O(1+\deg(v))$

② Adjacency Matrix

- dir = sym. matrix
- query edge $O(1)$
- find neighbors $O(|V|)$
- space: $O(V^2)$

Q1: Is the graph connected?

Q2: flood fill + implicit graph

↳ adj = m. hor or vert edges