COL 106 Lecture 32

Topic : Graph Data Structures

Some interesting computational problems

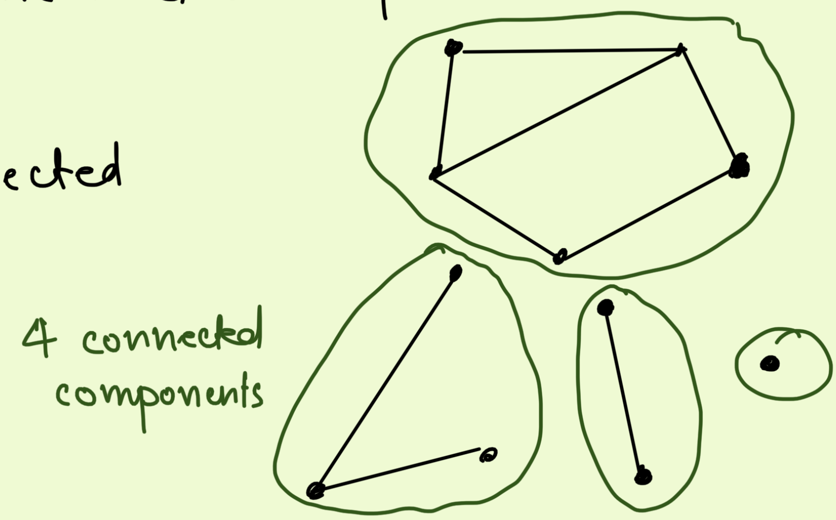
Given a graph G:

1. Reachability: Does there exist a path between u and v?

2. Find a shortest path between u and v.

3. Connectivity: Does there exist a path between every pair of vertices?

4. Identify the connected components of G.



The Graph ADT:

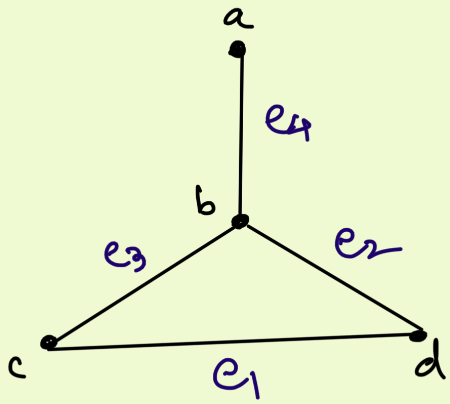
Accessor Methods

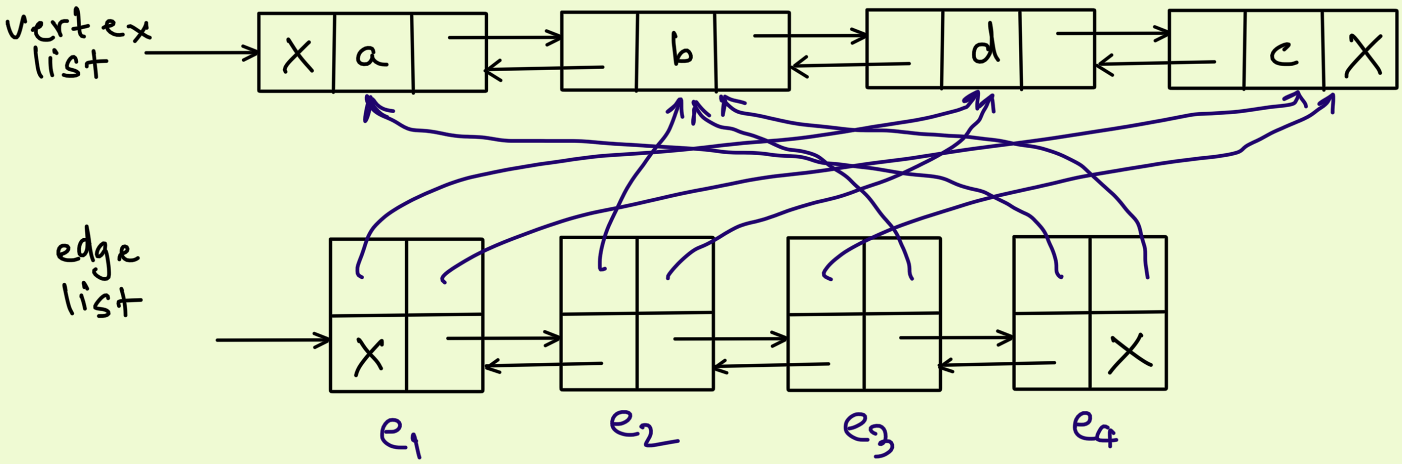
- isAdjacent (u, v)
- listNeighbors (v)

Modifier Methods

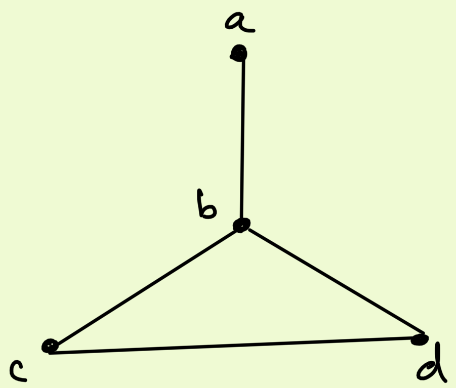
- addVertex (v)
- addEdge (u, v)
- deleteEdge (u, v)
- deleteVertex (v)

Graph Representation 1: Vertex and Edge lists.





Graph Representation 2: Adjacency Matrix

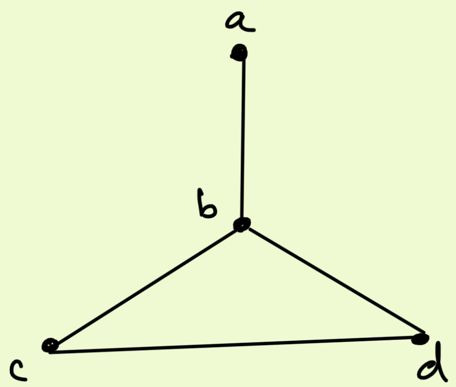


	a	b	c	d
a	0	1	0	0
b	1	0	1	1
c	0	1	0	1
d	0	1	1	0

Question: A: adjacency matrix of a graph.
What do the entries of A^k represent?

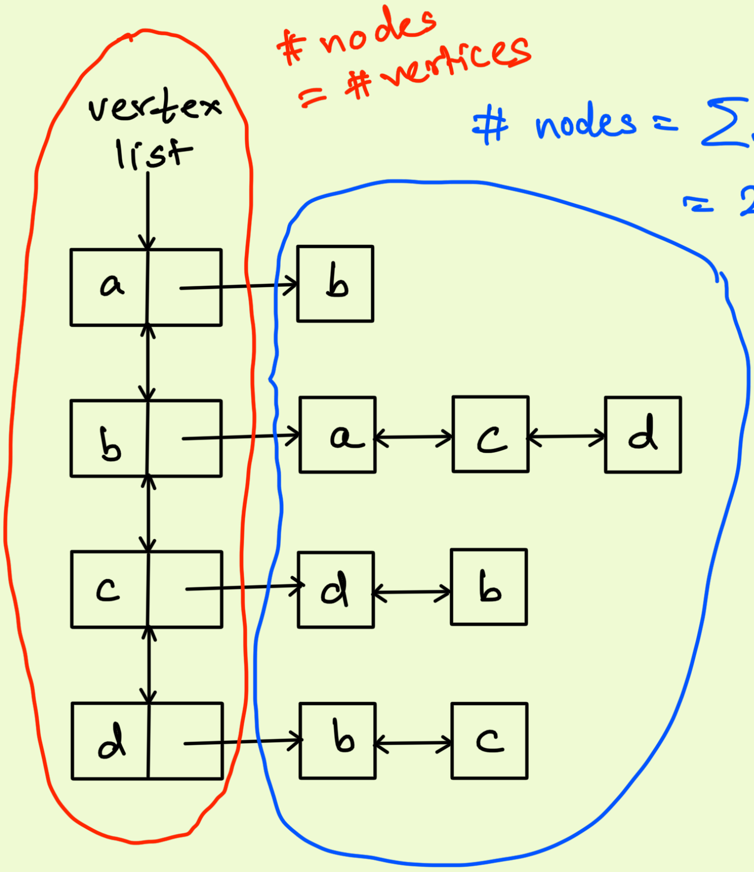
#Walks of length k between two vertices

Graph Representation 3: Adjacency lists



nodes = # vertices

nodes $= \sum_v$ degree of v $= 2 \times$ #edges.



Time / Space Complexity

Let n = # of vertices, m = # edges, d_v = degree of v.

Operation	Vertex and Edge lists	Adjacency Matrix	Adjacency Lists
Space	$n+m$	n^2	$n+m$
isAdj(u,v)	m	1	$\min(d_u, d_v)$ Traverse two lists in parallel
listNbrs(v)	m	n	~~1~~ d_v

Operation	Vertex and Edge lists	Adjacency Matrix	Adjacency Lists
addVertex(v)	1	?	1
addEdge(u,v)	1	1	1. (if u,v edge is guaranteed to not exist already)
delEdge(u,v)	m	1	$d_u + d_v$
delVertex(v)	m	depends!	$\sum_{u \text{ nbr of } v} d_u$
			Can be improved to d_v with a little trick. How?