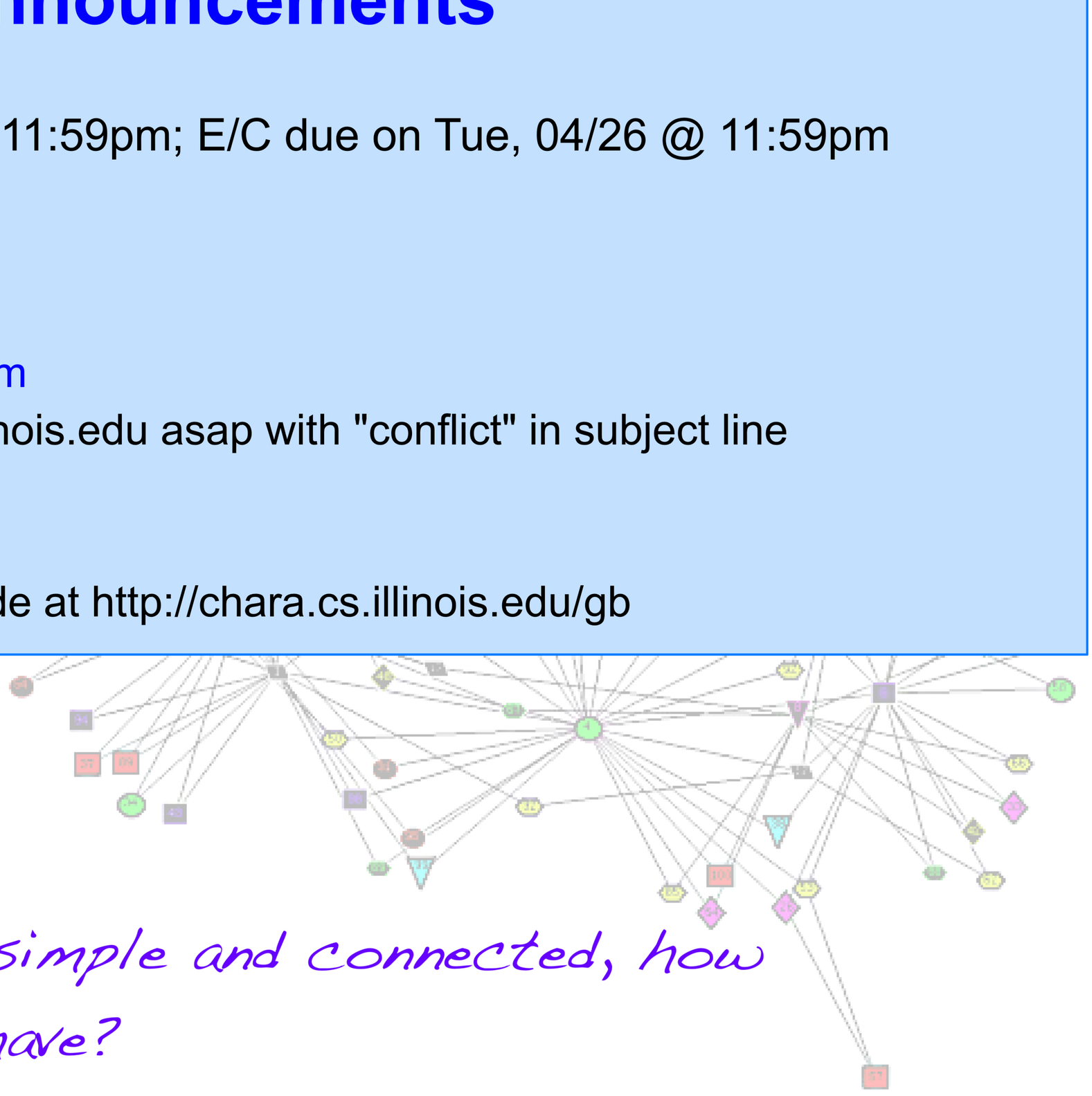
Announcements

* MP7 is out, on Tue, 05/03 @ 11:59pm; E/C due on Tue, 04/26 @ 11:59pm

* Lab_graphs - Sun, 05/01

* **Final Exam:** 05/09 1:30-4:30pm
  Email to Thierry @ ramais@illinois.edu asap with "conflict" in subject line
  Topics: Cumulative + Graphs

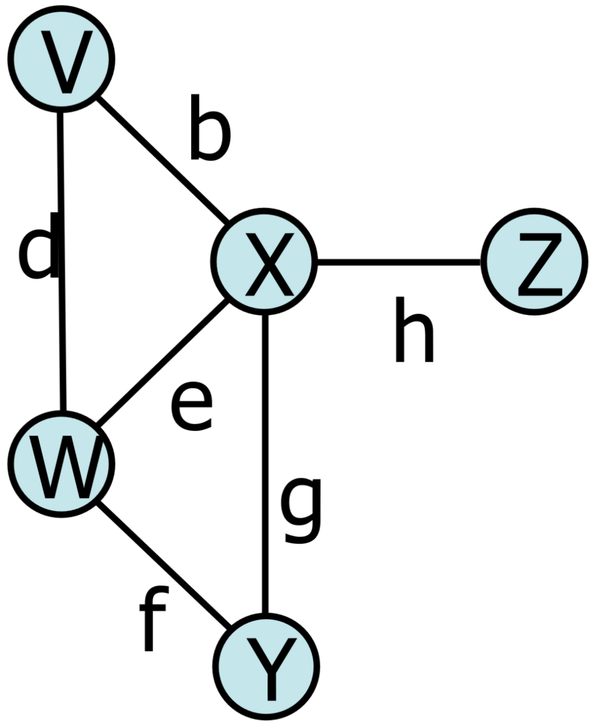
* Please check your current grade at http://chara.cs.illinois.edu/gb



Q: if this graph is simple and connected, how many edges does it have?

A: at least _____, at most _____

Graphs: Toward implementation…(ADT)



Data:

Vertices

Edges

+ some structure that reflects the connectivity of the graph

Functions: (merely a smattering…)

insertVertex(pair keyData)

insertEdge(vertex v1, vertex v2, pair keyData)

removeEdge(edge e);

removeVertex(vertex v);

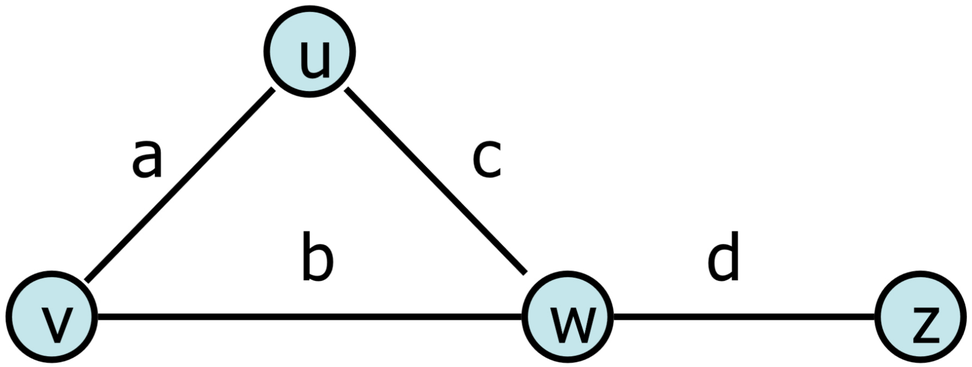
incidentEdges(vertex v);

areAdjacent(vertex v1, vertex v2);

origin(edge e);

destination(edge e);

Graphs: Edge List (a first implementation)



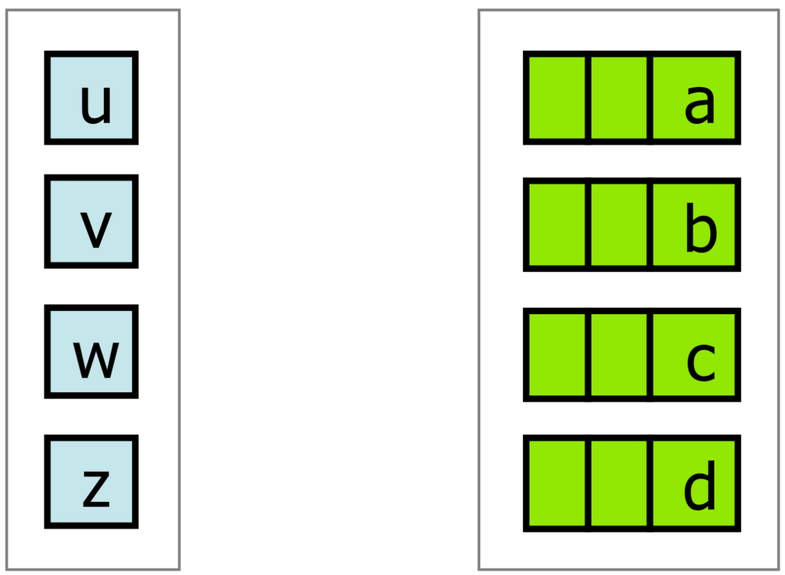
Some functions we'll compare:

insertVertex(vertex v)

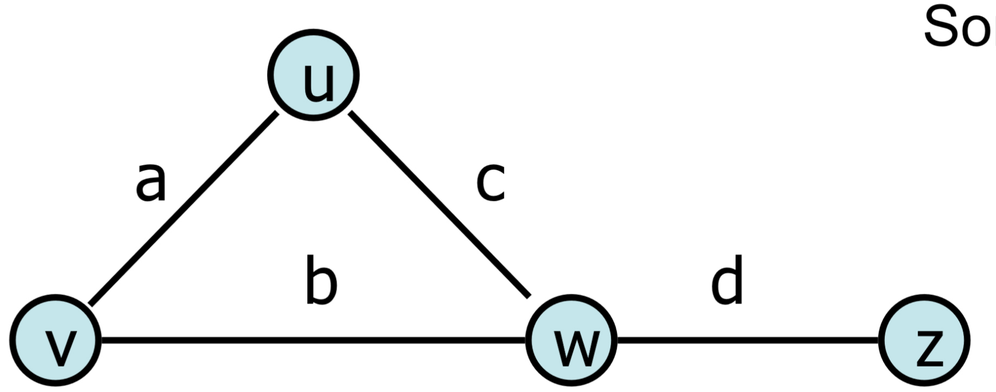
removeVertex(vertex v)

areAdjacent(vertex v, vertex u)

incidentEdges(vertex v)



Graphs: Adjacency Matrix



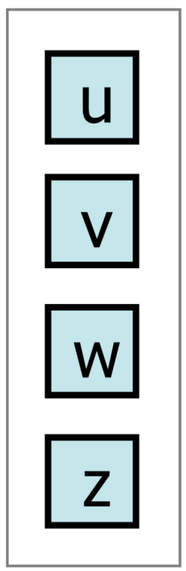
Some functions we'll compare:

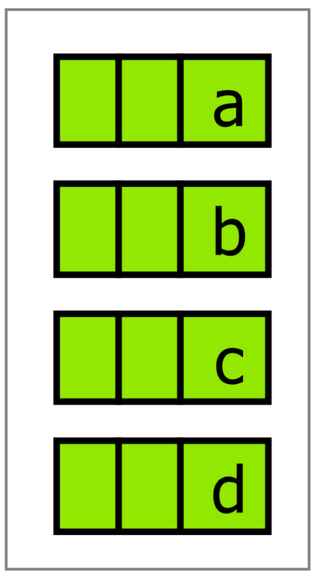
insertVertex(vertex v)

removeVertex(vertex v)

areAdjacent(vertex v, vertex u)

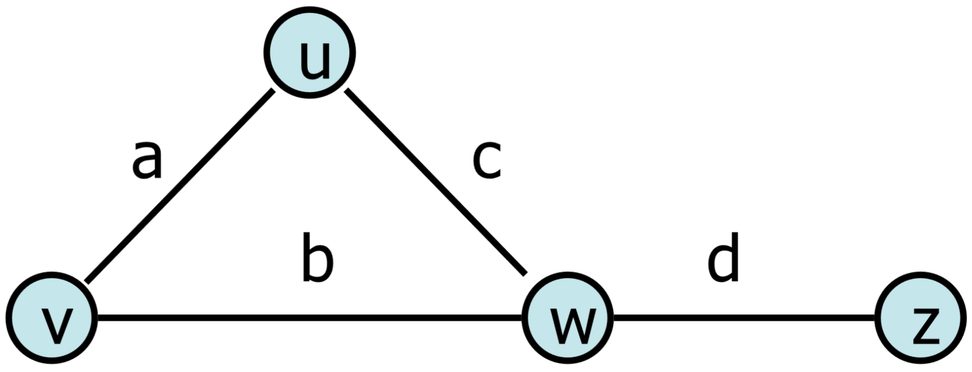
incidentEdges(vertex v)





	u	v	w	z
u				
v				
w				
z				

Graphs: Adjacency List



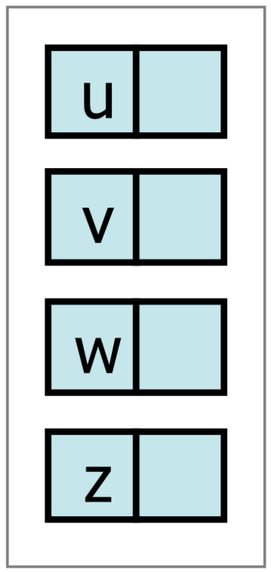
Some functions we'll compare:

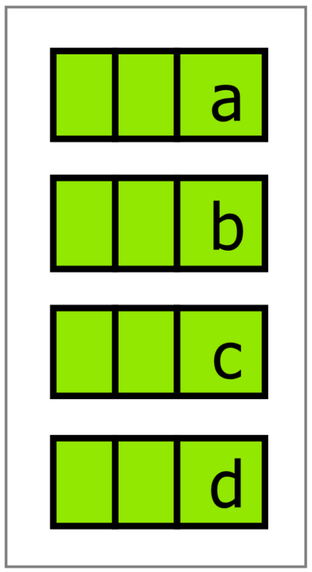
insertVertex(vertex v)

removeVertex(vertex v)

areAdjacent(vertex v, vertex u)

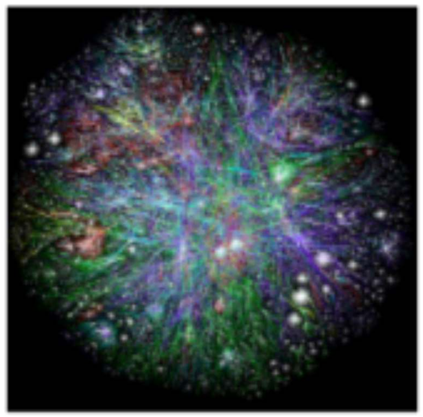
incidentEdges(vertex v)

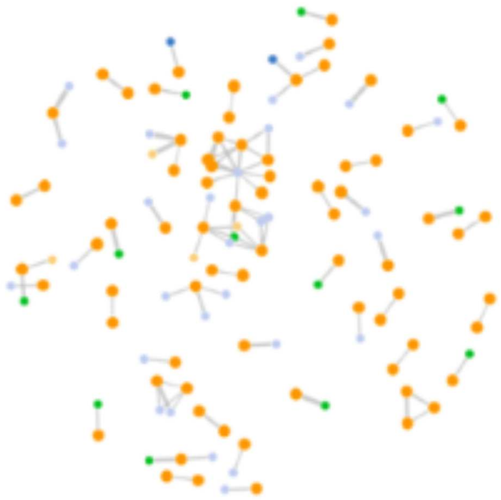




Graphs: Asymptotic Performance

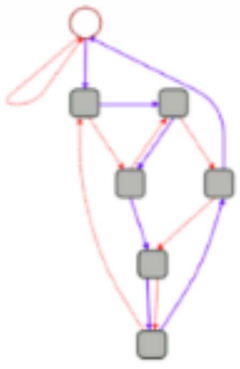
• n vertices, m edges • no parallel edges • no self-loops • Bounds are big-O	Edge List	Adjacency List	Adjacency Matrix
Space	$n + m$	$n + m$	n
incidentEdges(v)	m	$\deg(v)$	n
areAdjacent (v, w)	m	$\min(\deg(v), \deg(w))$	1
insertVertex(o)	1	1	n
insertEdge(v, w, o)	1	1	1
removeVertex(v)	m	$\deg(v)$	n
removeEdge(e)	1	1	1

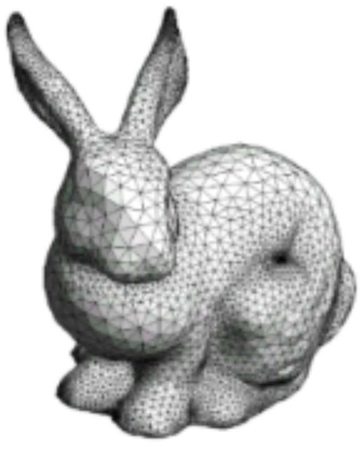


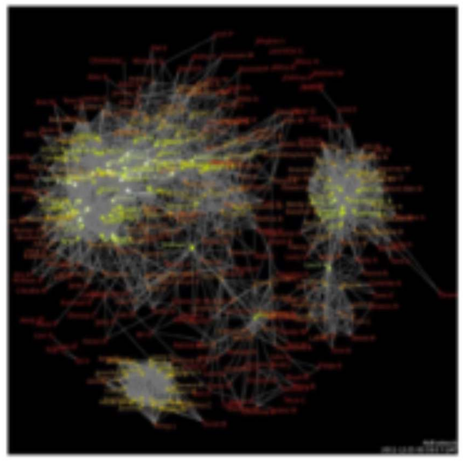


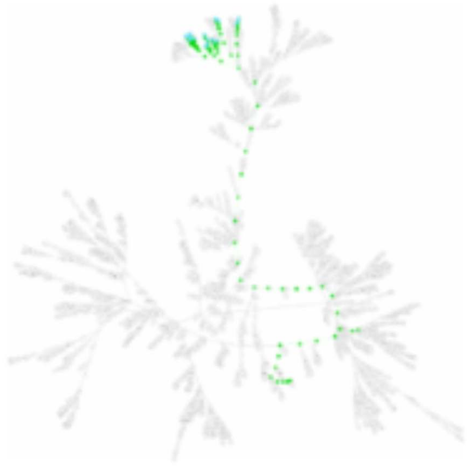
How do we get from here to there?
Need:

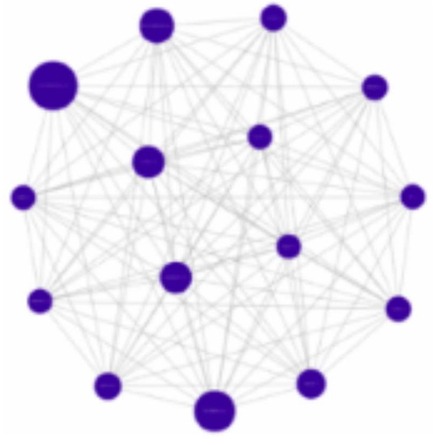
1. Common Vocabulary
2. Graph implementation
3. Traversal
4. Algorithms.











Next we will talk about adjacency matrix.