CS 225

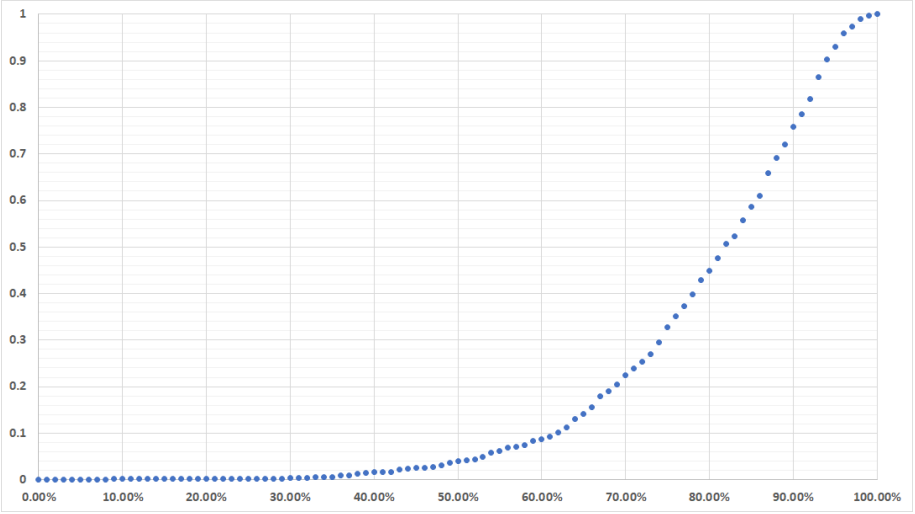
#35: Graph Implementations – Part 2

November 17, 2017 · *Wade Fagen-Ulmschneider*

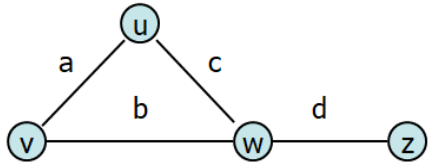
Pre-Fall Break Grade Update

Your CS 225 grade is updated in Compass 2g as "Fall Break Update".

Overall Class CDF:



Graph Implementation #2: Adjacency Matrix



Vert.	Edges	Adj. Matrix

u
v
w
z

		a
		b
		c
		d

	u	v	w	z
u				
v				
w				
z				

Operations on an Adjacency Matrix:

insertVertex(K key):

removeVertex(Vertex v):

areAdjacent(Vertex v1, Vertex v2):

incidentEdges(Vertex v):

Graph Implementation #2: Adjacency List

Vertex List	Edges

u
v
w
z

		a
		b
		c
		d

Operations on an Adjacency List:

insertVertex(K key):

removeVertex(Vertex v):

areAdjacent(Vertex v1, Vertex v2):

incidentEdges(Vertex v):

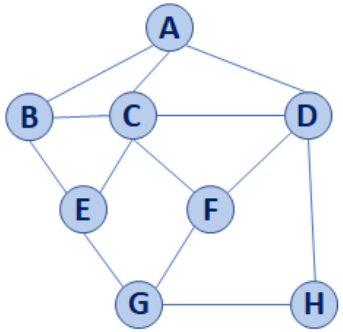
Running Times of Classical Graph Implementations

	Edge List	Adj. Matrix	Adj. List
Space	**n+m**	**n+m**	**n^2**
insertVertex	**1**	**n**	**1**
removeVertex	**m**	**n**	**deg(v)**
insertEdge	**1**	**1**	**1**
removeEdge	**1**	**1**	**1**
incidentEdges	**m**	**n**	**deg(v)**
areAdjacent	**m**	**1**	**min(deg(v), deg(w))**

How do the algorithms compare?

...is one clearly better?

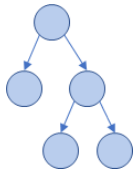
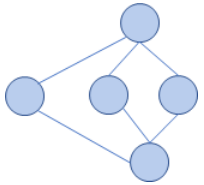
BST Graph Traversal



Graph Traversal

Objective: Visit every vertex and every edge in the graph.
Purpose: Search for interesting sub-structures in the graph.

We've seen traversal before – this is different:

BST	Graph
	

CS 225 – Things To Be Doing:

1. Exam #11 (theory) starts Monday after break
2. MP6 due tonight
3. MP7 released today (due last week of classes, +14 EC possible)
4. lab_dict ongoing; due Wed. Nov. 29 @ 7pm