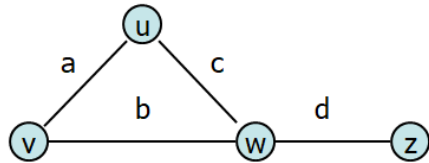
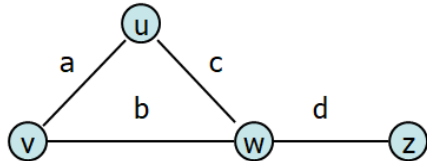


Graph Implementation Review #1: Edge List

Vert.	Edges		
u			a
v			b
w			c
z			d



Graph Implementation Review #2: Adjacency Matrix



Vert.	Edges			Adj. Matrix			
u			a				
v			b				
w			c				
z			d				

Graph Implementation Review #3: Adjacency List

Vertex List	Edges		
u			a
v			b
w			c
z			d

Running Times of Classical Graph Implementations

	Edge List	Adj. Matrix	Adj. List
Space	n+m	n+m	n²
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))

Implementations and Use Cases

Ex. 1:

Ex. 2:

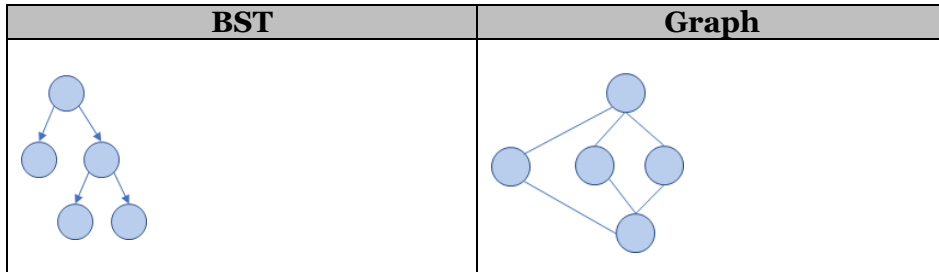
Ex. 3:

Graph Traversal

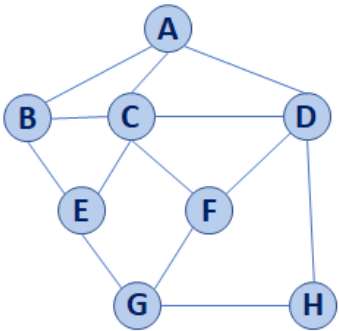
Objective: Visit every vertex and every edge in the graph.

Purpose: Search for interesting sub-structures in the graph.

We've seen traversal before – this is different:



BST Graph Traversal



A	
B	
C	
D	
E	
F	
G	
H	

Pseudocode for BFS

```

1  BFS(G):
2  Input: Graph, G
3  Output: A labeling of the edges on
4         G as discovery and cross edges
5
6  foreach (Vertex v : G.vertices()):
7      setLabel(v, UNEXPLORED)
8  foreach (Edge e : G.edges()):
9      setLabel(e, UNEXPLORED)
10 foreach (Vertex v : G.vertices()):
11     if getLabel(v) == UNEXPLORED:
12         BFS(G, v)
13
14 BFS(G, v):
15     Queue q
16     setLabel(v, VISITED)
17     q.enqueue(v)
18
19     while !q.empty():
20         v = q.dequeue()
21         foreach (Vertex w : G.adjacent(v)):
22             if getLabel(w) == UNEXPLORED:
23                 setLabel(v, w, DISCOVERY)
24                 setLabel(w, VISITED)
25                 q.enqueue(w)
26             elseif getLabel(v, w) == UNEXPLORED:
27                 setLabel(v, w, CROSS)
    
```

CS 225 – Things To Be Doing:

1. Exam #11 (theory) starts today!
2. MP7 released (due last week of classes, +14 EC possible)
3. lab_dictionary due Wednesday at 7:00pm
4. lab_graphs starts Wednesday
5. Multi-day “puzzle” POTDs available M/W/F