Nate Gibson

CptS 223 Homework #4 - Graphs

Please complete the homework problems on the following page using a separate piece of paper. Note that this is an individual assignment and all work must be your own. Be sure to show your work when appropriate.

1. [13] Define these terms as they relate to graph and graph algorithms: Use mathematical terms where appropriate.

Graph: A set of vertices and edges.

Vertice: A point which can be connected to other vertices via edges.

Edge: A connection between two vertices. Can be directed and/or weighted

Undirected Graph: A graph where edges are bi-directional.

Directed Graph: A graph where edges are directed.

Path: A sequence of vertices.

Loop: An edge that connects a vertex to itself.

Cycle: A directed graph of distinct edges, length ≥ 1, such that $W_2 = W_n$.

Acyclic: A graph which contains no cycles.

Connected: A graph such that there's a path from every vertex to every other vertex

Sparse: $|E| << |V|$

Weight: A value associated with an edge.

2. [4] Under what circumstances would we want to use an adjacency matrix instead of an adjacency list to store our graph?

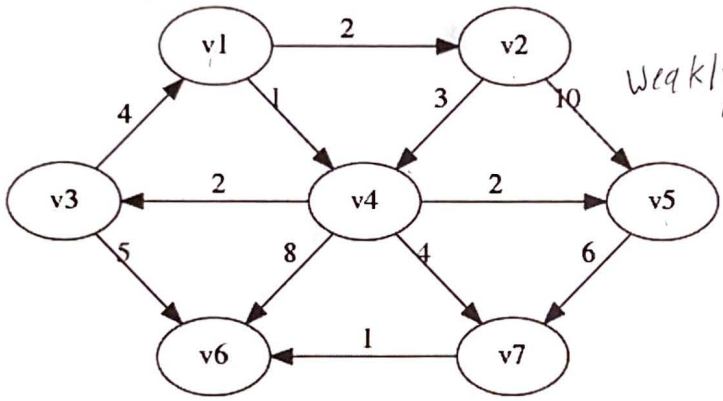
You would want to use an adjacency matrix over an adjacency list for a graph that is not sparse.

3. [6] Name three problems or situations where a graph would be a good data structure to use:

Every situation! But I want credit so...

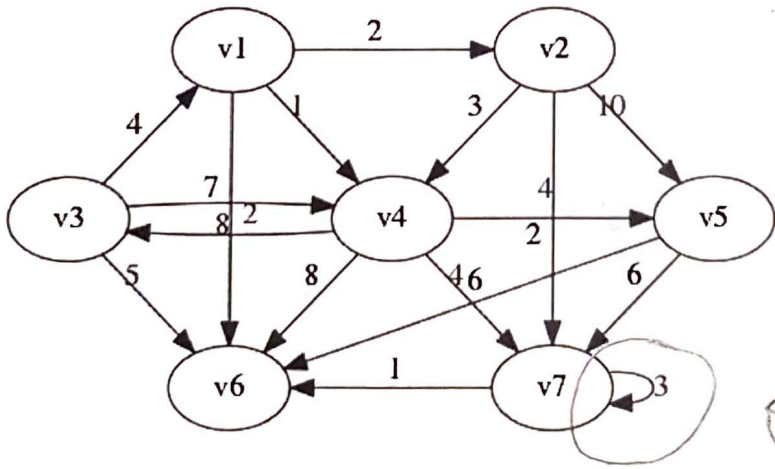
1. GPS navigation
2. Video game NPC movement
3. Scheduling finals without conflicts and minimizing time-slots. (There's an MIT lecture on YT with this example, so I better get credit).

4. [4] What kind of graph is this?



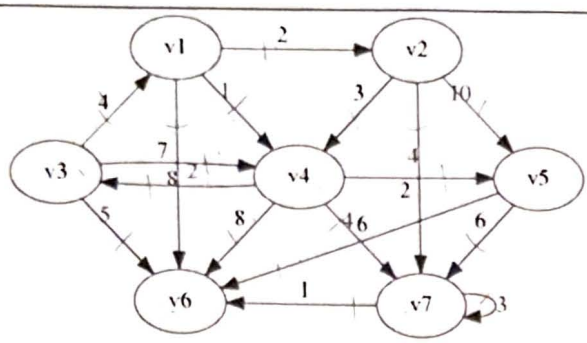
weakly connected, cyclic, Directed graph

5. [4] Identify the loop in this graph:



The Loop

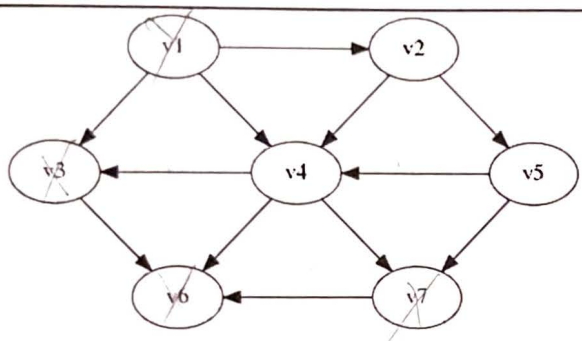
6. **[4]** How many vertices and edges are in this graph:



Vertices 7

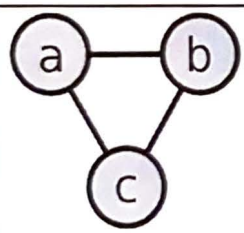
Edges 17

7. **[6]** Are these cyclic or acyclic graphs?



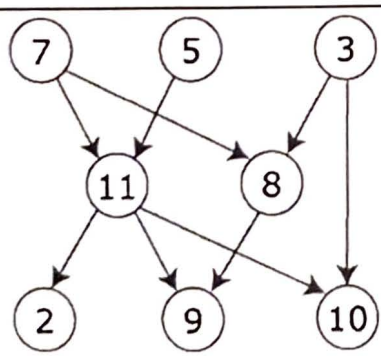
Cyclic?

Yes (No)



Cyclic?

(Yes) No



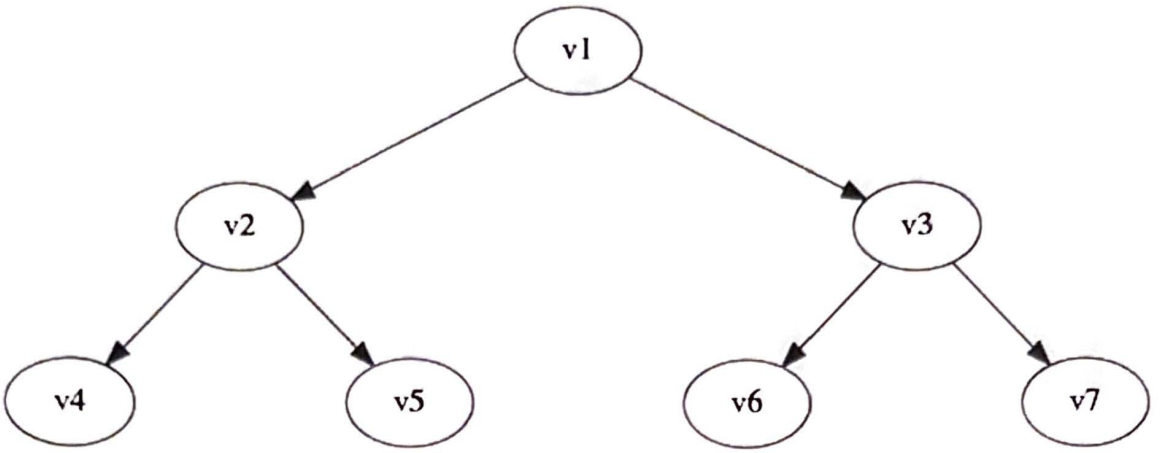
Cyclic?

Yes (No)

8. **[5]** A tree is a particular kind of graph. What kind of graph is that?

(next page)

8.



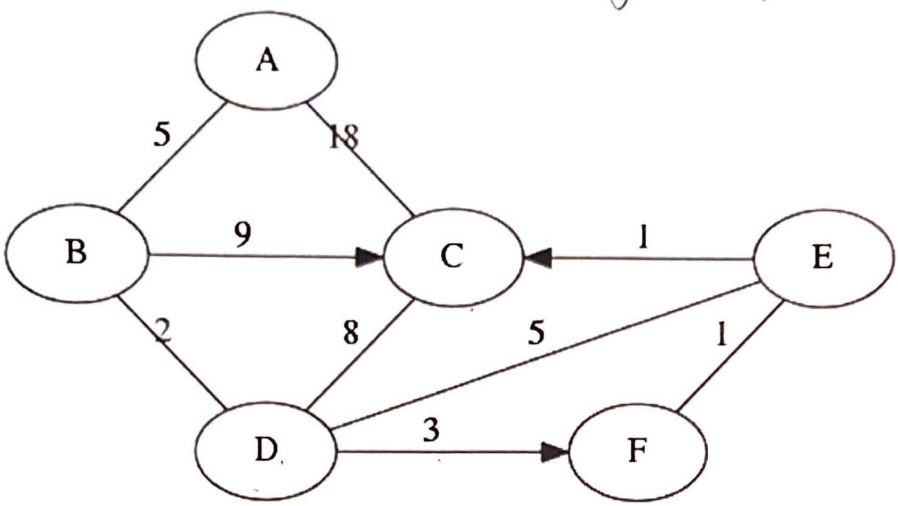
Directed acyclic graph.

9. [4] What is the difference between a breadth-first search and a depth first search?

Breadth-first search moves across a level (height) until it is exhausted, then moves one level down. Depth-first search moves down to the bottom level first, exhausting the lowest level known before it moves back up. Their names are pretty informative.

10. [10] Dijkstra's Algorithm. Use Dijkstra's Algorithm to determine the shortest path starting at **A**. Note that edges without heads are bi-directional. To save time, you do not have to add items to the "priority queue" column after it has been discovered (listed in the "distance" column). Use the table below to show your work.

What's the shortest route (by weight) from A to C?
A → B → D → F → E → C (weight 12)

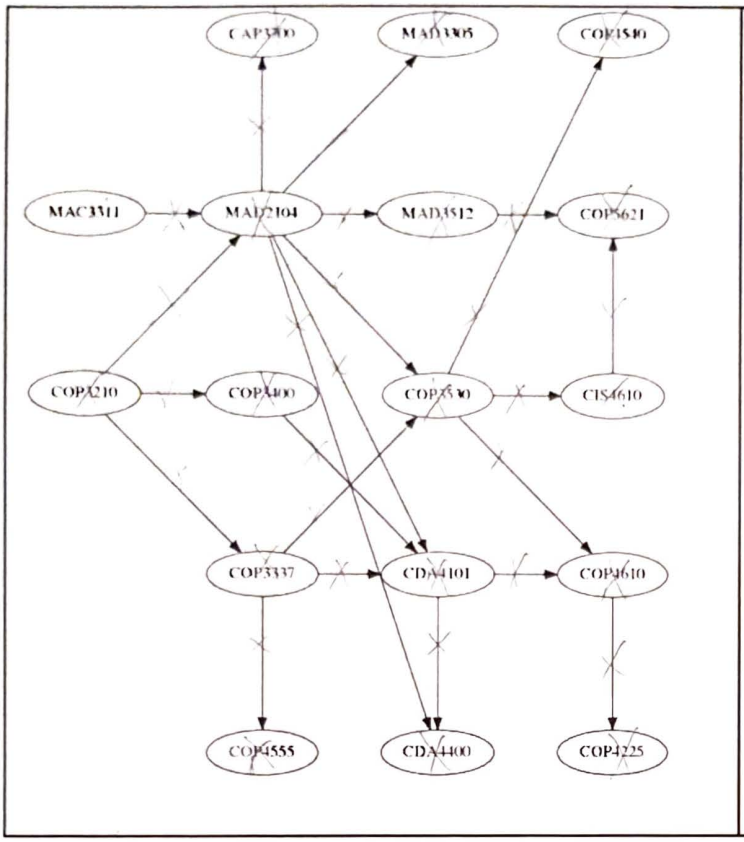


Node: Distance	Priority Queue
	A:0
A:0	B:5 C:18
B:5	D:7 C:14 C:18
D:7	F:10 E:12 C:14 C:15 C:18
F:10	E:11 E:12 C:14 C:15 C:18
E:11	C:12 C:14 C:15 C:18
C:12	C:14 C:15 C:18
	Done

11. [10] Topo sort. Show the final output of running Topo Sort on this graph:

(next page)

11.



What's the vertice with the largest degree and its value?

MAD 2104

8

What's the vertice with the highest indegree and its value?

CDA 4101

3

What's the vertice with the highest outdegree and its value?

MAD 2104

6

Topo sort output:

MAC 3311, Cop 3210, MAD 2104, CAP 3700, MAD 3305, Cop 3400, Cop 3337, Cop 3530, CDA 4101, Cop 4555, CDA 4400, MAD 3512, Cop 4540, CIS 4610, Cop 5621, Cop 4610, Cop 4225