# 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.

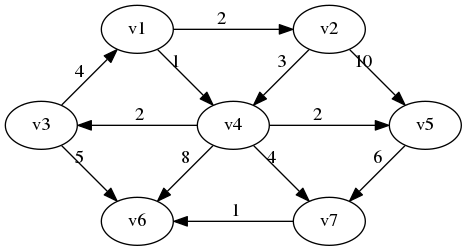
Dillon Ching

4-24-17

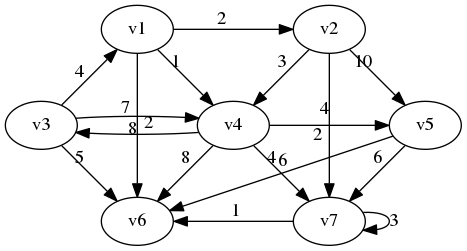
**1. [13]** Define these terms as they relate to graph and graph algorithms:  
 Use mathematical terms where appropriate.

Graph: A pictorial representation of a set of objects where the objects are vertices that are connected by edges  
Vertice: An object represented in a graph  
Edge: A link between two objects represented in a graph  
Undirected Graph: A graph where all the edges are bidirectional   
Directed Graph: A graph where all the edges are directed instead of bidirectional  
Path: A route of vertices and edges that exist between a starting vertice and an ending vertice  
Loop: An edge that connects the vertice to itself  
Cycle: A path that exists where the starting and ending vertice are the same  
Acyclic: A graph that has no graph cycles  
Connected: A graph that has a path between every pair of vertices  
Sparse: A graph that doesn’t have a lot of edges  
Weight: The cost or distance of traveling across one edge

**2. [4]** Under what circumstances would we want to use an adjacency matrix instead of an adjacency list to store our graph?  
  
An adjacency matrix should be used when the graph has about as many edges as vertices. A dense graph is better represented in a matrix due to the amount of connected vertices.   
  
  
  
  
  
  
**3. [6]** Name three problems or situations where a graph would be a good data structure to use:  
  
A graph could be used as a network of computers where the vertices are the computers and the edges are where they are connected. The brain would be a very complex graph where the vertices are the cell bodies and the edges are the axons. A schedule of classes could be a graph where the vertices are the classes and the edges are the routes between them.

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


A directed, dense, cyclic graph

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

V7 points to itself

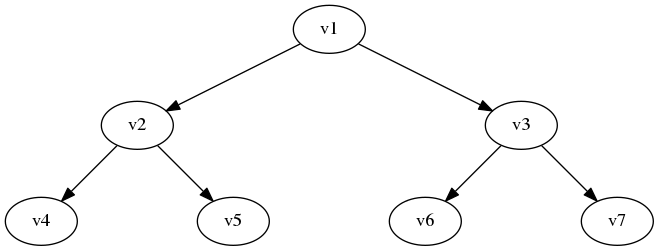
**6. [4]** How many vertices and edges are in this graph:

|  |  |
| --- | --- |
|  | Vertices: 7  Edges: 16 |

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

|  |  |
| --- | --- |
|  | Cyclic?  Yes No |
| Image result for cyclic graph | Cyclic?  Yes No |
| Image result for acyclic graph | Cyclic?  Yes No |

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

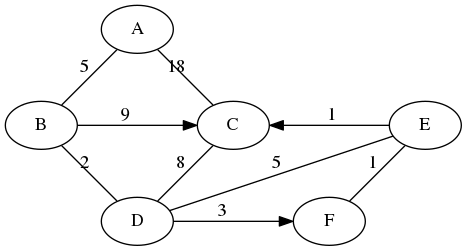


A connected acyclic graph

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

A depth first search algorithm traverses a tree by following a path of nodes all the way down from the root. A breadth first search traverses the tree by visiting neighbor nodes before moving on to the next level.

**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  
  
  


|  |  |
| --- | --- |
| **Node: Distance** | **Priority Queue** |
| **A : 0** | B(5), C(18) |
| **B : 5** | D(7), C(14) |
| **D : 7** | F(10), E(12), C(15) |
| **F : 10** | E(11) |
| **E : 11** | C(12) |
| **C : 12** |  |
|  |  |
|  |  |
|  |  |

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

|  |  |
| --- | --- |
|  | What’s the vertice with the largest degree and its value?  MAD2104 : 8  What’s the vertice with the highest indegree and its value?  CDA4101 : 3  What’s the vertice with the highest outdegree and its value?  MAD2104 : 6 |

Topo sort output: CAP3700, MAD3305, COP4540, COP5621, MAD3512, CIS4610, COP4225, COP4610, COP3530, CDA4400, CDA4101, MAD2104, MAC3311, COP3400, COP4555, COP3337, COP3210