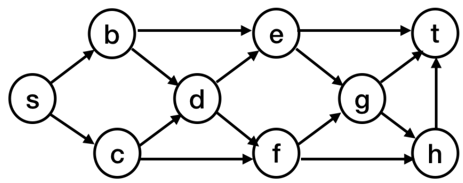
CSCI 200

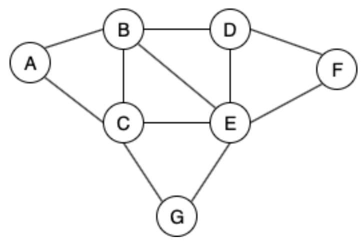
HW #5: Graphs

Directions: *Complete your work on a separate sheet of paper. Submit the physical copy of your work at the beginning of class on the specified due date. Show your work. You may work in groups of up to 3 students provided that all students participate in each question. Provide a short preliminary explanation of how an algorithm works before running an algorithm or presenting a formal algorithm description, and use examples or diagrams if they are needed to make your presentation clear.*

1. List two different topological orders of vertices of the following graph.



2. Consider the graph G depicted below. Show the resulting graph after executing a graph traversal algorithm (by labeling each edge as a discovery, cross, or back edge). Whenever faced with a decision of which vertex to pick from a set of vertices, **pick the vertex whose label occurs earliest in the alphabet.**



   (a) Trace the execution of **BFS** beginning at vertex A. List the vertices in the order in which they are visited.

   (b) Trace the execution of **DFS** beginning at vertex A. List the vertices in the order in which they are visited.

3. A company named RT&T has a network of n switching stations connected by m high-speed communication links. Each customer's phone is directly connected to one station in his or her area. The engineers of RT&T have developed a prototype video-phone system that allows two customers to see eachother during a phone call. In order to have acceptable image quality, however, the number of links used to transmit video signals between the two parties cannot exceed 4. Suppose that RT&T network is represented by a graph. Design and give the **pseudo-code** for an efficient algorithm that computes, for each station, the set of stations it can reach using no more than 4 links. Analyze its running time.

4. Consider the following greedy strategy for finding a shortest path from vertex *start* to vertex *goal* in a given connected graph with positive edge weights. Does it always find a shortest path from *start* to *goal*? Either explain intuitively why it works, or give a counter-example.

   (a) Initialize *path* to *start*, and initialize *visitedVertices* to $\{start\}$

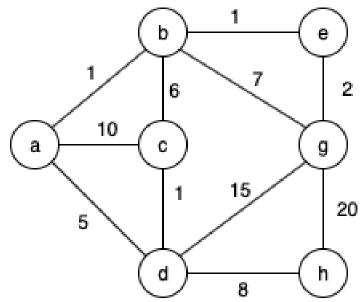
   (b) If $start = goal$, return *path* and exit. Otherwise, continue.

   (c) Find the edge $(start, v)$ of minimum weight such that v is adjacent to *start* and v is not in *visitedVertices*.

   (d) Add v to *path*, and add v to *visitedVertices*.

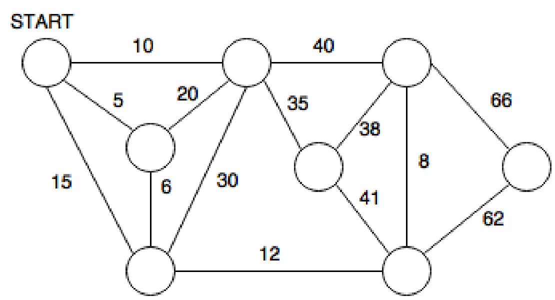
   (e) Set *start* equal to v and go to step (b).

5. Illustrate the execution of Dijkstra's algorithm on the following graph to construct a shortest path tree rooted at vertex a.



6. Give a linear time algorithm to remove all the cycles in an undirected graph $G = (V, E)$. Removing a cycle means removing an edge of the cycle. If there are k cycles in G, the algorithm should only remove at most $O(k)$ edges.

7. Consider the following graph. Use the specified algorithm to find the MST.



   (a) Use Prim's algorithm. At each step, show the vertex and the edge added to the tree and the resulting values of D after the relaxation operation. Use START vertex as the first vertex in your traversal.

   (b) Use Kruskal's algorithm. Give a list of edges in the order in which they are considered, and indicate whether or not they are added to the MST.

8. Suppose you are given a diagram of a telephone network, which is a graph G whose vertices represent switching centers, and whose edges represent communications lines between two centers. The edges are marked by their bandwidth. The bandwidth of a path is the bandwidth of its lowest bandwidth edge. Give the **pseudocode** for an algorithm that, given a diagram and two switching centers a and b, will output the maximum bandwidth of a path between a and b. (Just report the maximum bandwidth; you do not have to give the actual path). Analyze the running time of your algorithm.

9. NASA wants to link n stations spread over the country using communication channels. Each pair of stations has a different bandwidth available, which is known a priori. NASA wants to select $n - 1$ channels (the minimum possible) in such a way that all the stations are linked by the channels and the total bandwidth (defined as the sum of the individual bandwidths of the channels) is maximum. Describe an efficient algorithm for this problem and determine its worst-case time complexity. Consider the weighted graph $G = (V, E)$, where V is the set of stations and E is the set of channels between the stations. Define the weight $w(e)$ of an edge $e \in E$ as the bandwidth of the corresponding channel.

10. Suppose we are given the minimum spanning tree T of a given graph G and a new edge $e = uv$ of weight w that we will add to G. Give an $O(n)$ time algorithm to find the minimum spanning tree of the graph $G + e$.