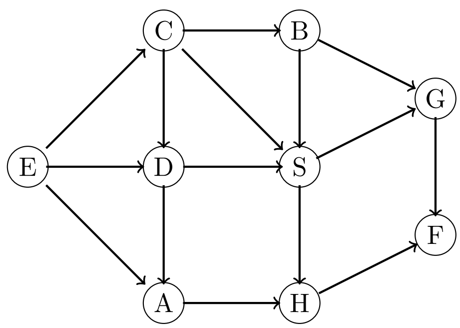
CS 344 Midterm II, Spring 2020
Maximum Points 200 Points

NAME——————————————————-

Show all work and justify your answers. Good Luck!

Problem 1. (35 pts)

(a) (15 pts) Do a DFS starting at C, assuming vertices are to be considered in alphabetical order. Show discovery and finish times near each vertex, separated by a comma. Classify edges as tree edges, back edges, forward edge, cross edge.



(b) (10 pts) Use the DFS in part (a) to do a topological sort on this graph. Draw the resulting graph with vertices on a line, i.e. list the vertices in topologically-sorted order.

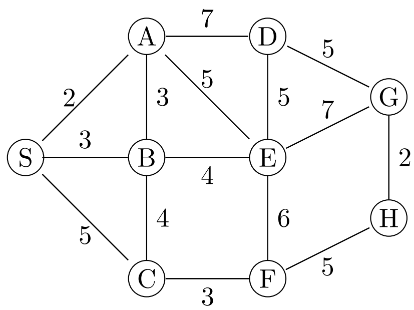
(c) (10 pts) Using inspection, identify the strongly connected components. If we reverse the direction of the edges from D to A and E to D, what are the strongly connected components?

Problem 2. (15 pts) Suppose n people attend a party and some shake hands with others. Prove there are at least two people who have shaken hands with the same number of people.

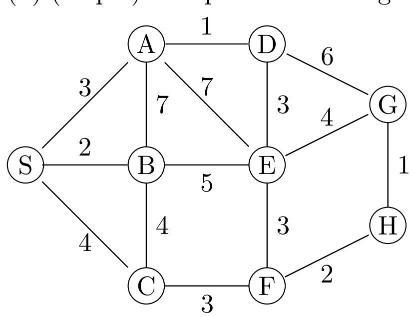
Problem 3. (20 pts) Suppose in the process of computing the MST of a graph with 17 vertices via Kruskal's algorithm we have obtained connected components having sizes 2, 4, 5, 6. Using weighted-union, give all possible scenarios as well the minimum and maximum number of possible operations needed to get an MST. (An operation is changing a link. For example, if the component with 2 elements gets added to the component with 4 elements, we change two links for the smaller set plus linking the head of shorter to the tail of the longer, for a total of 3 operations.)

Problem 4. (50 pts) Show all your work on graphs. The edge weights are given.

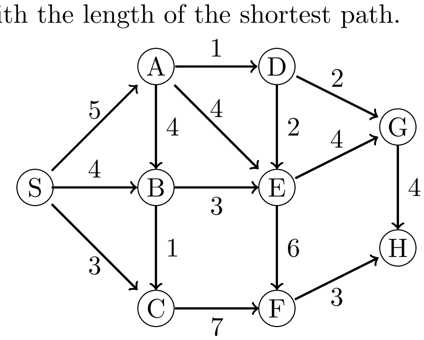
(a) (15 pts) Compute MST using Kruskal's algorithm. Show the order of the selected edges as well as the sets formed. If there is a tie use alphabetical ordering.



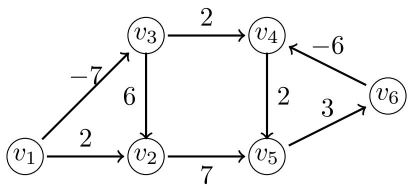
(b) (15 pts) Compute MST using Prim's algorithm. Start at S and give the order of the selected edges.



(c) (20 pts) Using Dijkstra's algorithm find the shortest path from S to all other vertices. Show the complete update history of each vertex v, starting with infinity as the value of $d(v)$, $v \neq S$, and finishing with the length of the shortest path.



Problem 5. (20 pts) Write the system of linear inequalities corresponding to the following graph, then use the approach based on Bellman-Ford to find a solution to the system of inequalities using inspection or prove there is no solution. Do the same when the weight of the edge (v_6, v_4) is changed to -5.



Problem 6. (40 pts)

(a) (10 pts) Assume $G = (V, E)$ is an undirected connected graph consisting of a spanning tree plus one more edge. The edges have weights. Describe an efficient algorithm for computing MST and its complexity in terms of $n = |V|$.

(b) (15 pts) Let $G = (V, E)$ be an undirected complete graph with n vertices. Suppose after we have computed the MST of G we realize that we missed one of the nodes so that G is actually a complete graph with $n + 1$ nodes. Describe an efficient algorithm for computing the MST of the augmented graph.

(c) (15 pts) Suppose $G = (V, E)$, $n = |V|$, $m = |E|$ is an edge weighted graph where the edge weights are distinct (no two edges have the same weight). Prove MST of G is unique.

Problem 7. (20 pts) True-False.

(a) In an undirected graph the sum of the degrees of vertices is always twice the number of edges. (Note: The degree of a vertex is the number of edges that are incident to that vertex.)

(b) We can use BFS to decide if a graph is bipartite. (Note: A graph is bipartite if its vertices can be divided into two disjoint sets so that all edges go from a vertex in one set to a vertex in the other set.)

(c) Given a complete graph on n nodes, the number of cycles with at most 4 vertices is $O(2^n)$.

(d) We can use any minimum spanning tree algorithm to compute the maximum spanning tree, i.e. a spanning tree where the sum of the weights of its edges is maximum.