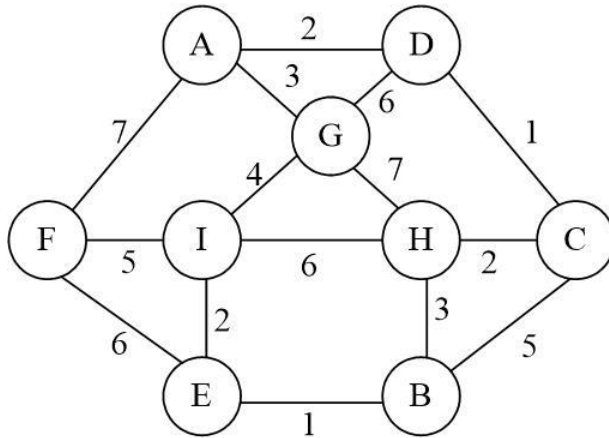
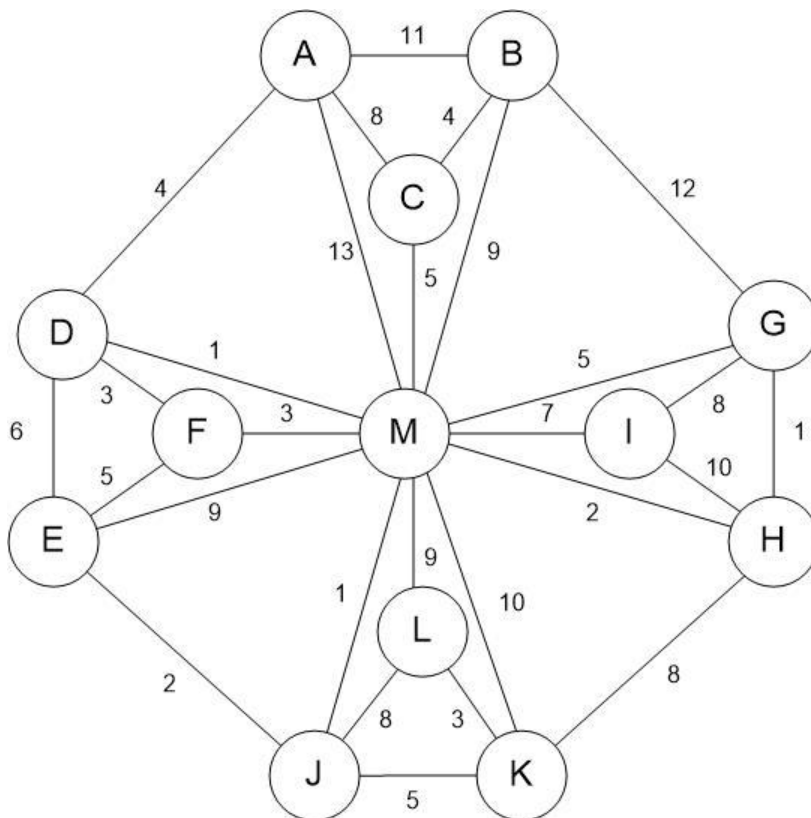


1. Execute Prim's minimum spanning tree algorithm by hand on the graph below showing how the data structures evolve specifically indicating when the distance from a fringe vertex to the tree is updated. Clearly indicate which edges become part of the minimum spanning tree and in which order. Start at vertex F.



2. Execute Kruskal's algorithm on the weighted tree shown below. Assume that edges of equal weight will be in the priority queue in alphabetical order and each edge name is ordered alphabetically. Clearly show what happens each time an edge is removed from the priority queue and how the dynamic equivalence relation changes on each step and show the final minimum spanning tree that is generated.



3. Examine the minimum spanning trees generated in the previous two problems. In both cases, indicate whether the spanning tree is unique. If it is not unique provide all other minimum spanning trees. Explain how you made the determination whether the minimum spanning tree is unique.
4. Given the following adjacency lists (with edge weights in parentheses) for a directed graph:

A: B(2), C(7), D(6)

B: C(3), F(1)

C: E(3)

D: E(3)

E: F(1)

F: C(3), D(1)

Execute Dijkstra's shortest-path algorithm by hand on this graph, showing how the data structures evolve, with A as the starting vertex. Clearly indicate which edges become part of the shortest path tree and in which order.