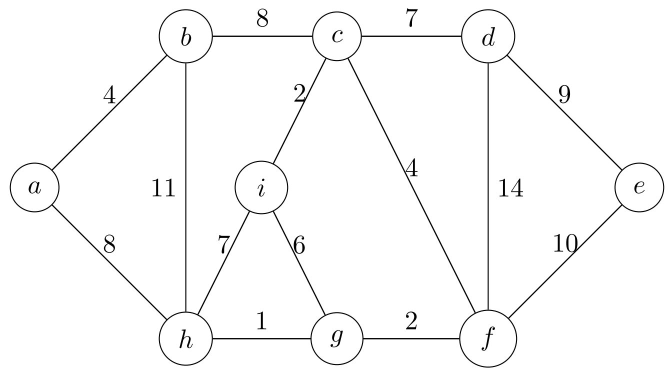
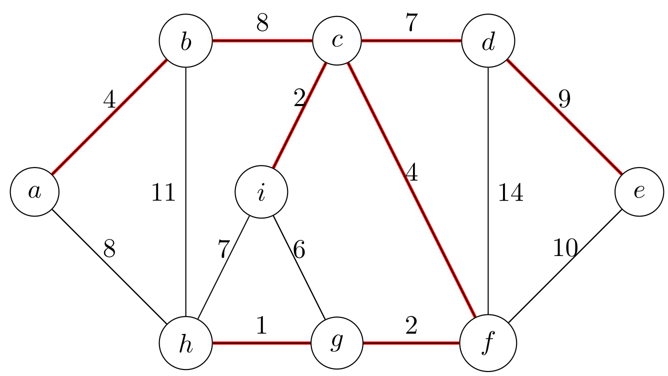
DSC 40B - Discussion 09

Problem 1.

Compute the minimum spanning tree for the following graph using Kruskal's algorithm. (Also compute the MST using Prim's algorithm and compare the results.)



Solution:



Problem 2.

Suppose we are given both an undirected graph G with weighted edges and a minimum spanning tree T of G.

a) Describe an efficient algorithm to update the minimum spanning tree when the weight of one edge e in T is decreased.

Solution: The minimum spanning tree of the updated graph would be T.

b) Describe an efficient algorithm to update the minimum spanning tree when the weight of one edge e not in T is increased.

Solution: The minimum spanning tree of the updated graph would be T.

c) Describe an efficient algorithm to update the minimum spanning tree when the weight of one edge e in T is increased.

Solution: Let e = (u,v) be the edge whose weight is increased. Remove the edge e from the minimum spanning tree T. This divides the tree T into two connected components. Let T_u be the component which contains u and T_v be the component which contains v. We can identify T_u and T_v by running a BFS with u and v as the sources. This takes time O(V + E). While running BFS we can also label each node as 0 if it is a part of T_u and 1 if it is a part of T_v. We can now examine each edge e in the graph and find the minimum weight edge which connects a node labelled 0 to a node labelled 1. This takes time O(E). We can then add this edge to T to get the minimum spanning tree of the updated graph. The total time complexity is O(V + E).

d) Describe an efficient algorithm to update the minimum spanning tree when the weight of one edge e not in T is decreased.

Solution: Let e= (u,v) be the edge whose weight is decreased. Add the edge e to the minimum spanning tree T. This would result in a cycle in T. We can identify the nodes and the edges on the cycle by running BFS with u or v as the source. This takes time O(V + E). Find the maximum weight edge on the cycle and remove it from T. This takes time O(E). The resultant tree would be a minimum spanning tree for the updated graph. The total time complexity is O(V +E).