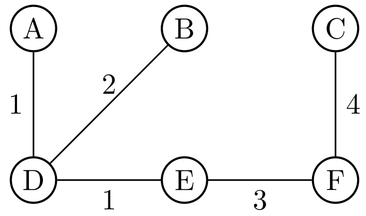
COMP S265F Design and Analysis of Algorithms
Lab 11: Minimum Spanning Tree: Kruskal's Algorithm – Suggested Solution

Question 1. A minimum spanning tree is a spanning tree. By Fact 2 in Lab 11 slides, the number of edges of a spanning tree is $n-1$, where n is the number of vertices.

Question 2. The Kruskal's algorithm sorted all edges in non-descending order and consider whether to include it in the resultant minimum spanning tree (MST) one by one:

order	1	2	3	4	5	6	7	8	9
edge	(A,D)	(D,E)	(D,B)	(B,E)	(E,F)	(C,F)	(B,C)	(E,C)	(A,B)
weight	1	1	2	2	3	4	4	5	6
include or not	Yes	Yes	Yes	No	Yes	Yes	No	No	No

The resultant MST is as follows:



Question 3. Let T be the minimum spanning tree containing the edge (u,v). Let T_0 and T_1 be the two trees obtained by removing edge (u,v) from T. Let V_0 and V_1 be the vertices in T_0 and T_1, respectively.

Since the spanning tree T contains all vertices in V, and V_0 and V_1 are disjoint sets, (V_0, V_1) is a cut of the graph, and (u,v) is an edge crossing the cut (V_0, V_1).

Suppose, for the sake of contradiction, that (u,v) is not a light edge of (V_0, V_1). Then, there exists another edge (a,b) crossing the cut with a smaller weight, i.e., $w(a,b) < w(u,v)$. We can get a new spanning tree T' by adding (a,b) to connect T_0 and T_1. The weight of T' is

$$w(T') = w(T) - w(u,v) + w(a,b) = w(T) - (w(u,v) - w(a,b)) < w(T) \ ,$$

which contradicts that T is a minimum spanning tree. Therefore, (u,v) must be a light edge of (V_0, V_1).