3.5 Kruskal's Algorithm for MST

Kruskal's algorithm is used to find the shortest way between two connected weighted nodes, it divides a graph into a forest and considers each node as an individual tree. Kruskal's algorithm connects each tree or node in such a way that the connecting edge has the minimum value and no cycle is created in the resulting minimum spanning tree. In this article, we are going to discuss what is Kruskal's algorithm, the working of Kruskal's algorithm along with examples, complexity, and implementation.

Kruskal's algorithm begins with selecting the edge with minimum weight and keeps adding edges until all the vertices aren't covered.

Steps of Kruskal's Algorithm

Input: A connected weighted graph G.
Output: A minimal spanning tree T.

Step 1: **Remove Loops and Parallel Edges:** Discard all the loops and parallel edges from the weighted graph G, if they exist. Before sorting, keep only the edge with the minimum weight between any pair of vertices.

Step 2: **Sort All Edges by Weight:** Arrange all edges of the graph in ascending order based on their weights.

Step 3: **Select the Smallest Edge:** Start by selecting the edge with the smallest weight (if there is more than one edge of minimal weight, arbitrarily choose one of these edges).

Step 4: **Check for Cycle Formation:** If adding an edge creates a cycle, then reject that edge and go for the next least weight edge.

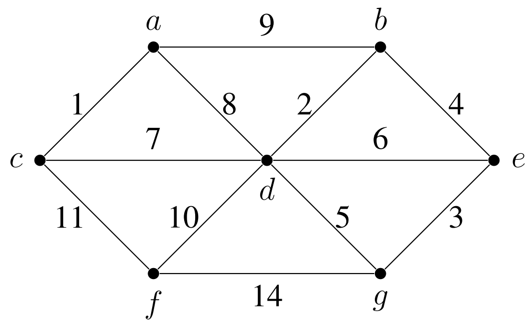
Step 5: **Repeat Until the MST is Formed:** Keep adding edges until all the vertices are connected and a Minimum Spanning Tree (MST) is obtained.

Step 6: **Stop After $(n-1)$ Edges:** If G has n vertices, stop after $(n-1)$ edges have been chosen. Otherwise, repeat Step 3.

Note *The time complexity of Kruskal's Algorithm is $O(m \log n)$ where n is the number of vertices and m is the number of edges in G.*

3.5.1 Illustrative Examples

Example 3.4 Construct the minimum spanning tree (MST) for the given graph using Kruskal's Algorithm



Graph $G(V, E)$

The weight of the edges of the above graph is given in the table below:

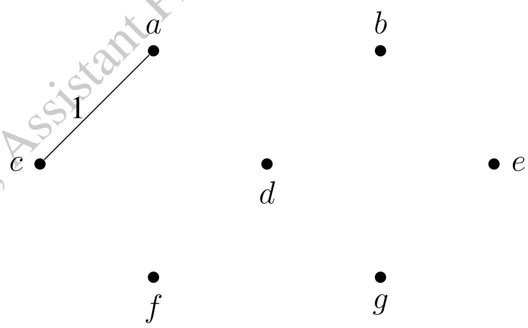
Edge	ab	be	eg	fg	cf	ca	da	db	de	dg	df	dc
Weight	9	4	3	14	11	1	8	2	6	5	10	7

Now, after sorting the edges given above in the ascending order of their weights.

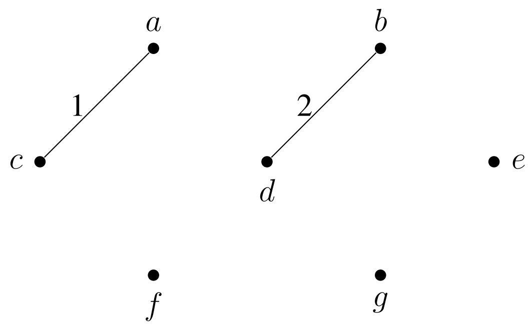
Edge	ca	db	eg	be	dg	de	cd	da	ab	df	cf	fg
Weight	1	2	3	4	5	6	7	8	9	10	11	14

Now, let's start constructing the minimum spanning tree.

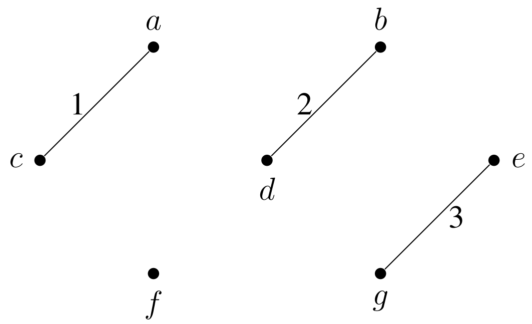
Step 1: The first pick edge ca, as it has a minimum edge weight that is 1 to the MST.



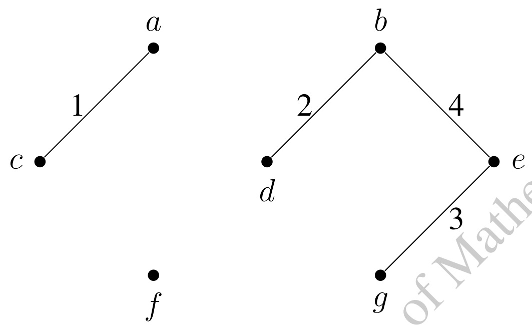
Step 2: Add the edge db with weight 2 to the MST as it is not creating the cycle.



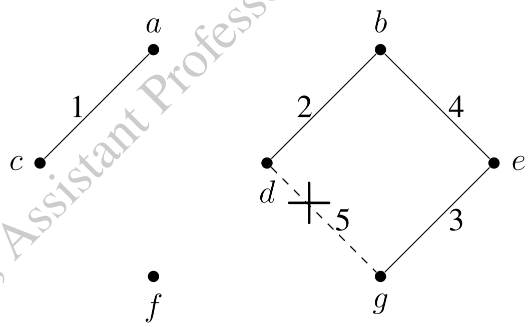
Step 3: Add the edge eg with weight 3 to the MST, as it is not creating any cycle or loop.



Step 4: Next up is edge be with weight 4.

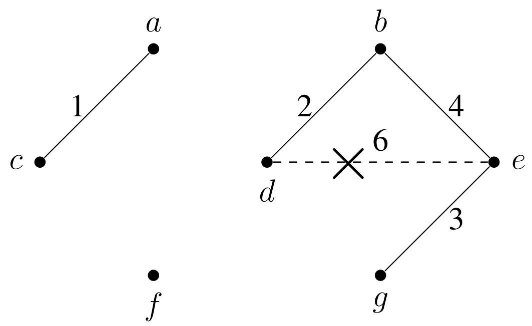


Step 5: Following edge be, you have edge dg with weight 5. This edge also creates the loop; hence you will discard it.



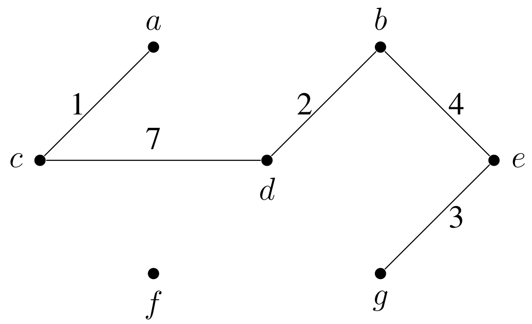
Edge dg should be discarded, as it creates loop.

Step 6: Next up is edge BD with weight 6. This edge also formulates a loop, so you will discard it as well.

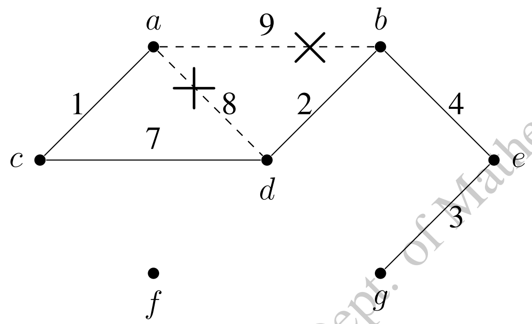


Edge de should be discarded, as it creates loop.

Step 7: Add the edge cd with weight 7 to the MST, as it is not creating any cycle or loop.

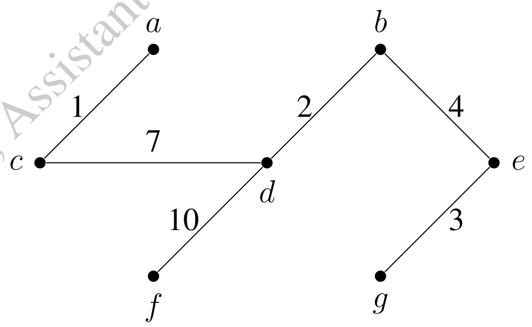


Step 8: Next, add the edges ds and ab with weights 8 and 9, respectively, individually to the MST. Since they create a cycle, discard them.



Edges da and ab should be discarded, as each creates a loop individually.

Step 9: Next on your sorted list is edge df with weight 10. This edge does not generate any cycle, so you need to include it in the MST structure.

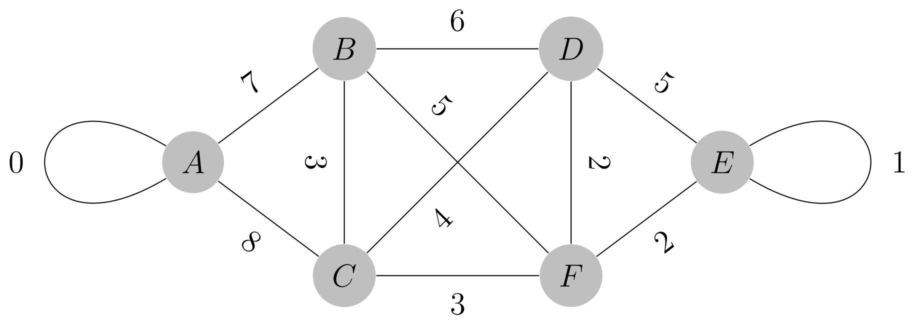


Minimum Spanning Tree $T(V', E')$.

Since the number of vertices in the graph is 7 and we have chosen 6 edges, we stop the algorithm and the minimal spanning tree is produced. The spanning tree contains all the vertices of G.
The summation of all the edge weights in MST $T(V', E')$ is

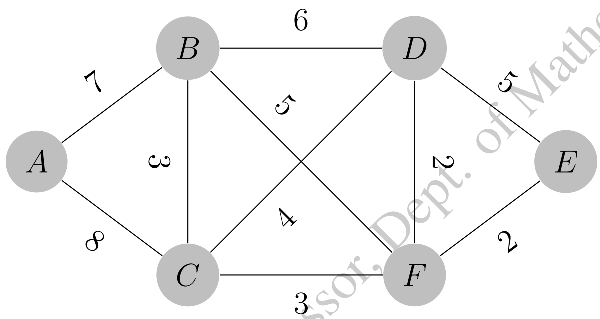
$$ac + db + eg + be + dc + df = 1 + 2 + 3 + 4 + 7 + 10 = 27,$$

which is the least possible edge weight for any possible spanning tree structure for this particular graph.
Example 3.5 The graph $G(V, E)$ given below contains 6 vertices and 12 edges. You will create a minimum spanning tree $T(V', E')$ for $G(V, E)$ such that the number of vertices in T will be 6, and the number of edges will be 5 (i.e., $6 - 1$).



Graph $G(V, E)$

If you observe this graph, you'll find two looping edges connecting the same node to itself again. And you know that the tree structure can never include a loop or parallel edge. Hence, primarily you will need to remove these edges from the graph structure.



After removing parallel edges or loops from the graph.

The weight of the edges of the above graph is given in the table below:

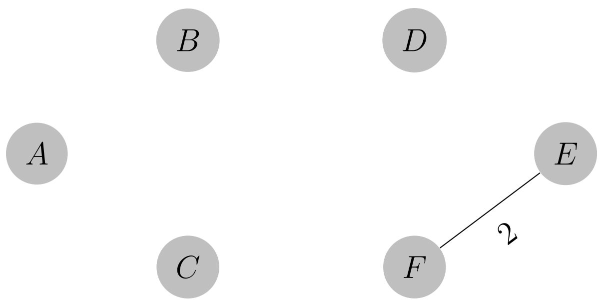
Edge	AB	BC	CD	DE	EF	FC	CA	DF	BD	BF
Weight	7	3	4	5	2	3	8	2	6	5

Now, after sorting the edges given above in the ascending order of their weights.

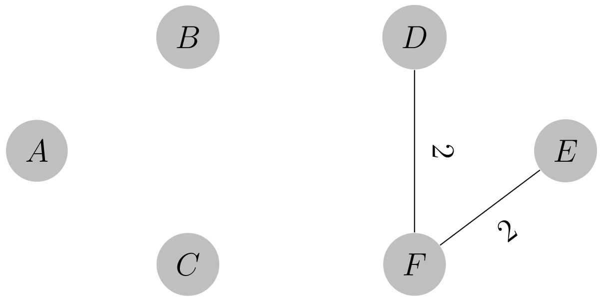
Edge	EF	DF	BC	FC	CD	BF	DE	BD	AB	CA
Weight	2	2	3	3	4	5	5	6	7	8

Now, let's start constructing the minimum spanning tree.

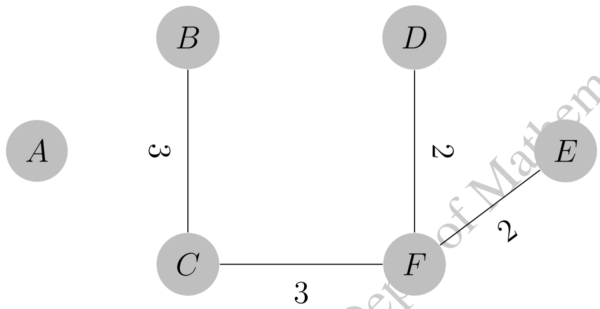
Step 1: The first pick edge EF, as it has a minimum edge weight that is 2 to the MST.



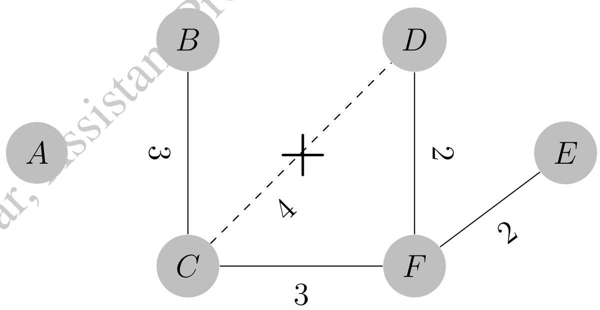
Step 2: Add the edge DF with weight 2 to the MST as it is not creating the cycle.



Step 3: Add the edge BC and edge CF with same weights 3 to the MST, as it is not creating any cycle or loop.

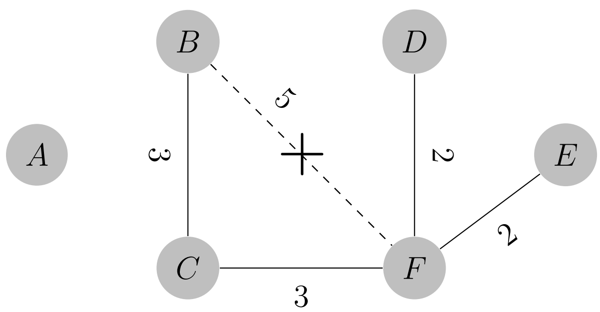


Step 4: Next up is edge CD with weight 4. This edge generates the loop in Your tree structure. Thus, you will discard this edge.



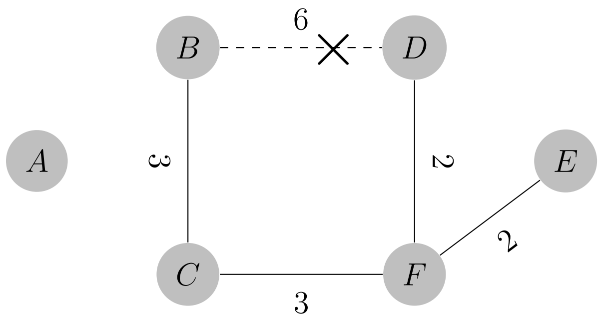
Edge CD should be discarded, as it creates loop.

Step 5: Following edge CD, you have edge BF with weight 5. This edge also creates the loop; hence you will discard it.



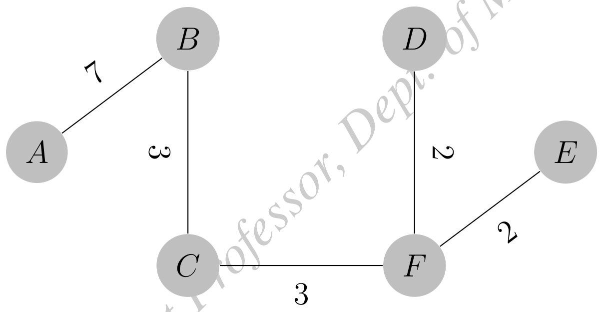
Edge BF should be discarded, as it creates loop.

Step 6: Next up is edge BD with weight 6. This edge also formulates a loop, so you will discard it as well.



Edge BD should be discarded, as it creates loop.

Step 7: Next on your sorted list is edge AB with weight 7. This edge does not generate any cycle, so you need to include it in the MST structure. By including this node, it will include 5 $(= 6 - 1)$ edges in the MST, so you don't have to traverse any further in the sorted list. The final structure of your MST is represented in the image below:



Minimum Spanning Tree $T(V', E')$.

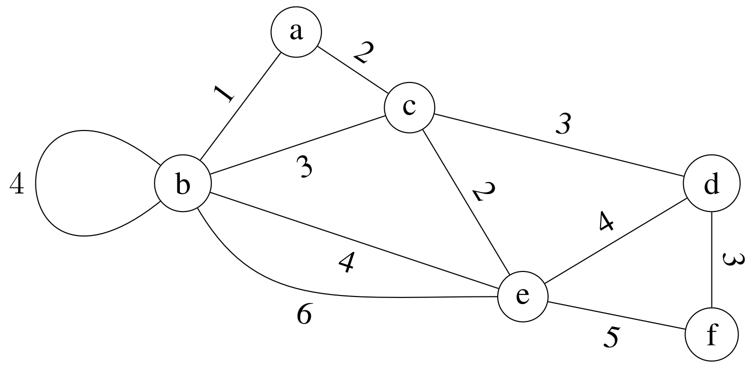
The summation of all the edge weights in MST $T(V', E')$ is

$$EF + DF + CF + BC + AB = 2 + 2 + 3 + 3 + 7 = 17,$$

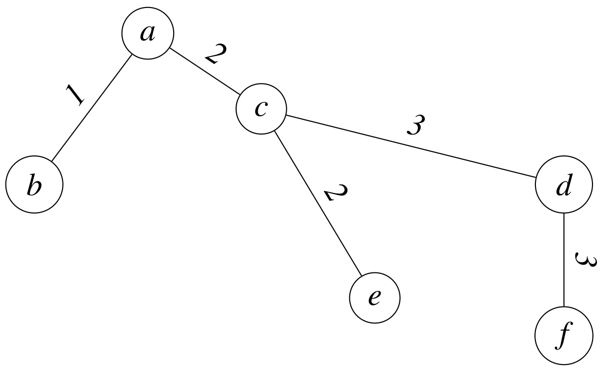
which is the least possible edge weight for any possible spanning tree structure for this particular graph.

3.6 Worked Out Exercises

Exercise 3.3 Obtain a minimal spanning tree of the following graph using Kruskal's Algorithm:

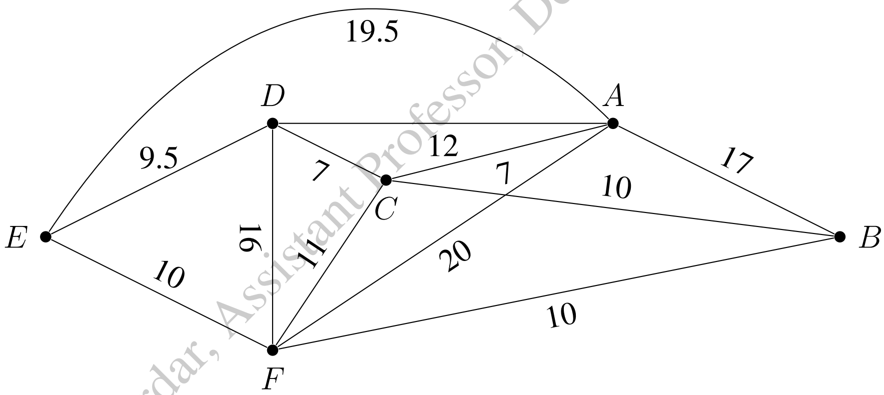


Solution *(hints)*

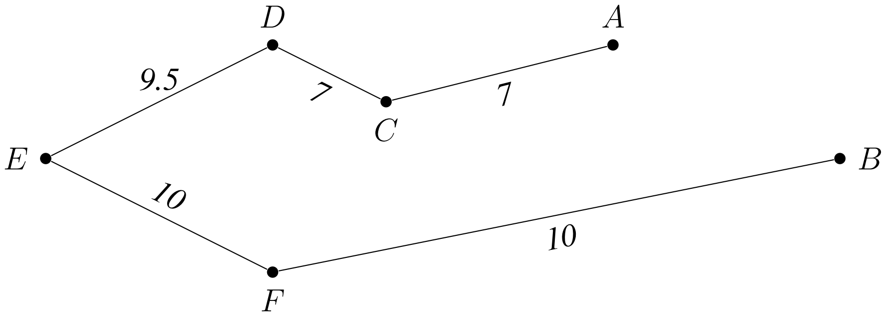


The weight of the minimum spanning tree is $1 + 2 + 2 + 3 + 3 = 11$.

Exercise 3.4 By Kruskal's algorithm find a minimal spanning tree and the corresponding weight of the spanning tree in the following graph:

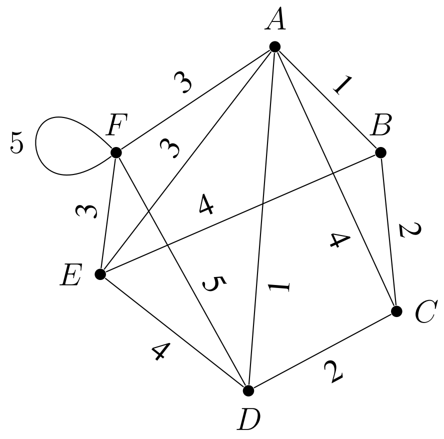


Solution *(hints)*

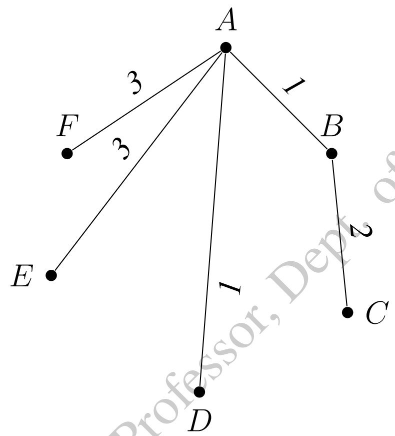


The weight of the minimum spanning tree is $7 + 7 + 9.5 + 10 + 10 = 43.5$.

Exercise 3.5 Obtain a minimal spanning tree of the following graph using Kruskal's Algorithm:



Solution *(hints)*



The weight of the minimum spanning tree is $1 + 1 + 2 + 3 + 3 = 10$.