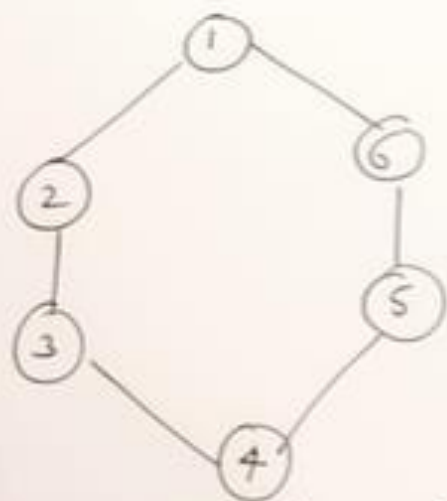


Prims and Kruskal Algorithms (Minimal Spanning Tree)

Minimum Cost Spanning Tree

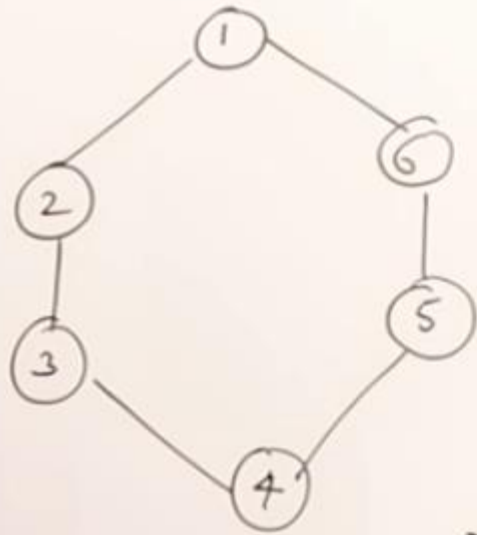


$$G = (V, E)$$

$$V = \{1, 2, 3, 4, 5, 6\}$$

$$E = \{(1, 2), (2, 3), (3, 4), \dots\}$$

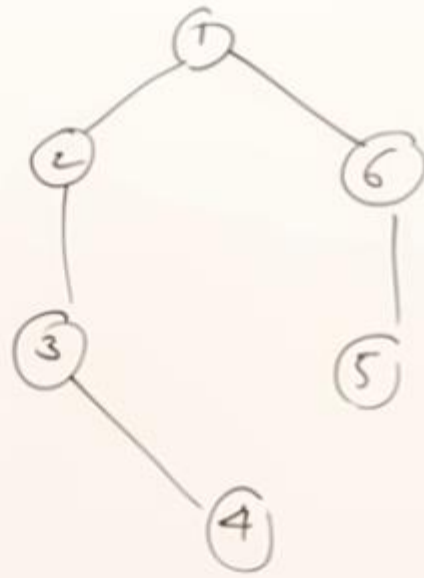
Minimum Cost Spanning Tree



$$G = (V, E)$$

$$V = \{1, 2, 3, 4, 5, 6\}$$

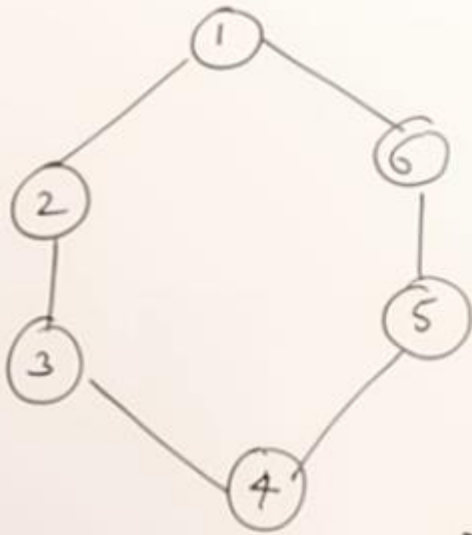
$$E = \{(1,2), (2,3), (3,4), \dots\}$$



$$|V| = n = 6$$

$$|V| - 1 = 5$$

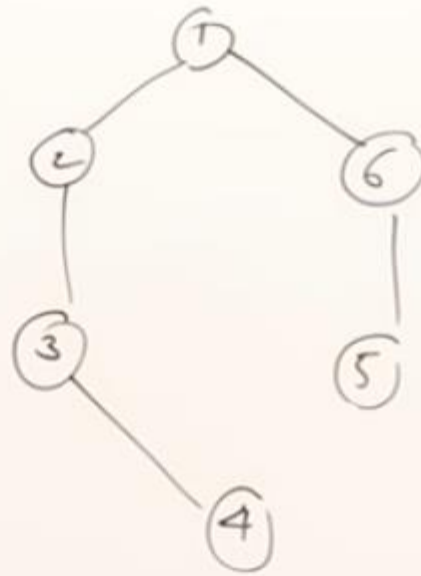
Minimum Cost Spanning Tree



$$G = (V, E)$$

$$V = \{1, 2, 3, 4, 5, 6\}$$

$$E = \{(1,2), (2,3), (3,4), \dots\}$$



$$|V| = n = 6$$

$$|V| - 1 = 5$$

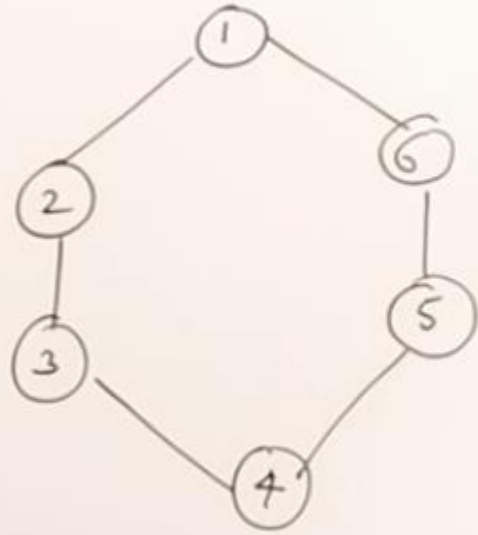
$$S \subseteq G$$

$$S = (V', E')$$

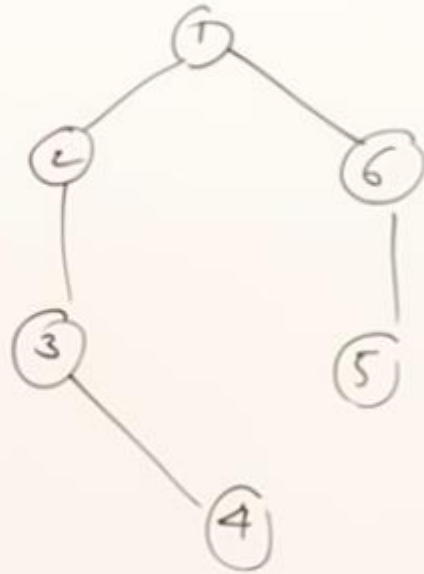
$$V' = V$$

$$|E'| = |V| - 1$$

Minimum Cost Spanning Tree

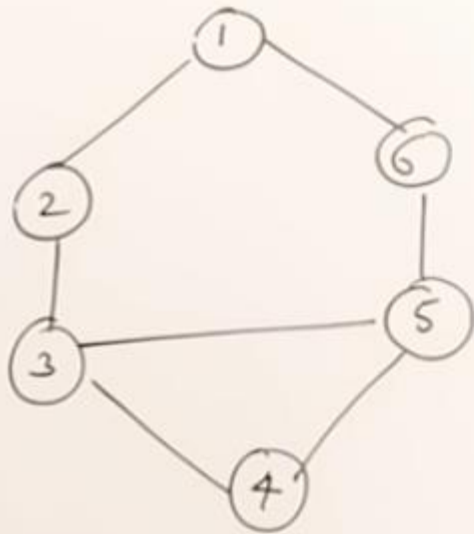


$$|E| = 6$$

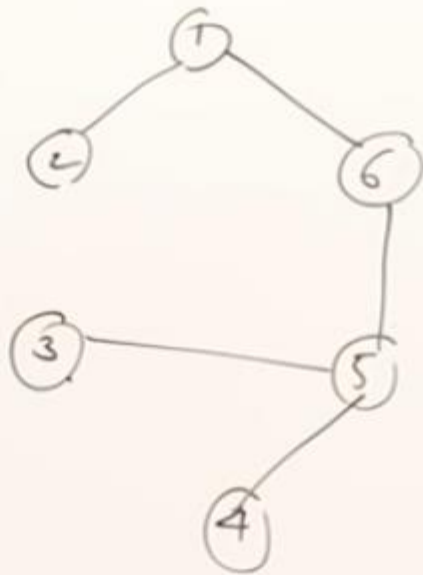


$$6C_5 = 6$$

Minimum Cost Spanning Tree



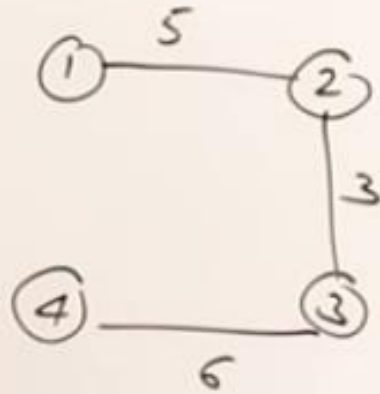
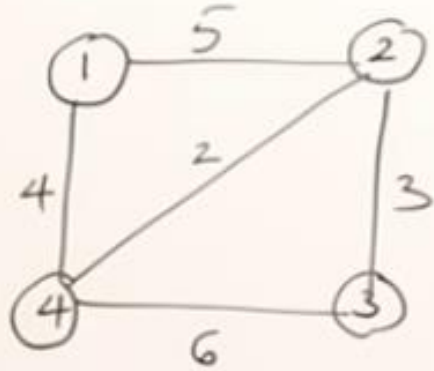
$$|E| = 6$$



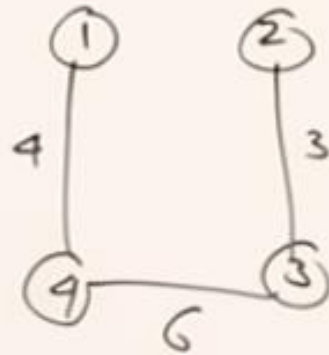
$$7C_5 - 2$$

$$|E| \leq |V| - 1 - \text{no. of cycles}$$

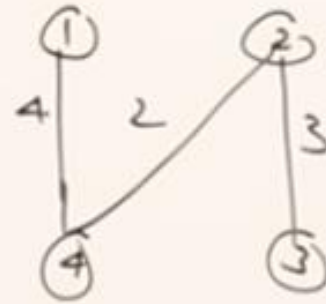
Minimum Cost Spanning Tree



Cost = 14



Cost = 13



Cost = 9

Prim's Algorithm for Minimum Spanning Tree (MST)

Prim's Algorithm

Step 1: Determine an arbitrary vertex as the starting vertex of the MST.

Step 2: Follow steps 3 to 5 till there are vertices that are not included in the MST (known as fringe vertex).

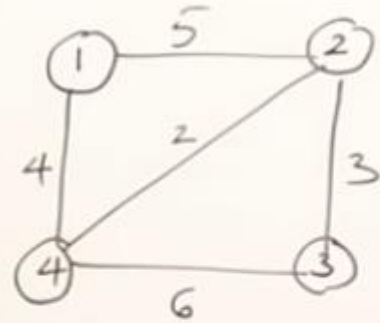
Step 3: Find edges connecting any tree vertex with the fringe vertices.

Step 4: Find the minimum among these edges.

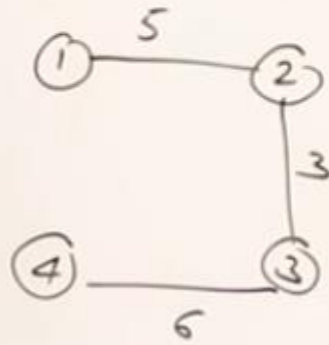
Step 5: Add the chosen edge to the MST if it does not form any cycle.

Step 6: Return the MST and exit

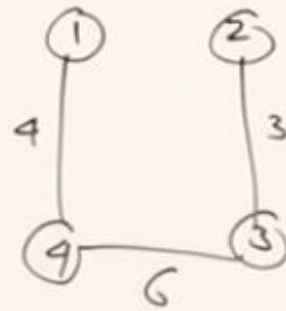
Minimum Cost Spanning Tree



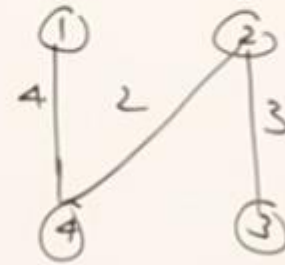
- 1- Prim's
- 2- Kruskal's



$$\text{Cost} = 14$$



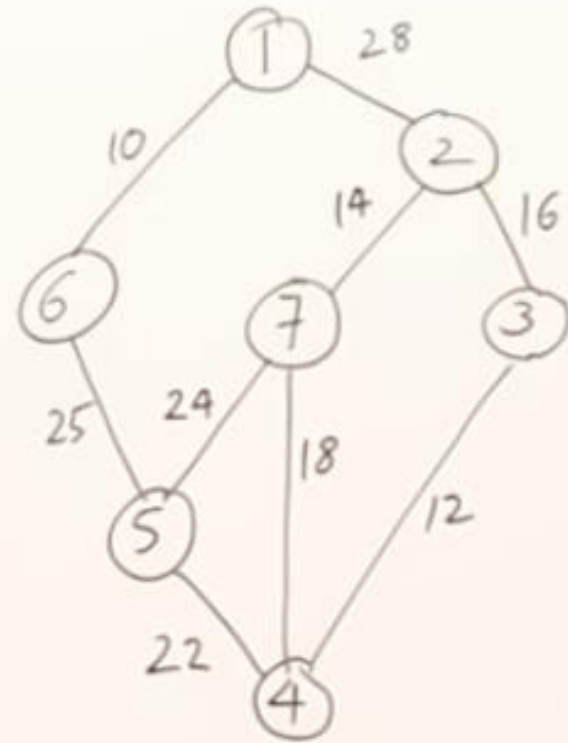
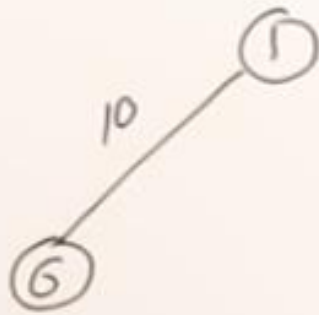
$$\text{Cost} = 13$$



$$\text{Cost} = 9$$

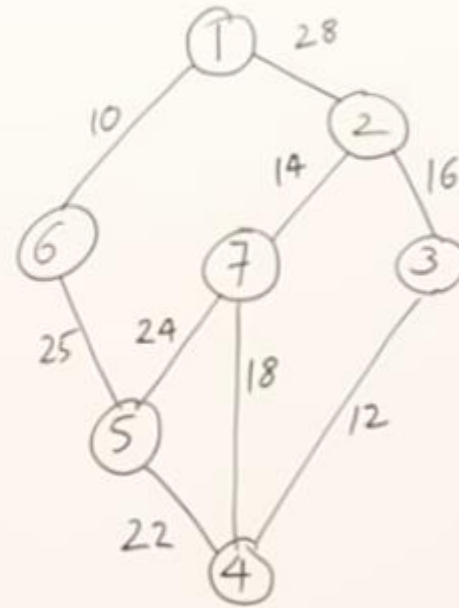
Minimum Cost Spanning Tree

Prim's



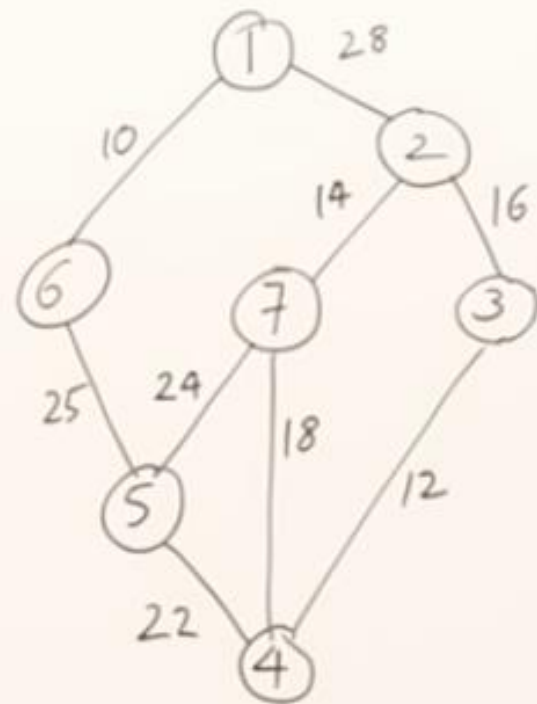
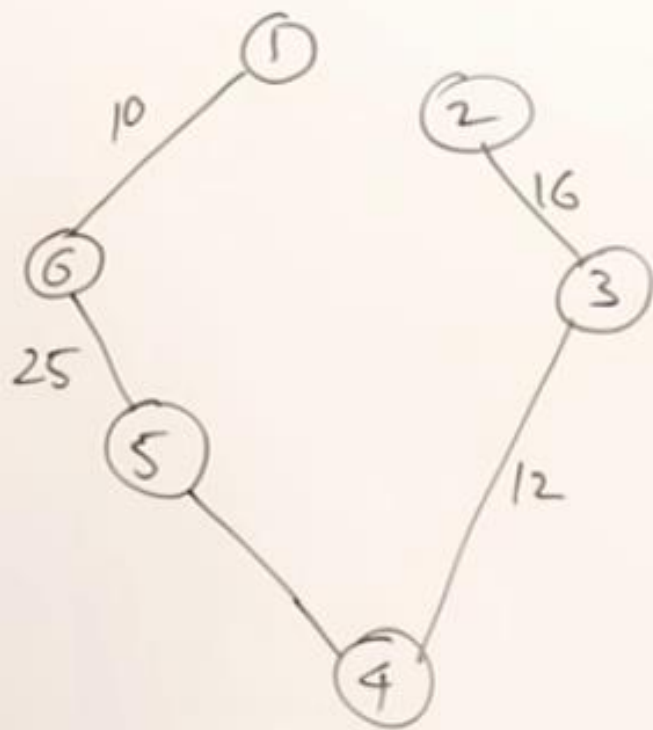
Minimum Cost Spanning Tree

Prim's



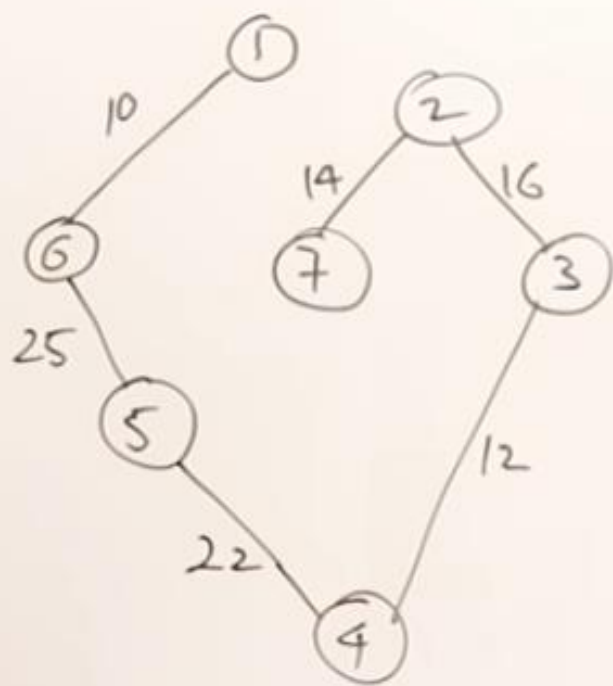
Minimum Cost Spanning Tree

Prim's

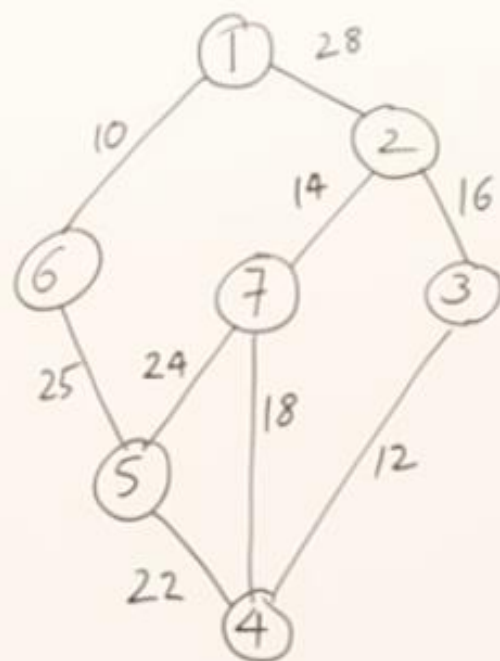


Minimum Cost Spanning Tree

Prim's

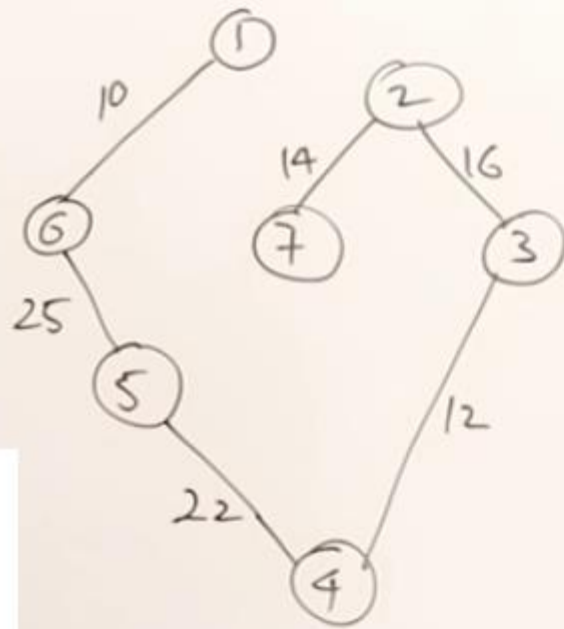


$$\text{Cost} = 99$$

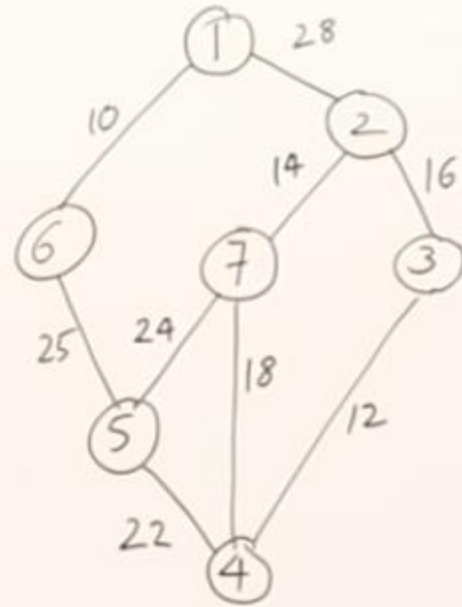
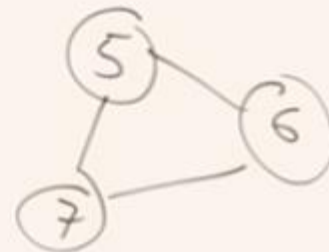
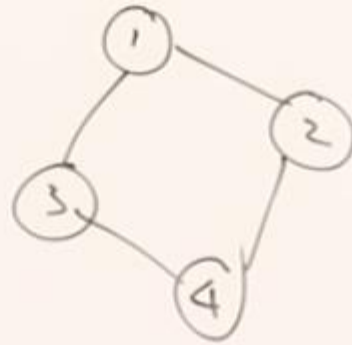


Minimum Cost Spanning Tree

Prim's



$$\text{Cost} = 99$$



Kruskal's Algorithm for Minimum Spanning Tree (MST)

Kruskal's Algorithm

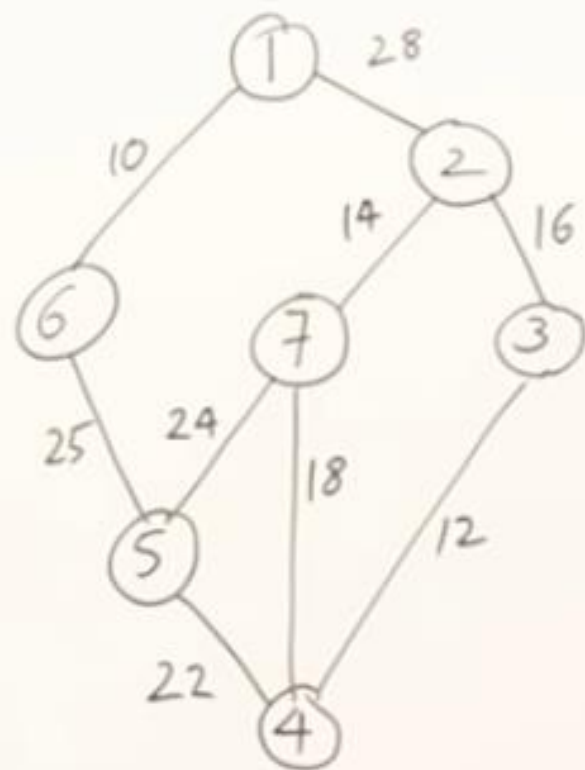
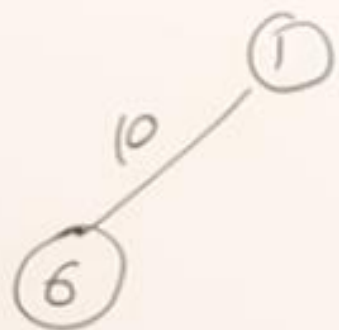
Step 1: Sort all the edges in non-decreasing order of their weight.

Step 2: Pick the smallest edge. Check if it forms a cycle with the spanning tree formed so far. If the cycle is not formed, include this edge. Else, discard it.

Step 3: Repeat step 2 until there are $(V-1)$ edges in the spanning tree.

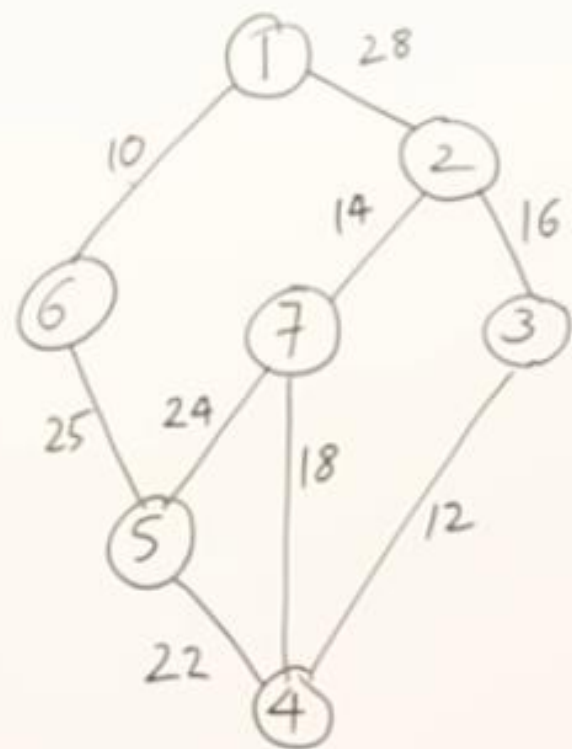
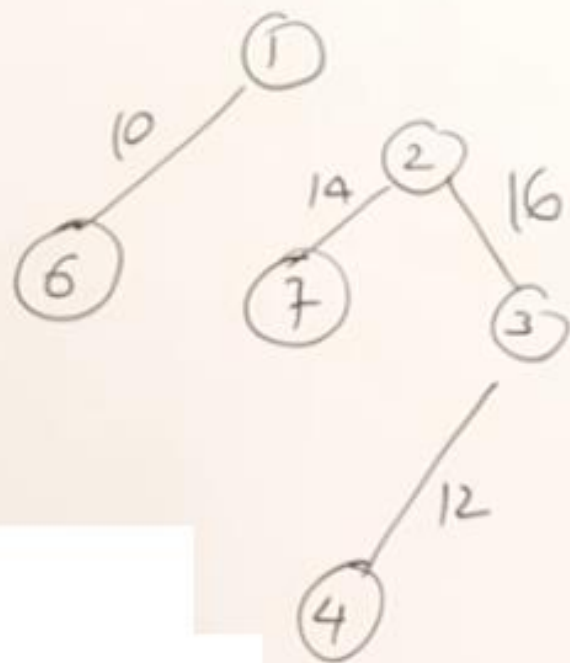
Minimum Cost Spanning Tree

Kruskal's



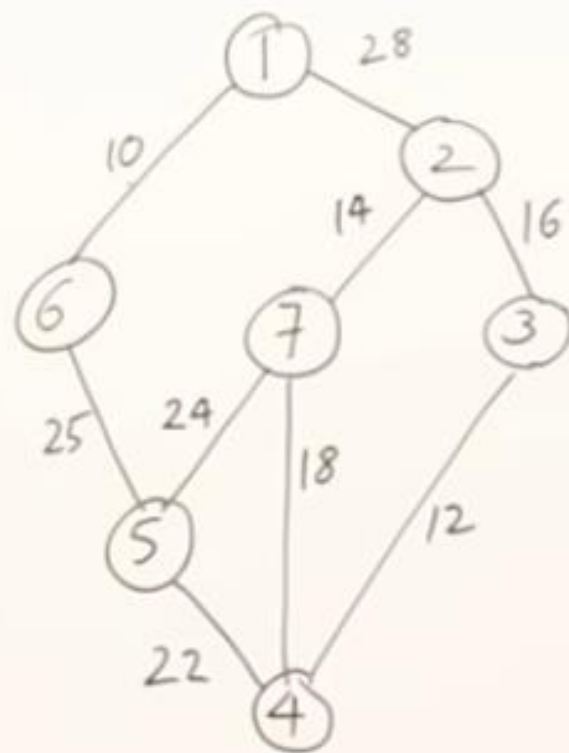
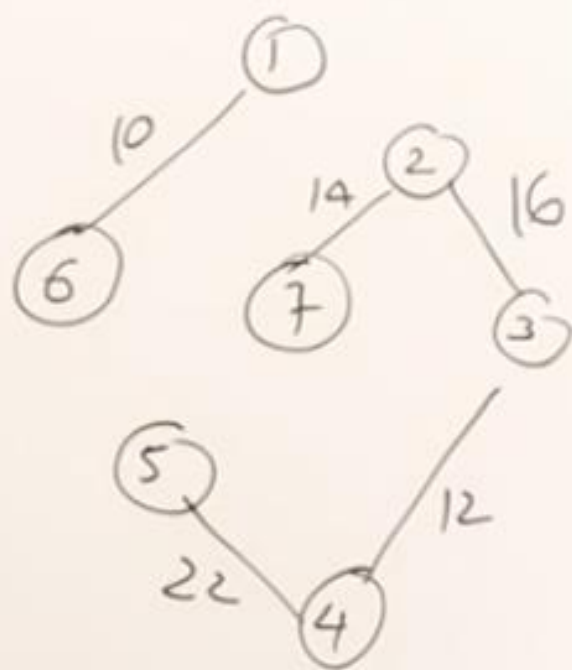
Minimum Cost Spanning Tree

Kruskal's



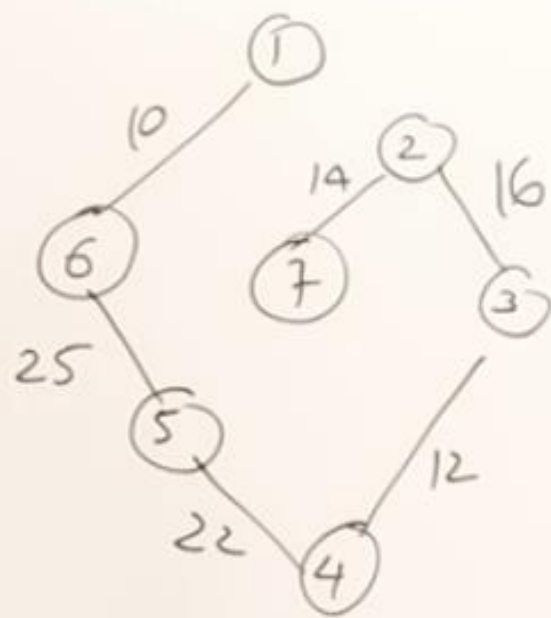
Minimum Cost Spanning Tree

Kruskal's

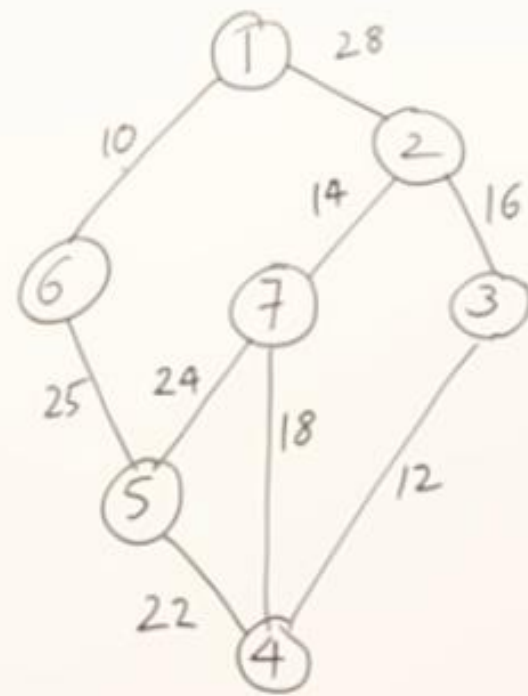


Minimum Cost Spanning Tree

Kruskal's



Cost = 99



Difference between Prim's and Kruskal Algorithm

S.No.	Prim's Algorithm	Kruskal's Algorithm
1	This algorithm begins to construct the shortest spanning tree from any vertex in the graph.	This algorithm begins to construct the shortest spanning tree from the vertex having the lowest weight in the graph.
2	To obtain the minimum distance, it traverses one node more than one time.	It crosses one node only one time.
3	The time complexity of Prim's algorithm is $O(V^2)$.	The time complexity of Kruskal's algorithm is $O(E \log V)$.
4	In Prim's algorithm, all the graph elements must be connected.	Kruskal's algorithm may have disconnected graphs.
5	When it comes to dense graphs, the Prim's algorithm runs faster.	When it comes to sparse graphs, Kruskal's algorithm runs faster.
6	It prefers list data structure.	It prefers the heap data structure.

Thank you