SPANNING TREE

A spanning tree is a subset of graph G which has all the vertices covered with minimum possible number of edges.

If $G = (V, E)$ is a graph

Here, V = set of vertices

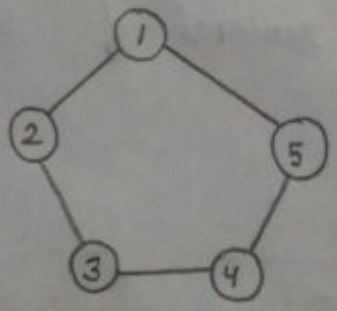
E = set of edges

then $G' = (V', E')$ is a ST

Here, $V' = V$

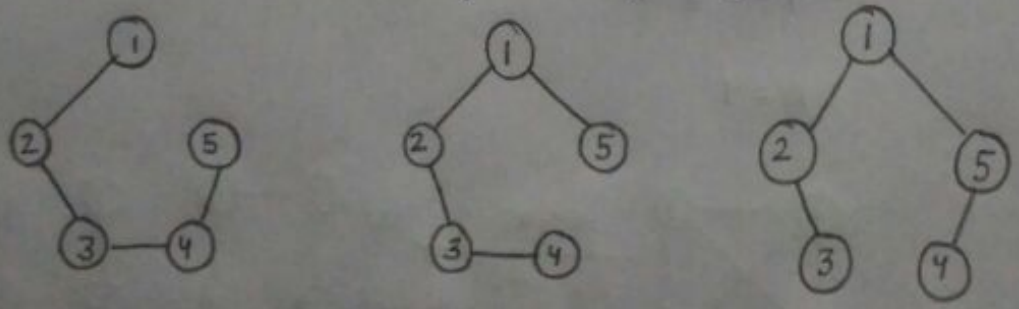
$E' \subset E$ and $n(E') = n(V) - 1$

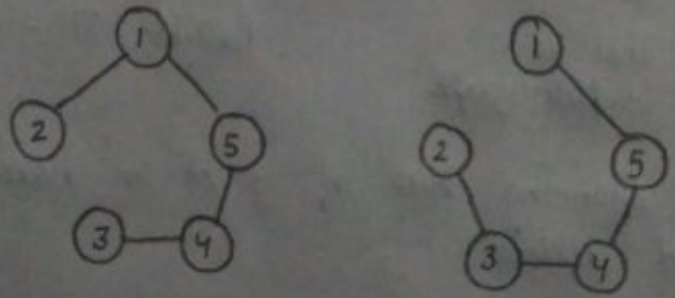
Example :



graph G

Possible spanning trees for graph G:

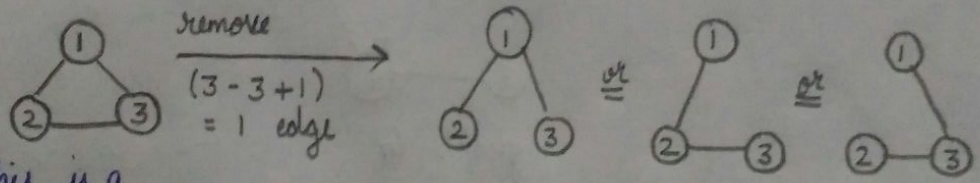




Properties of a Spanning Tree

- Removal of any one edge from the ST will make it disconnected.
- Adding one edge to the ST will create a loop
- A ST ~~must~~ can not contain cycles.
- A ST can not be disconnected
- ~~If each edge~~ A complete undirected graph can have n^{n-2} number of ST
- Every connected and undirected graph has atleast one ST.
- Disconnected graph doesn't have any ST.
- From complete graph, by removal of $(e-n+1)$ edges, we can construct a ST

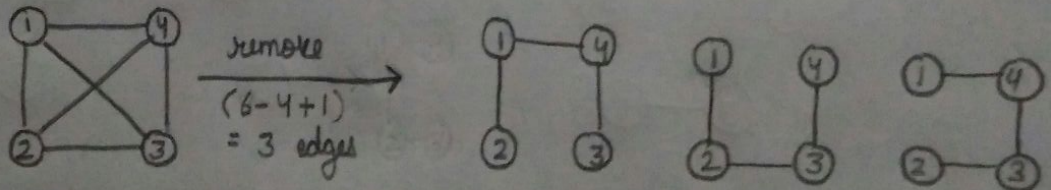
example : $n = 3$



This is a complete graph

Possible ST = $3^{3-2} = 3$

$n = 4$



This is a complete graph

Possible ST = $4^{4-2} = 16$

- All possible spanning trees of graph G have same number of edges and vertices.

Applications of Spanning Tree

Spanning tree is used to find a minimum path to connect or span all the nodes in a graph.

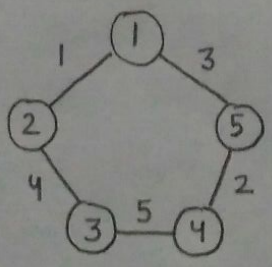
Common applications are:

- Civil network planning
- Computer network routing protocol
- Cluster Analysis.

MINIMUM SPANNING TREE

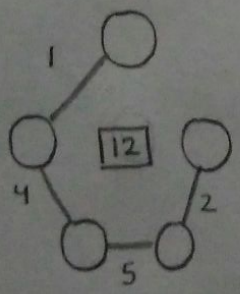
In a weighted graph, a minimum spanning tree is a spanning tree that has minimum weight than all other spanning trees of the same graph.

Example:

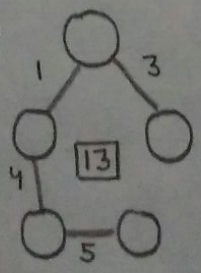


graph G (weighted)

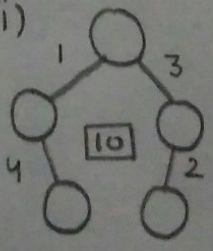
(i)



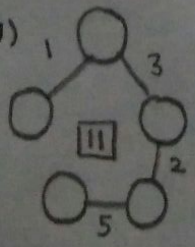
(ii)



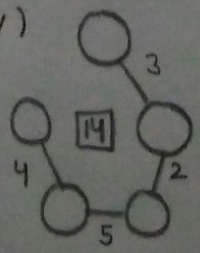
(iii)



(iv)



(v)



From above ST, (iii) is the minimum spanning tree because it has minimum weight = 10

- If each edge has distinct weight then, there will be only one unique MST
- It has minimum total cost.

Applications of MST

- To find paths in the map.
- To design networks like telecommunication networks, water supply networks & electrical grids.

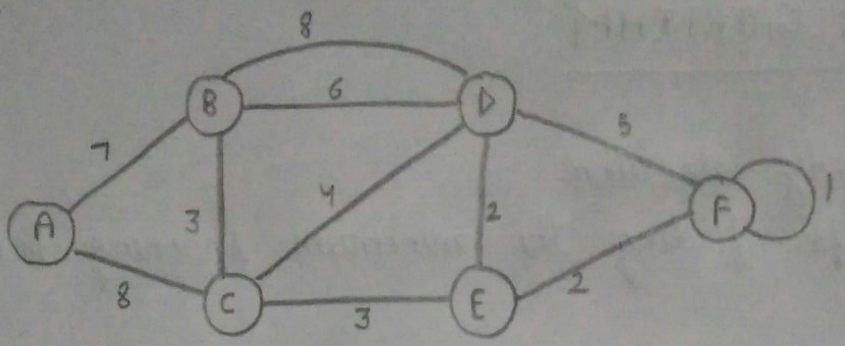
PRIM'S ALGORITHM

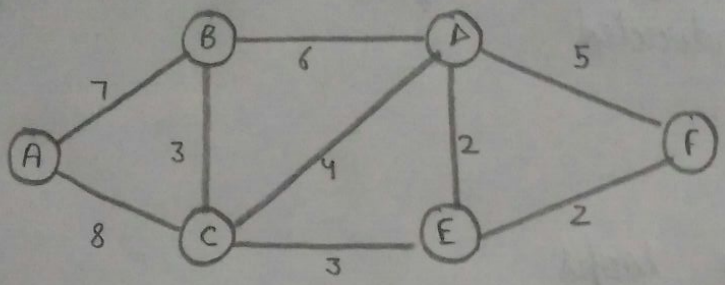
- It is a greedy algorithm
- It is used for finding the minimum spanning tree of a given graph.
- To apply Prim's algorithm, graph must be weighted, connected and undirected.

Algorithm:

1. Remove all the loops
2. Remove parallel edge with greater weight
3. Randomly choose any vertex. The vertex connecting to the edge having least weight is usually selected
4. Find all the edges that connect the tree to new vertices. Find the least weight edge among those edges and include it in the existing tree. If including that edge creates a cycle, reject that edge and look for the next least weight edge.
5. Keep repeating step 4 until all the vertices are included and MST is obtained.

Example :

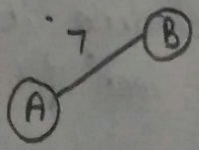




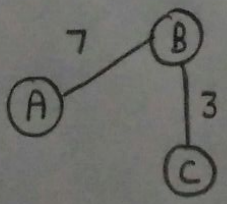
Loops & parallel edges are removed

MST :

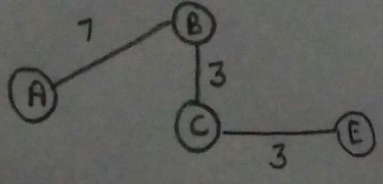
A is selected



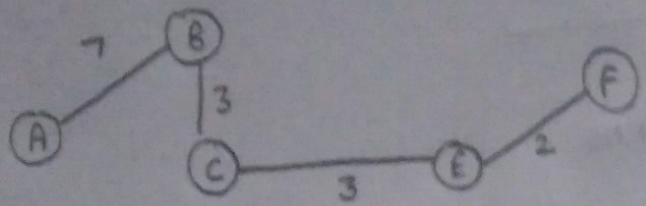
A is connected to two edges with weights 7 & 8
7 is min therefore it is included



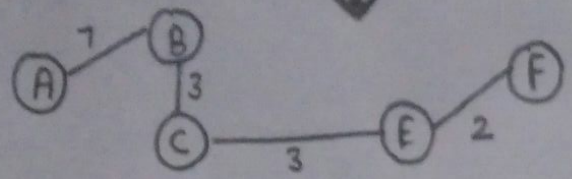
B is connected to edges with weights 3 and 6
Previous remaining edge - 8
Minimum is 3



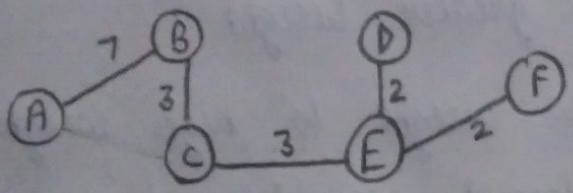
C is connected to edges with weights 8, 4 & 3
Previous remaining edge - 6
Minimum is 3



E is connected to edges with weights - 2, 2
Previous remaining edges - 8, 6, 4
Minimum = 2



F is connected to edges with weights - 5
Previous remaining edges - 8, 6, 4, 2
Minimum = 2



All nodes are included

This is the minimum spanning tree of given graph.

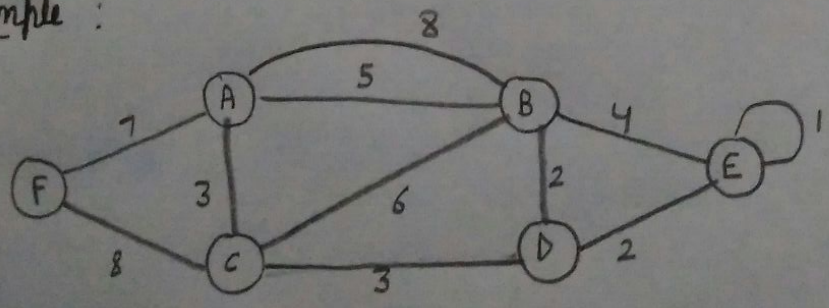
KRUSKAL'S ALGORITHM

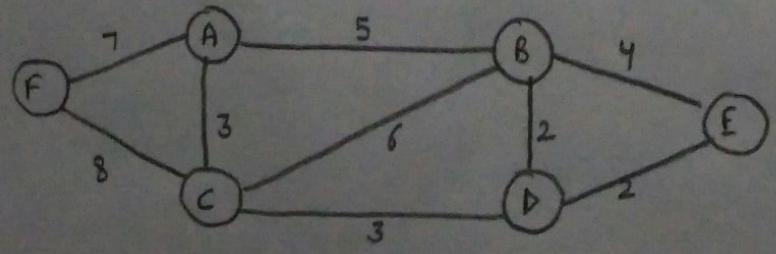
- It is a greedy algorithm
- It is used for finding the minimum spanning tree of a given graph.

Steps

1. Remove all the loops.
2. Remove parallel edges with greater weight.
3. Sort all the edges from low weight to high weight.
4. Take the edge with the lowest weight and use it to connect the vertices of graph. If adding an edge creates a cycle, then reject that edge and go for the next least weight edge.
5. Keep adding edges until all the vertices are connected & MST is obtained.

Example :



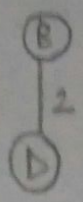


loops and parallel edges are removed

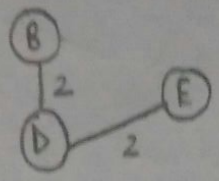
MST :

BD = 2
DE = 2
AC = 3
CD = 3
BE = 4
AB = 5
BC = 6
AF = 7
FC = 8

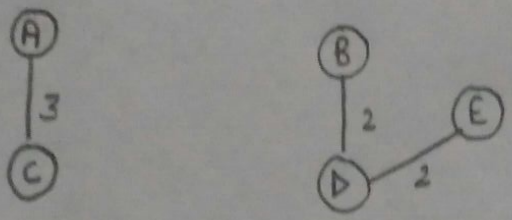
BD=2



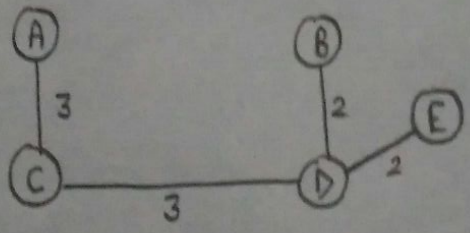
DE=2



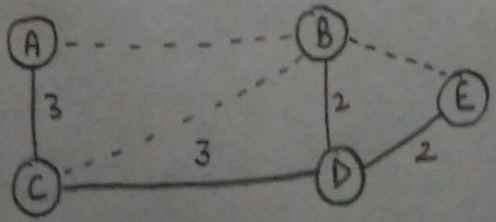
AC=3



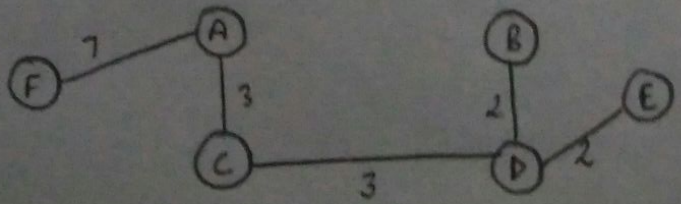
CD=3



BE = 4
AB = 5
BC = 6
forms cycle
Therefor rejected



AF=7



This is the MST for given tree graph