Networks

Network- A graph where one can go from any one vertex to any other vertex

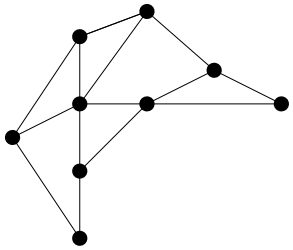
Minimum Network Problem- a problem in which we want to form a network of the smallest total cost

Subgraph- some portion of the original graph

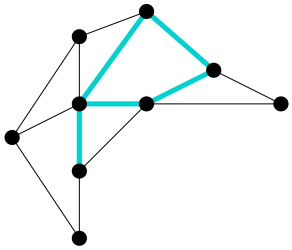
Spanning Subgraph- a subgraph that includes every one of the vertices of the original graph

Tree- a graph that is connected and has no circuits

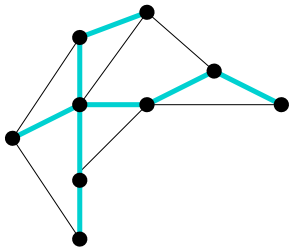
NETWORK



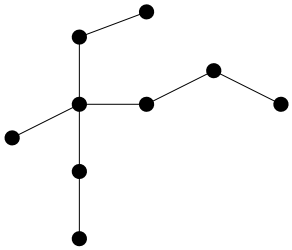
SUBGRAPH



SUBGRAPH



SPANNING SUBGRAPH



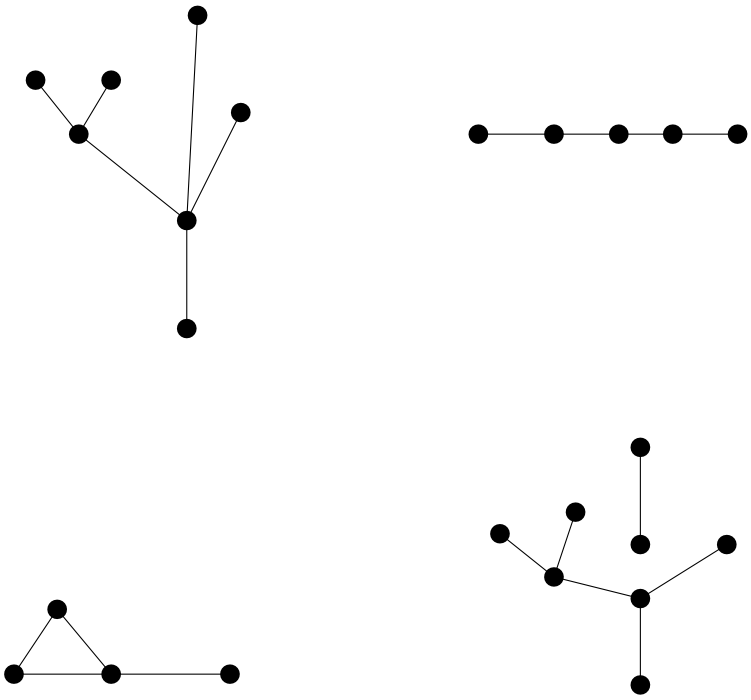
Properties of Trees

1. **If a graph is a tree, there is one and only one path joining any two vertices. Conversely, if there is one and only one path joining any two vertices of a graph, the graph must be a tree.**

2. **In a tree, every edge is a bridge. Conversely, if every edge of a connected graph is a bridge, then the graph must be a tree.**

3. **A tree with N vertices must have N-1 edges.**

4. **A connected graph with N vertices and N-1 edges must be a tree.**



Spanning Tree- a connected subgraph having N-1 of the edges and all N vertices

Minimum Spanning Tree- a spanning tree of a weighted graph that has the least total weight.

KRUSKAL'S ALGORITHM

1. **Find the cheapest edge in the graph and mark it.**

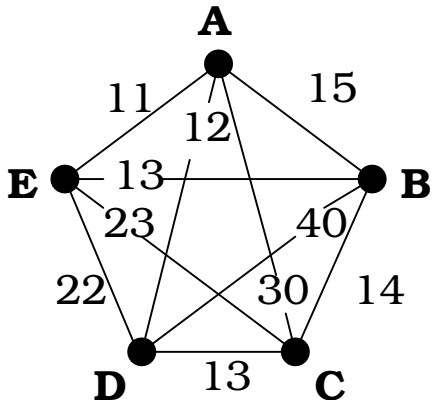
2. **Find the next cheapest edge in the graph and mark it.**

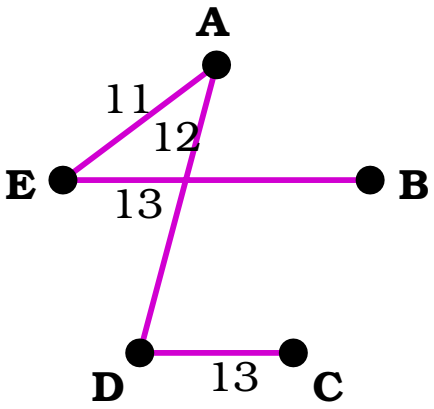
3. **Continue marking the next cheapest edge unless that edge creates a circuit with the other marked edges.**

4. **Continue until all vertices are included.**

Pros: Optimal Algorithm (guaranteed to find cheapest) Efficient Algorithm (work increases proportionally)

Find the minimum spanning tree of the graph below:





11+12+13+13=49

Find the minimum way to network the cities:

	Boston	Dallas	Houston	Louisville	Nashville
Boston	–	1748	1804	941	1088
Dallas	1748	–	243	819	660
Houston	1804	243	–	928	769
Louisville	941	819	928	–	168
Nashville	1088	660	769	168	–

Use Kruskal's Algorithm to find the Minimum Spanning Tree: