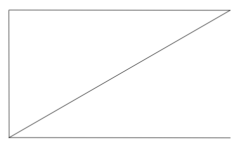
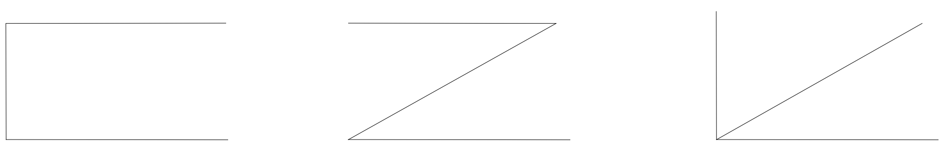
Minimum Spanning Tree

Spanning tree: A spanning tree T is a sub-graph of G(V,E), which has all the vertices covered with minimum possible number of edges (n-1).hence , a spanning tree does not contain cycles and it cannot be disconnected.
By the definition we can conclude that every connected and undirected graph G has at least one spanning tree. A disconnected graph does not have spanning tree.

Example: **G(V,E)**



Spanning tree possible:



Weighted graph: If each edge of E has a weight, G is called a weighted graph.

Minimum Spanning Tree: In a weighted graph, a minimum spanning tree is a spanning tree that has minimum weight than all other spanning tree of the same graph. In real world situation, this weight can be measured as distance , congestion,traffic load or any arbitrary value denoted to the edges.

Problem:
Given an undirected, connected, weighted graph G=(V,E).

- We wish to find an acyclic subset $T \subseteq E$ that connects all the vertices and whose total weight:

$$w(T) = \sum_{(u,v) \in T} w(u,v) \text{ is minimized.}$$

Where w(u,v) is the weight of edge (u,v).

- T is called a minimum spanning tree of G.

Solution:

- Using greedy method.
- Two algorithms:
    - ✓ **Prim's algorithm.**
    - ✓ **Kruskal's algorithm.**

Approach:

- The tree is built edge by edge.
- Let T be the set of edges selected so far.
- Each time a decision is made:

  * Include an edge e to T s.t. :
  Cost (T)+w (e) is minimized, and
  $T \cup \{e\}$ does not create a cycle.