Spanning Tree → A tree T is said to be a spanning tree of a connected graph G if T is a subgraph of G and T contains all vertices of G.

- also called skeleton or a maximal tree subgraph
- S.T. is defined only for a connected graph
- A disconnected graph with k components has a spanning forest consisting of k spanning trees.

How to find S.T. →

- If G has no circuit then it is its own S.T.
- If G has a circuit, delete an edge from the circuit
- If there are more circuits in G then repeat the operation till an edge from the last circuit is deleted — leaving a connected, circuit free graph that contains all the vertices of G.

Note → Every connected graph has at least one S. Tree.

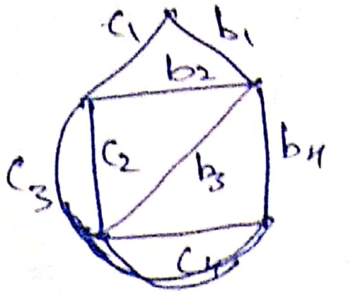
An edge in a S. Tree T is called a branch of T.
An edge of G that is not in a given S.T. T is called a chord

Let $T \to$ S.T. then $\bar{T} \to$ complement of T in G

$\bar{T} \to$ Chord Set $\quad (T^*) \to$ Co-spanning Tree or $\bar{T}$

$G = T \cup \bar{T}$

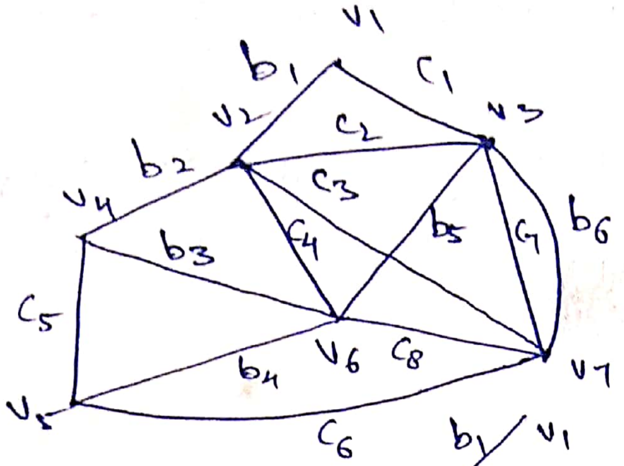
e.g.



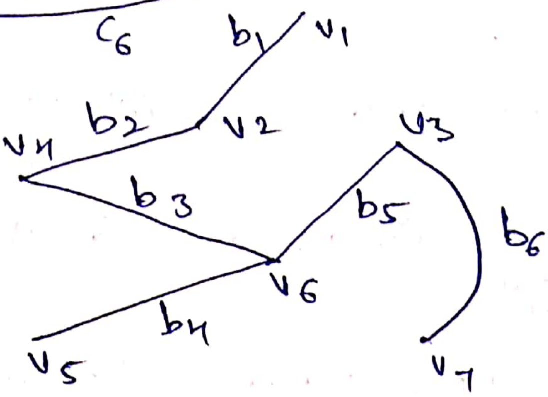
Graph 'G'

S.T. →

S.T. 'T'



S.T.



<u>Note</u> → w.r.t. any of its spanning trees, a connected graph with n vertices and e edges has $(n-1)$ tree branches and $(e-n+1)$ chords

e.g. For above graph. —

$n = 7, \; e = 14$

no. of branches $= 6$

no. of chords $= 14 - 7 + 1 = 8$

Any other spanning tree will yield the same number.

Rank & Nullity →

$n(V) = n$, $n(E) = e$

no. of components $= K$

rank $r = n - K$

nullity $\mu = e - n + K = e - r$, $\because r = n - K$

$\Rightarrow \mu + r = e$

$\Rightarrow$ rank + nullity = no. of edges in a graph

If G is connected then $K = 1$

$\Rightarrow r = n - 1$

and $\mu = e - n + 1$

$\therefore$ rank of a connected graph

= no. of branches in any S.T. of G

nullity of connected graph = no. of chords in G.

Fundamental Circuit →

Note → A connected graph G, is a tree iff adding an edge b/w any two vertices in G creates exactly one circuit.

Now consider a S.T. T in connected graph G. Adding any one chord to T will create exactly one circuit. Such a circuit formed by adding a chord to a S.T. is called a fundamental circuit.

How many fundamental circuits?

Exactly as many as the numbers of chords. i.e. $e - n + 1$

Set of these $(e - n + 1)$ fundamental circuits is called fundamental system of circuits relative to the S.T.

Minimal S.T. → G → weighted graph

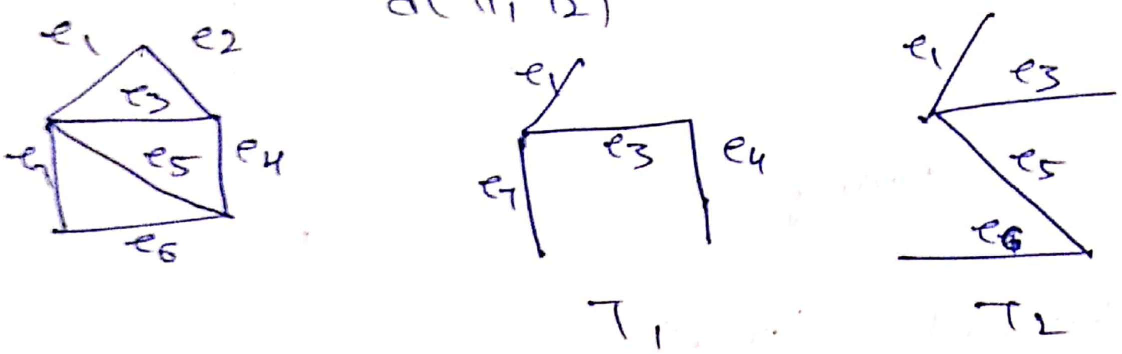
cost of S.T. = sum of all costs of all the branches in T

In general, different S.Ts of a costed graph will have diff. costs. A S.T. with the smallest cost in a weighted graph is called a minimal S.T. or a shortest S.T.

Distance b/w Two S.Ts → T_1, T_2

— no. of edges ~~of~~ of G present in one tree but not in the other

$d(T_1, T_2)$



$d(T_1, T_2) = 2$

Note → The distance b/w the S.Ts of a graph is a metric.

Note → A S.T. T of a costed graph G is a minimal S.T. iff. ∄ no other S.T. of G at a distance of one from T whose cost is smaller than that of T.

Algorithms for minimal S.T. →

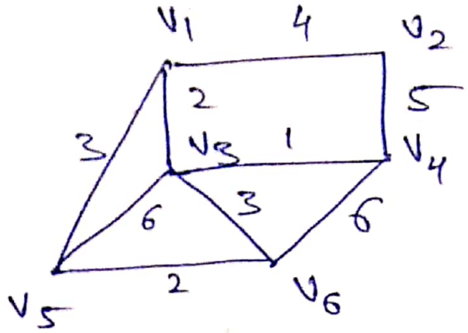
1. Kruskal's Algo 2. Prim's Algo

— List all edges of the weighted Graph G in order of non-decreasing cost.
— Select smallest edge of G
— Select a new edge of smallest possible cost that forms no circuit with the

∴ previously selected edges

— Repeat this process until $(n-1)$ edges have been selected.

Ex→



Total ~~edges~~ vertices $= 6$

∴ no. of edges in S.T. $= 6-1 = 5$

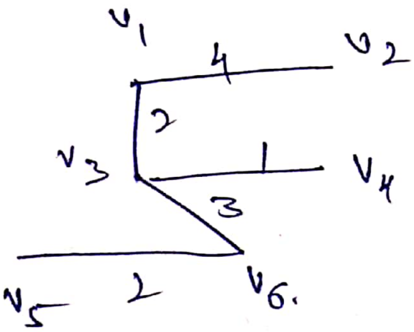
List all edges in order of non decreasing cost→

(v_3, v_4), (v_1, v_3), (v_5, v_6), (v_3, v_6), (v_1, v_5), (v_1, v_2), (v_2, v_4), (v_4, v_6), (v_3, v_5)

Select $(v_3, v_4) \rightarrow (v_1, v_3) \rightarrow (v_5, v_6) \rightarrow (v_3, v_6) \rightarrow (v_1, v_2)$

5 edges have been selected.

Minimal S.T. →



Total cost. of S.T. = 12 units.

Note→ Prim's algo might give different minimal S.T. but total cost will remain same