

# Minimal Spanning Tree

P. Sreenivasulu Reddy and Abdusalam Mahamed Derdar  
Department of mathematics, Samara University  
Semera, Afar Regional State, Ethiopia. Post Box No.131

**Abstract:** In this paper authors introduced new concept to determine minimal spanning tree and minimal spanning tree by using matrix method which is different from the Primes, kruskals and etc. algorithms.

## 1. INTRODUCTION

Kruskal's Algorithm and Prim's Algorithm are two most important spanning tree algorithms and both are called greedy algorithms. These greedy algorithms that find a minimum spanning tree for a connected weighted graph. They find a tree of that graph which includes every vertex and the total weight of all the edges in the tree is less than or equal to every possible spanning tree. In the paper authors determine minimal and maximal spanning trees in the simplest way for any weighted graph by using matrix method and we named it as PS Reddy method.

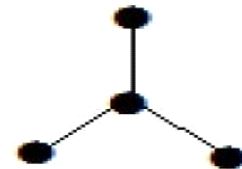
## 2. PRELIMINARIES:

**2.1 Connectivity:** A graph is said to be connected if there is a path between every pair of vertex. From every vertex to any other vertex, there should be some path to traverse. That is called the connectivity of a graph. A graph with multiple disconnected vertices and edges is said to be disconnected.

**2.2 Tree:** A connected acyclic graph is called a tree. In other words, a connected graph with no cycles is called a tree. The edges of a tree are known as branches. Elements of trees are called their nodes. The nodes without child nodes are called leaf nodes.

A tree with 'n' vertices has 'n-1' edges. If it has one more edge extra than 'n-1', then the extra edge should obviously has to pair up with two vertices which leads to form a cycle. Then, it becomes a cyclic graph which is a violation for the tree graph.

Example 1



The graph shown here is a tree because it has no cycles and it is connected. It has four vertices and three edges, i.e., for 'n' vertices 'n-1' edges as mentioned in the definition.

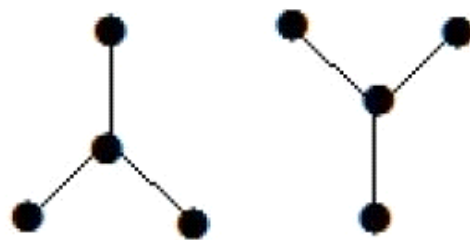
Example 2



In the above example, the vertices 'a' and 'd' has degree one. And the other two vertices 'b' and 'c' has degree two. This is possible because for not forming a cycle, there should be at least two single edges anywhere in the graph. It is nothing but two edges with a degree of one.

**2.3 Forest:** A disconnected acyclic graph is called a forest. In other words, a disjoint collection of trees is called a forest.

Example: The following graph looks like two sub-graphs; but it is a single disconnected graph. There are no cycles in this graph. Hence, clearly it is a forest.

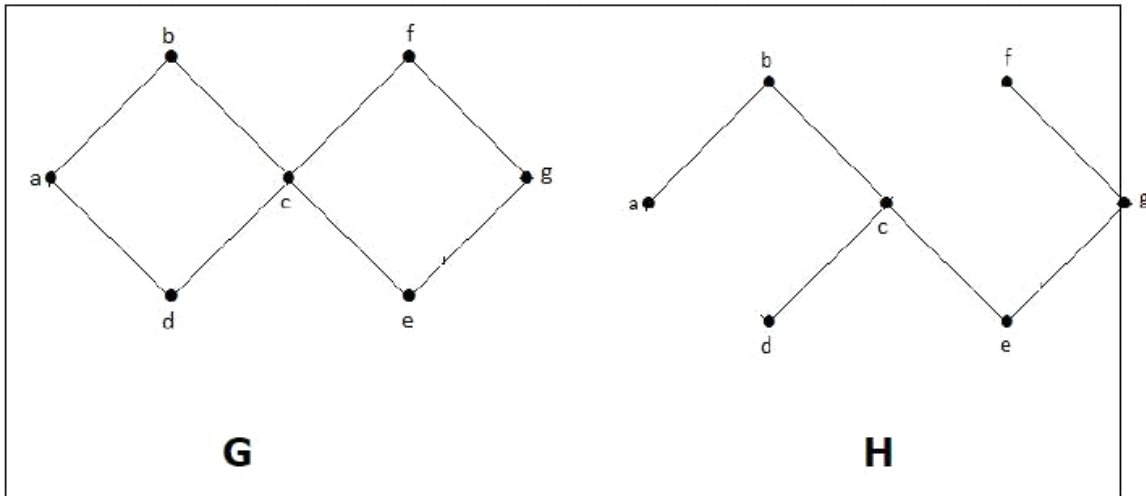


**2.4 Spanning Trees:** Let G be a connected graph, then the sub-graph H of G is called a spanning tree of G if –

1. H is a tree
2. H Contains All Vertices Of G.

A spanning tree T of an undirected graph G is a subgraph that includes all of the vertices of G.

Example



In the above example, G is a connected graph and H is a sub-graph of G.

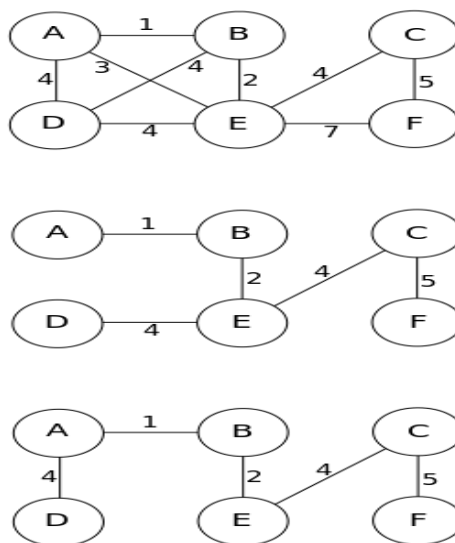
Clearly, the graph H has no cycles, it is a tree with six edges which is one less than the total number of vertices. Hence H is the Spanning tree of G.

2.5 Weighted Graph: A graph is called weighted graph if every edge assign some number or value or weight.

2.6 Cyclic graph: A cyclic graph or circular graph is a graph that consists of at least single cycle

2.7 Minimum Spanning Tree (MST): 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. In real-world situations, this weight can be measured as distance, congestion, traffic load or any arbitrary value denoted to the edges.

Example:



In the above example there are two minimal spanning trees with weight 16.

### 3. PS REDDY ALGORITHM FOR MINIMAL SPANNING TREE:

In general, it is easy to calculate number of spanning trees directly: If G is itself a tree, then number of spanning trees = 1. If G is the cyclic graph with n vertices then number of spanning trees = n. For a complete graph with n vertices gives the number of spanning trees as  $n^{n-2}$ .

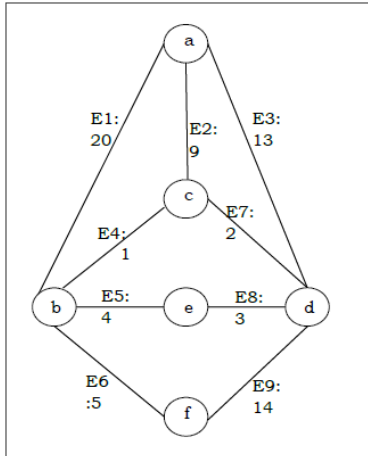
If G is the complete bipartite graph  $K_{(p,q)}$  then number of spanning trees =  $p^{q-1}q^{p-1}$ .

For the n-dimensional hypercube graph, the number of spanning trees =  $2^{2^n - n - 1}$ . It is very difficult to find minimal spanning tree by drawing each spanning tree. To avoid these difficulties authors are introducing this algorithm apart from Kruskals algorithm and Primes algorithm.

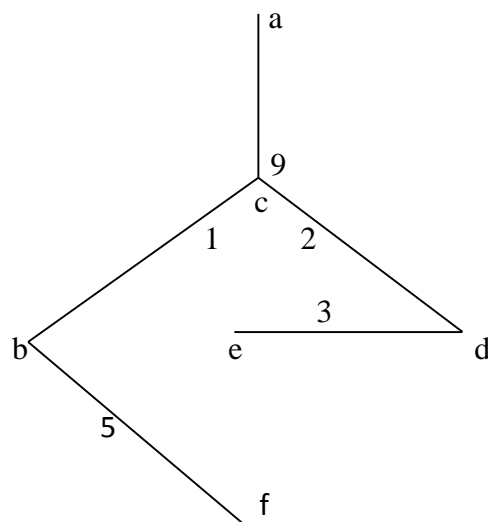
Main steps involving in the PS Reddy algorithm for minimal spanning tree:

1. Remove the loops at any vertex. And remove parallel edge which has maximum weight.
2. Form the adjacent matrix where as the elements of matrix is defined as  $v_{i,j} = \begin{cases} \text{weight of edge if } e \in (v_i, v_j) \\ 0, \text{ if } e \notin (v_i, v_j) \end{cases}$
3. Select the minimum element (except 0's) in first row and label that edge is 1
4. Select the minimum element in the second row which is not labeled in the first row and label it as 2. And continue the process until finishing all rows of the matrix.
5. One of the labeled edge forms the cycle of the graph so ignore that edge and then draw the graph which is minimal spanning tree.

Example1:

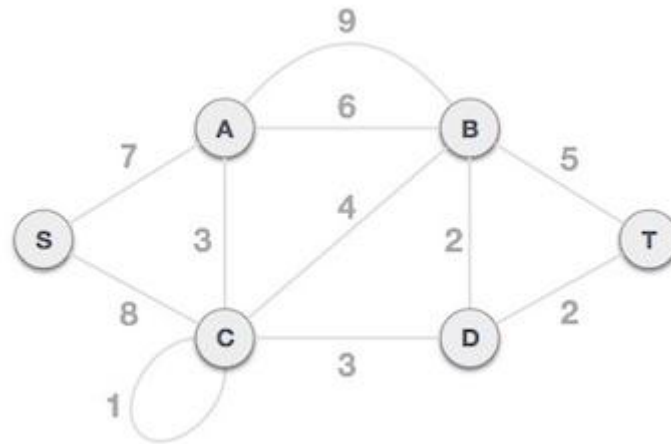


	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	
<i>a</i>	0	20	9	13	0	0	ac edge1
<i>b</i>	20	0	1	0	4	5	bc edge2
<i>c</i>	9	1	0	2	0	0	cd edge3
<i>d</i>	13	0	2	0	3	14	de edge4
<i>e</i>	0	4	0	3	0	0	eb edge5
<i>f</i>	0	5	0	14	0	0	fb edge6

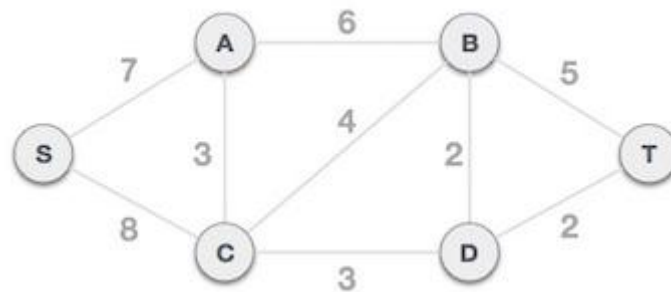


Draw the graph with all the labeled edges which is shown in the last figure except eb edge5 because it forms the cycle. This is the minimal spanning tree and its total weight is  $(1 + 2 + 3 + 5 + 9) = 20$ .

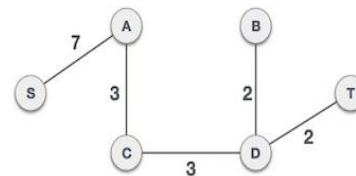
Example2:



By removing loop and maximum weight parallel edge then the graph is



	S	A	B	C	D	T	
S	0	7	0	8	0	0	SA edge1
A	7	0	6	3	0	0	AC edge2
B	0	6	0	4	2	5	BD edge3
C	8	3	4	0	3	0	CD edge4
D	0	0	2	3	0	2	DT edge5
T	0	0	5	0	2	0	TB edge6



Draw the graph with all the labeled edges which is shown in the last figure except TB edge6 because it forms the cycle. This is the minimal spanning tree and its total weight is  $(7 + 3 + 3 + 2 + 2) = 17$ .

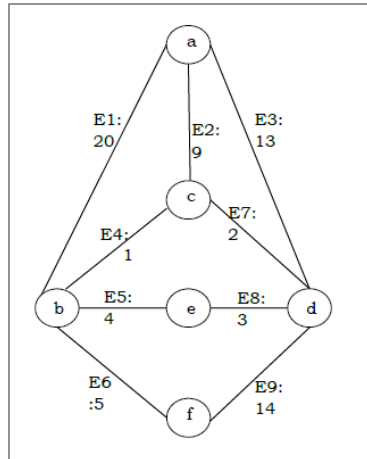
Main steps involving in the PS Reddy algorithm for maximal spanning tree:

1. Remove the loops at any vertex. And remove parallel edge which has minimum weight.
2. Form the adjacent matrix where as the elements of

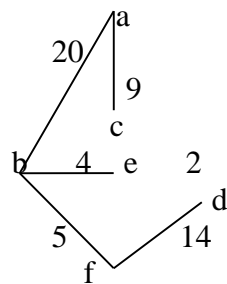
$$\text{matrix is defined as } v_{i,j} = \begin{cases} \text{weight of edge if } e \in (v_i, v_j) \\ 0, \text{ if } e \notin (v_i, v_j) \end{cases}$$

3. Select the maximum element (except 0's) in first row and label that edge is 1
4. Select the maximum element in the second row which is not labeled in the first row and label it as 2. And continue the process until finishing all rows of the matrix.
5. One of the labeled edge forms the cycle of the graph so ignore that edge and then draw the graph which is maximum spanning tree.

Remark: Algorithm for determination of maximal spanning tree is fails for some graphs. Evidence is given below by an example.



	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	
<i>a</i>	0	20	9	13	0	0	] <i>ab</i> edge1 <i>bf</i> edge2 <i>ca</i> edge3 <i>df</i> edge4 <i>eb</i> edge5
<i>b</i>	20	0	1	0	4	5	
<i>c</i>	9	1	0	2	0	0	
<i>d</i>	13	0	2	0	3	14	
<i>e</i>	0	4	0	3	0	0	
<i>f</i>	0	5	0	14	0	0	



According to algorithm draw the graph with all the labeled edges which is shown in the last figure. This is the maximal spanning tree and its total weight is  $(20 + 9 + 5 + 4 + 14) = 52$ . But maximal Spanning tree is  $(20 + 9 + 13 + 4 + 14) = 60$ . Hence algorithm fails to give maximal spanning tree.

REFERENCES:

- [1] CHAZELLE, B. 1997. A faster deterministic algorithm for minimum spanning trees. In Proceedings of the IEEE Symposium on Foundations of Computing Science. IEEE Computer Society Press, Los Alamitos, Calif., 22–31.
- [2] GABOW, H. N., GALIL, Z., SPENCER, T., AND TARJAN, R. E. 1986. Efficient algorithms for finding minimum spanning trees in undirected and directed graphs. *Combinatorica* 6, 109–122.
- [3] GRAHAM, R. L., AND HELL, P. 1985. On the history of the minimum spanning tree problem. *Ann. Hist. Comput.* 7, 43–57.
- [4] KING, V. 1997. A simpler minimum spanning tree verification algorithm. *Algorithmica* 18, 2, 263–270.