Discrete Structures

Lecture # 16

Dr. Muhammad Ahmad

Department of Computer Science

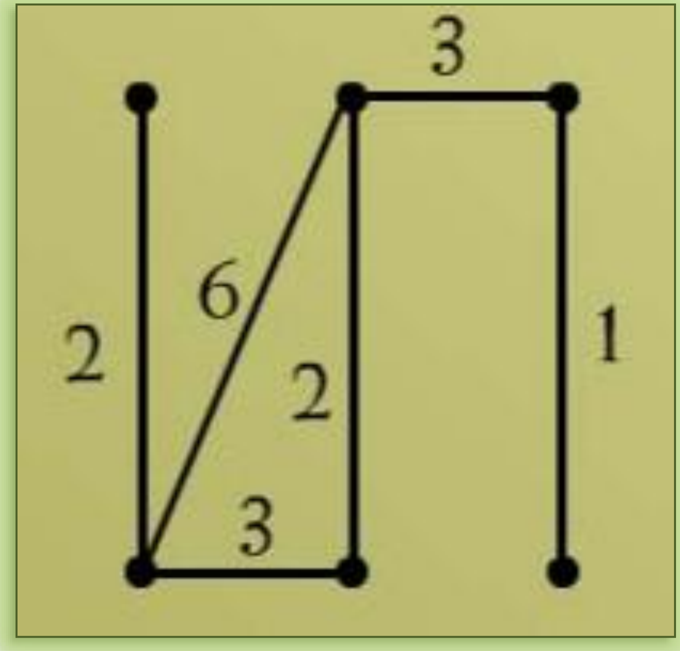
FAST -- National University of Computer and Emerging Sciences. CFD Campus

WEIGHTED GRAPH

A weighted graph is a graph for which each edge has an associated real number weight.

The sum of the weights of all the edges is the total weight of the graph.

EXAMPLE



Total weight of the graph is

$$2 + 6 + 3 + 2 + 3 + 1 = 17$$

MINIMAL SPANNING TREE

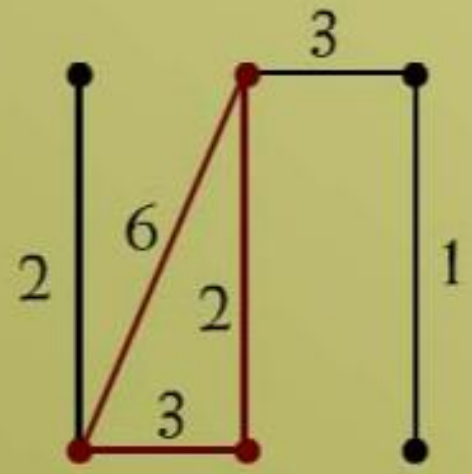
A minimal spanning tree for a weighted graph is a spanning tree that has the least possible total weight compared to all other spanning trees of the graph.

Connected

If G is a weighted graph and e is an edge of G then w(e) denotes the weight of e and w(G) denotes the total weight of G.

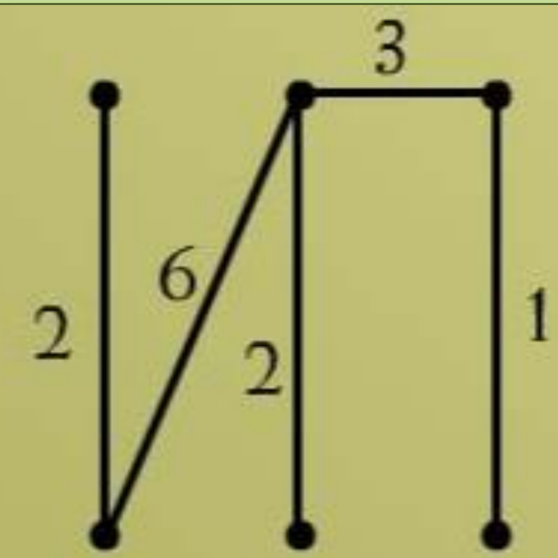
EXAMPLE

Find the three spanning trees of the weighted graph below. Also indicate the minimal spanning tree.



We can remove edge of weight 6,2,3.

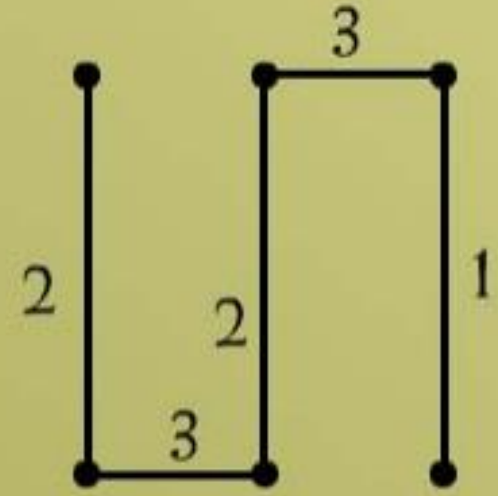
EXAMPLE



T_1

By removing edge of weight 3.

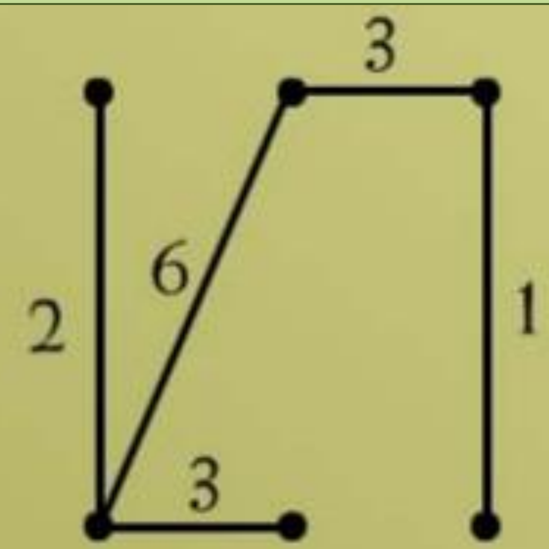
EXAMPLE



T_3

By removing edge of weight 6.

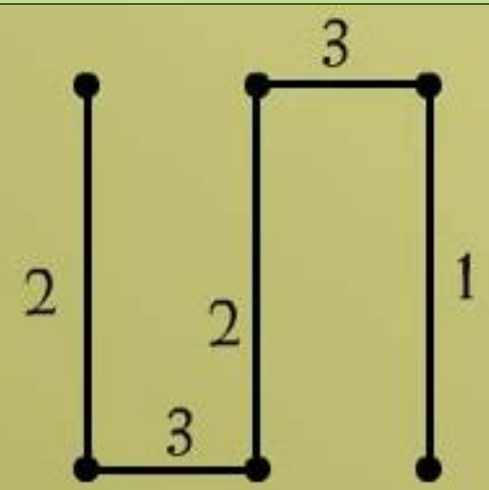
EXAMPLE



T_3

By removing edge of weight 2.

EXAMPLE



T_3

Weight of graph by removing the edge of weight 6 is the minimum Spanning tree. Because its weight is 11.

KARUSKAL'S ALGORITHM

Input: G [a weighted graph with n vertices]

Algorithm:

1. Initialize T (the minimal spanning tree of G) to have all the vertices of G and no edges.

2. Let E be the set of all edges of G and let m: = 0.

KARUSKAL'S ALGORITHM

Spanning tree will have n-1 edges, because the graph has n vertices.

While (m < n - 1)

3a. Find an edge e in E of least weight.

3b. Delete e from E.

3c. If addition of e to the edge set of T does not produce a circuit then add e to the edge set of T and set m: = m + 1

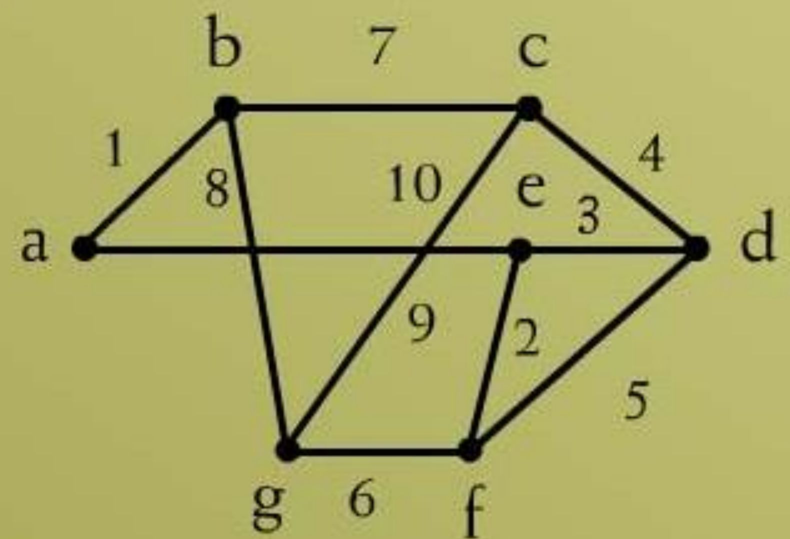
end while

KARUSKAL'S ALGORITHM

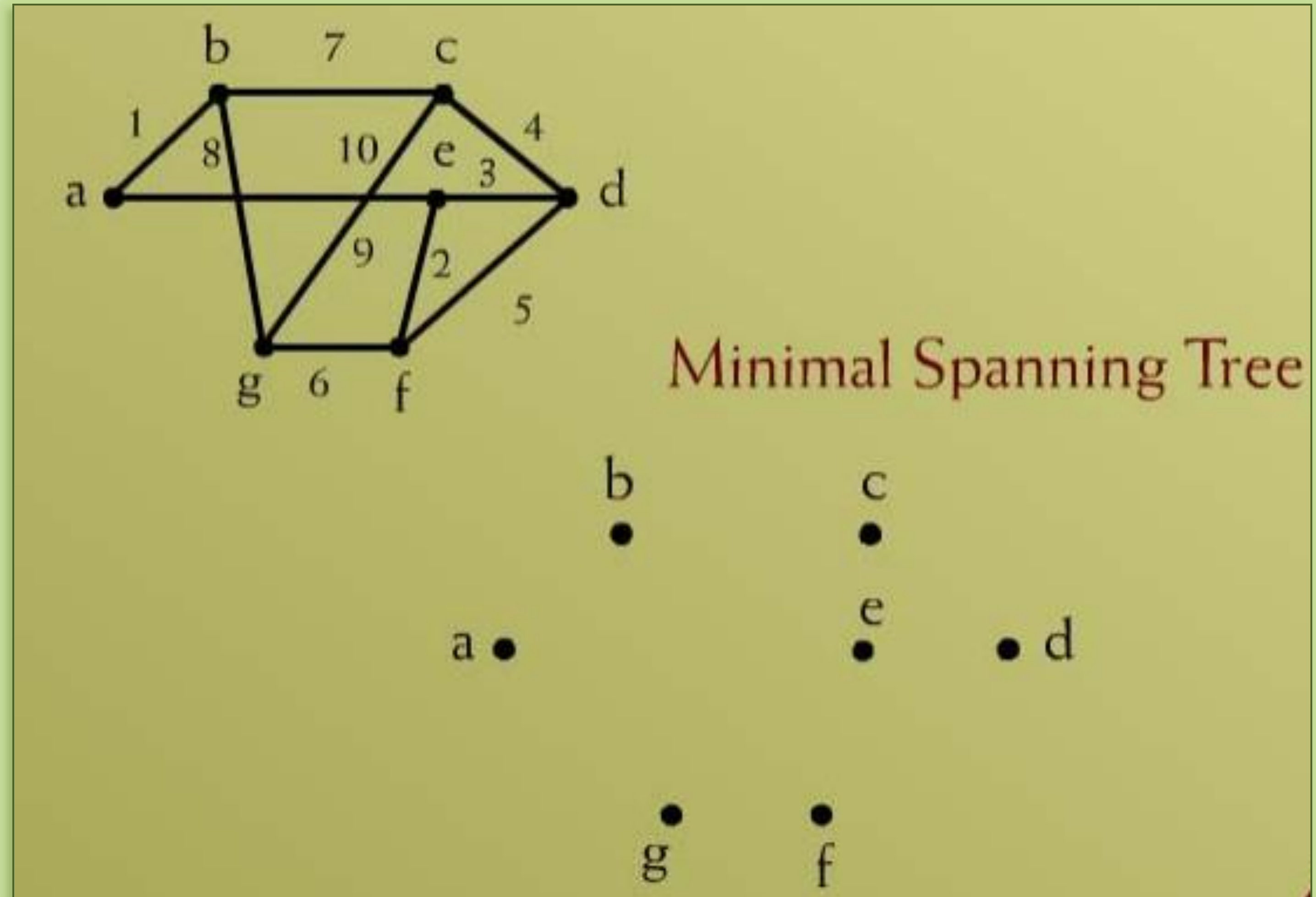
Output T
end Algorithm

EXAMPLE

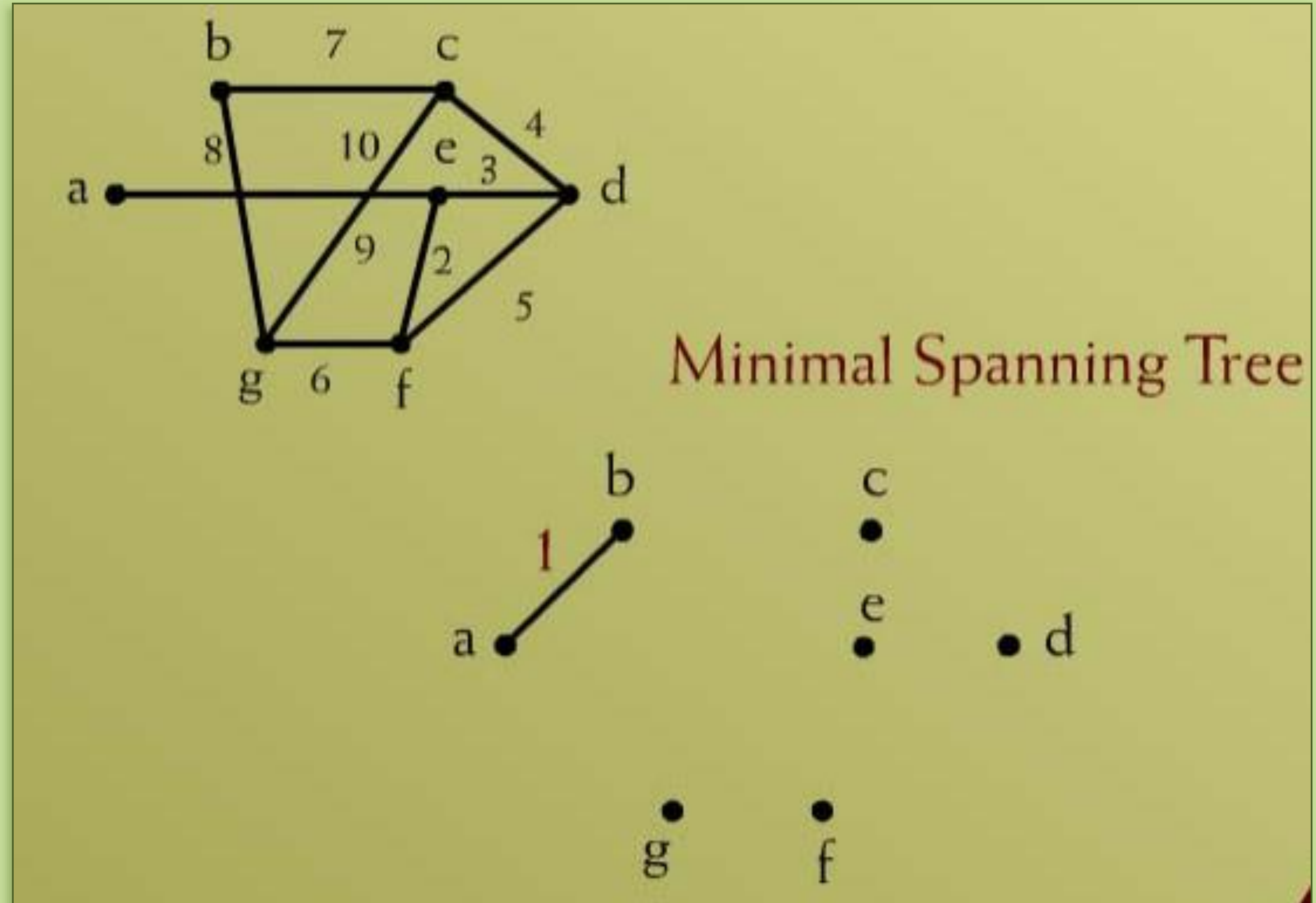
Use Kruskal's algorithm to find a minimal spanning tree for the graph below. Indicate the order in which edges are added to form the tree.



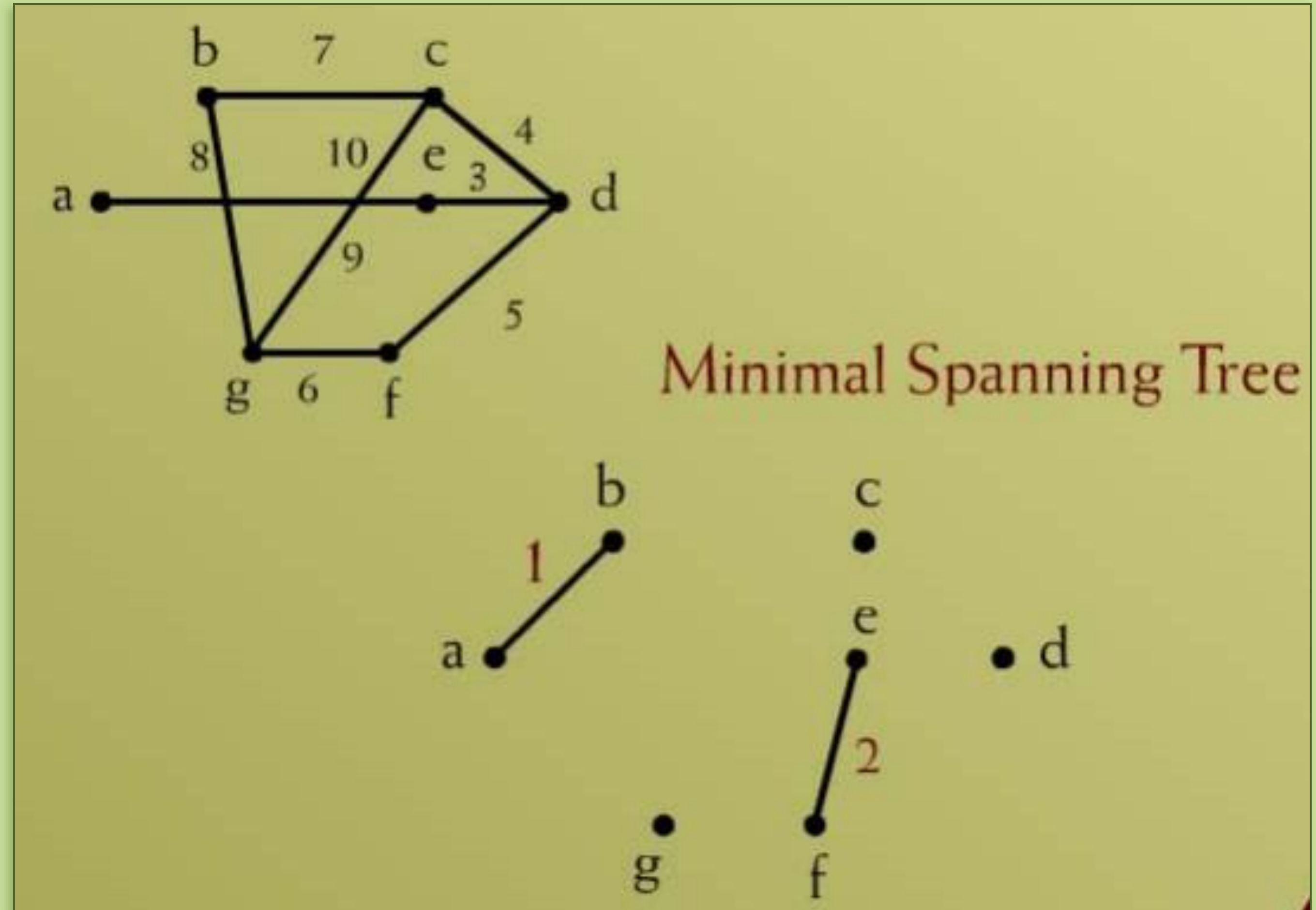
SOLUTION



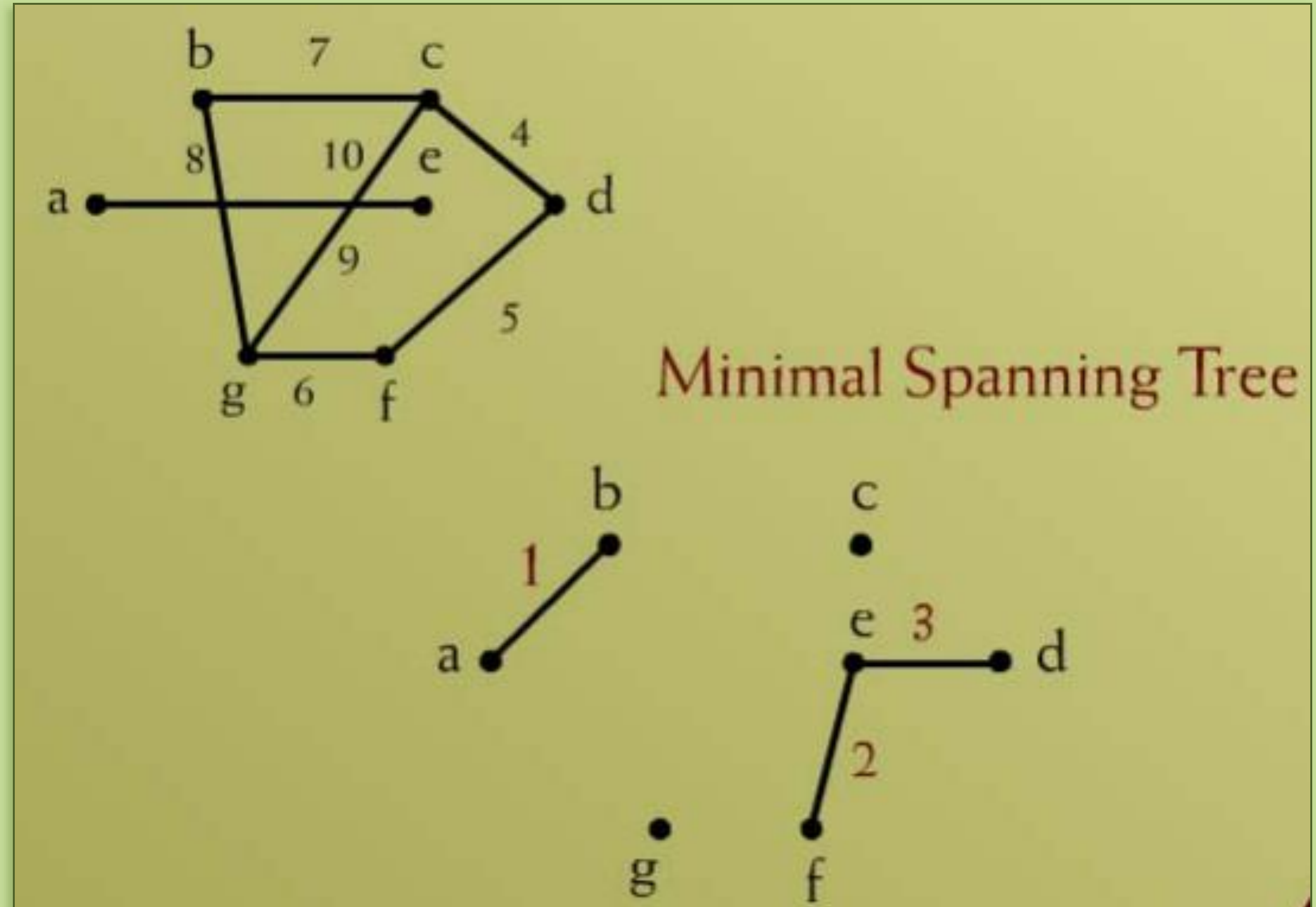
SOLUTION



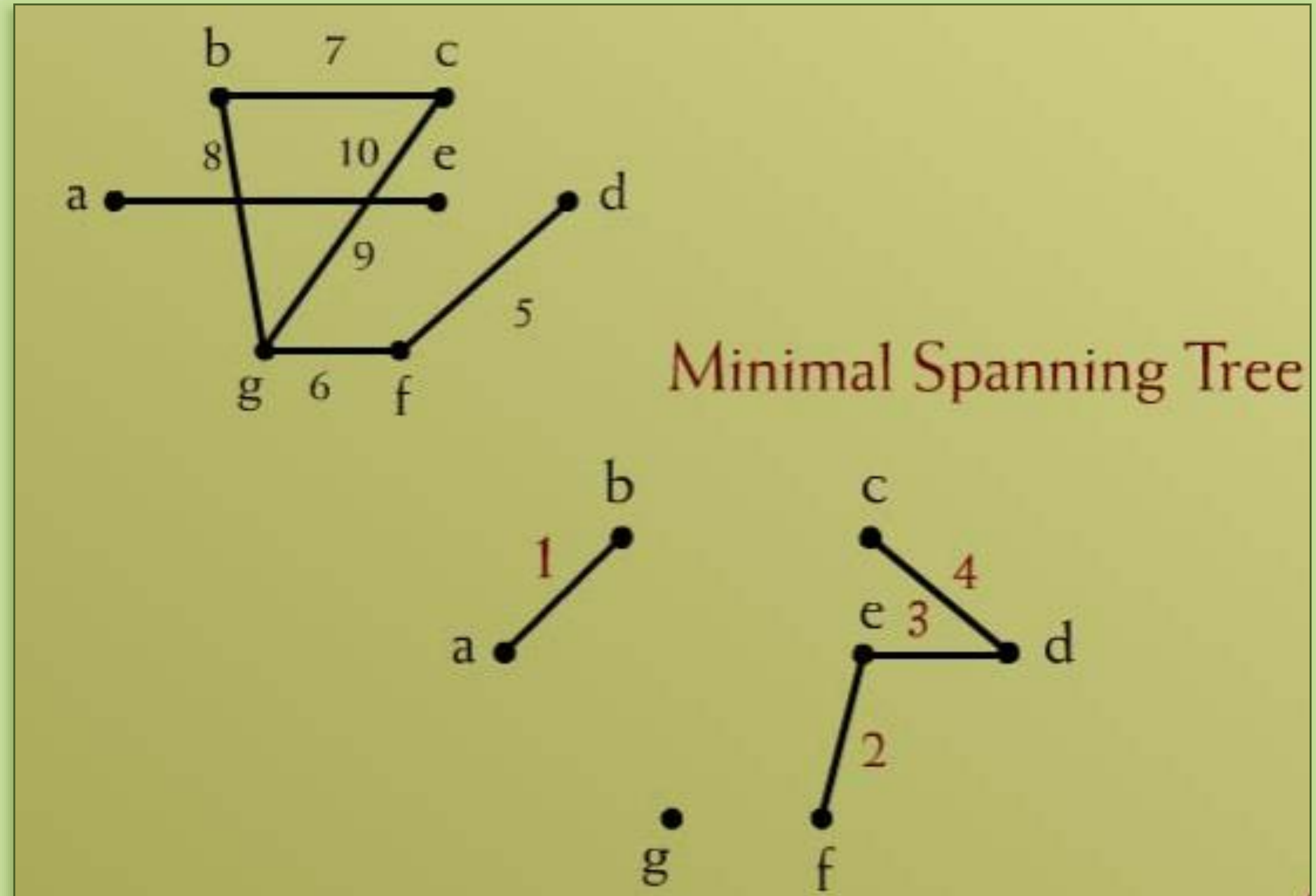
SOLUTION



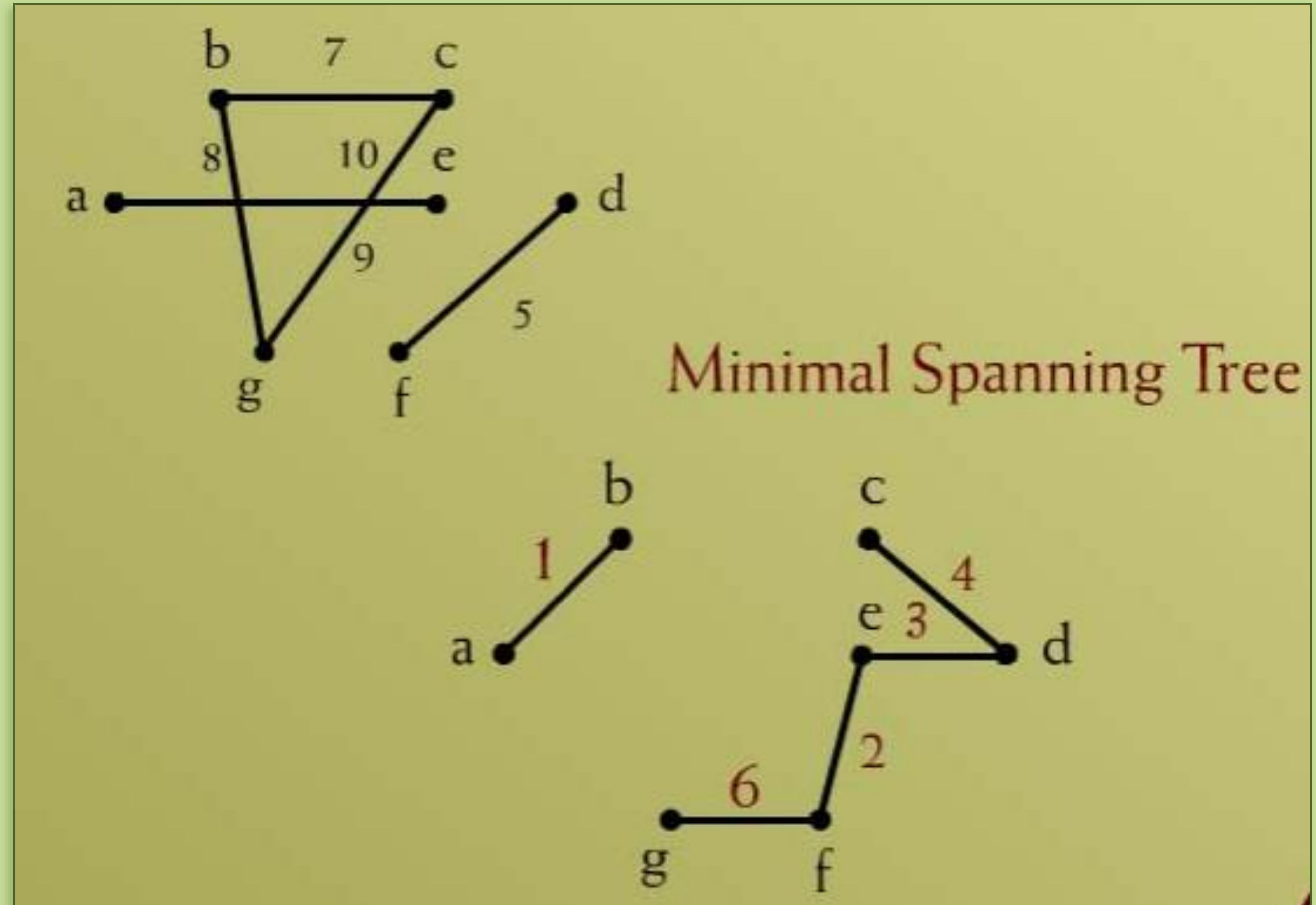
SOLUTION



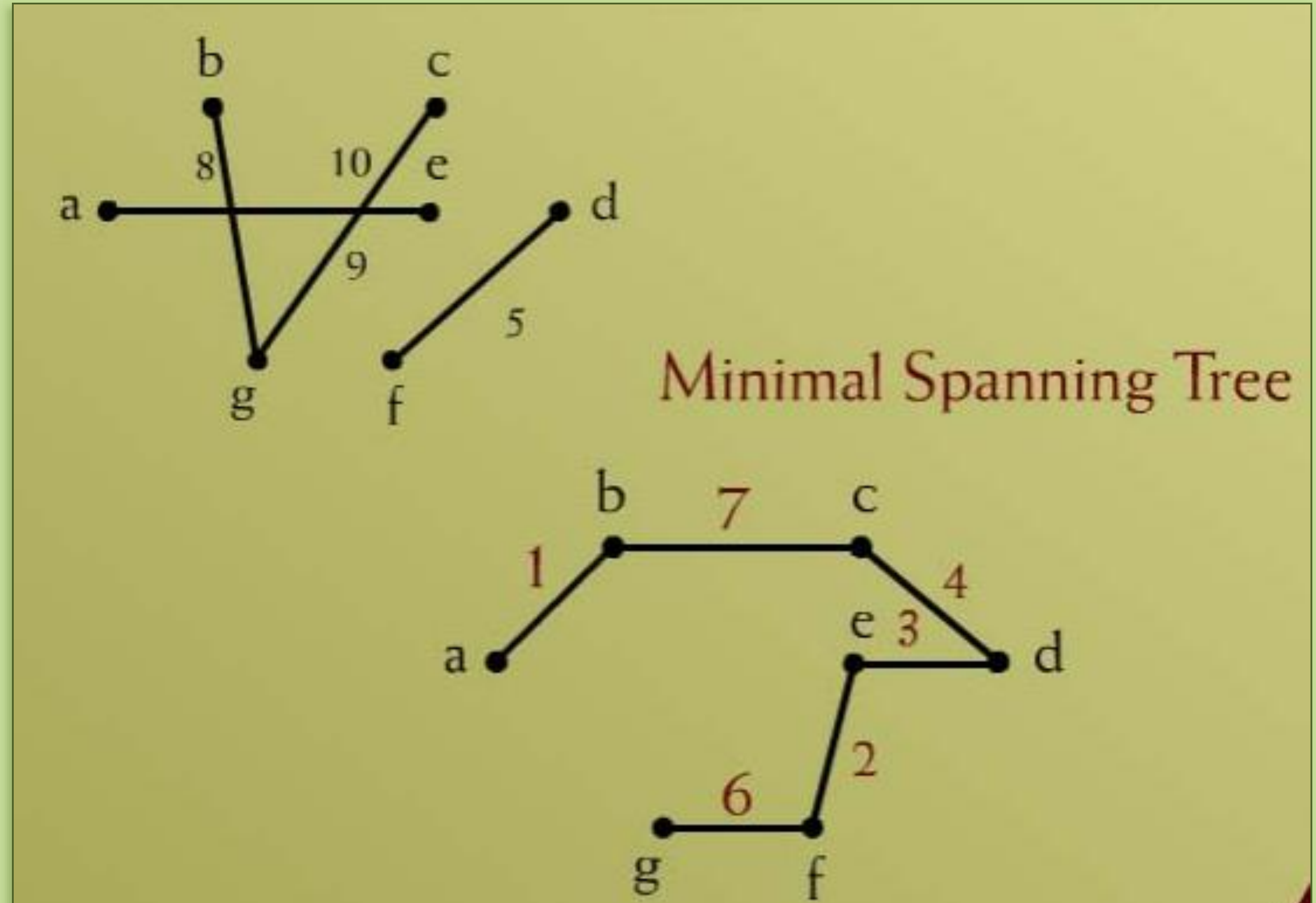
SOLUTION



SOLUTION

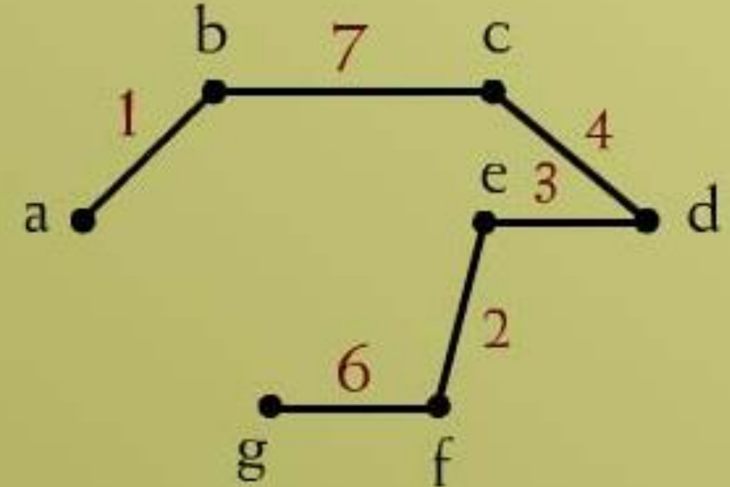


SOLUTION



SOLUTION

Minimal Spanning Tree



Order of adding the edges:

{a,b}, {e,f}, {e,d}, {c,d}, {g,f}, {b,c}

PRIM'S ALGORITHM

Input: G[a weighted graph with n vertices]

Algorithm Body :

1. Pick a vertex v of G and let T be the graph with one vertex v and no edges.

2. Let V be the set of all vertices of G except v.

PRIM'S ALGORITHM

for i: = 1 to n - 1

3a. Find an edge "e" of G such that

(1)"e" connects T to one of the vertices in V

(2)"e" has the least weight of all edges connecting T to a vertex in V.

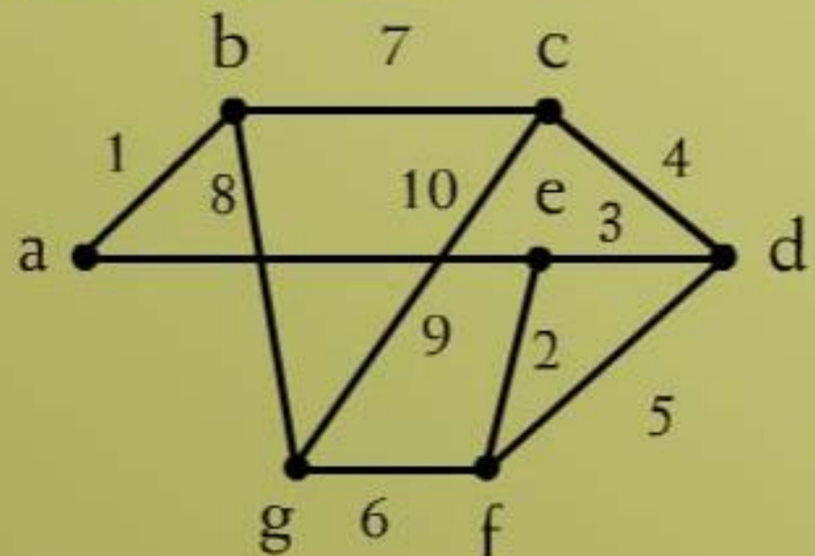
Let "w" be the end point of "e" that is in V.

3b. Add "e" and "w" to the edge and vertex sets of T and delete "w" from V.

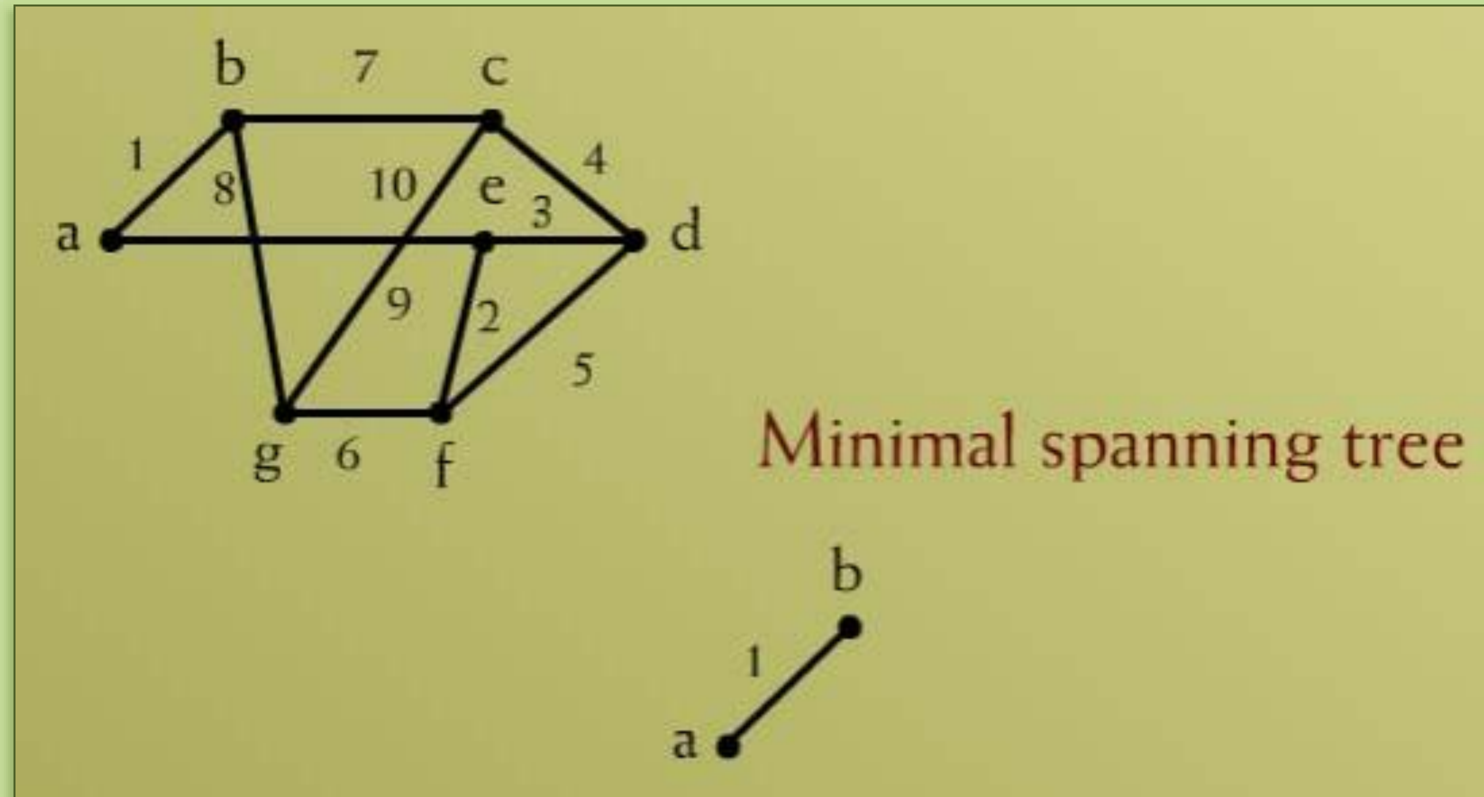
next i

EXAMPLE

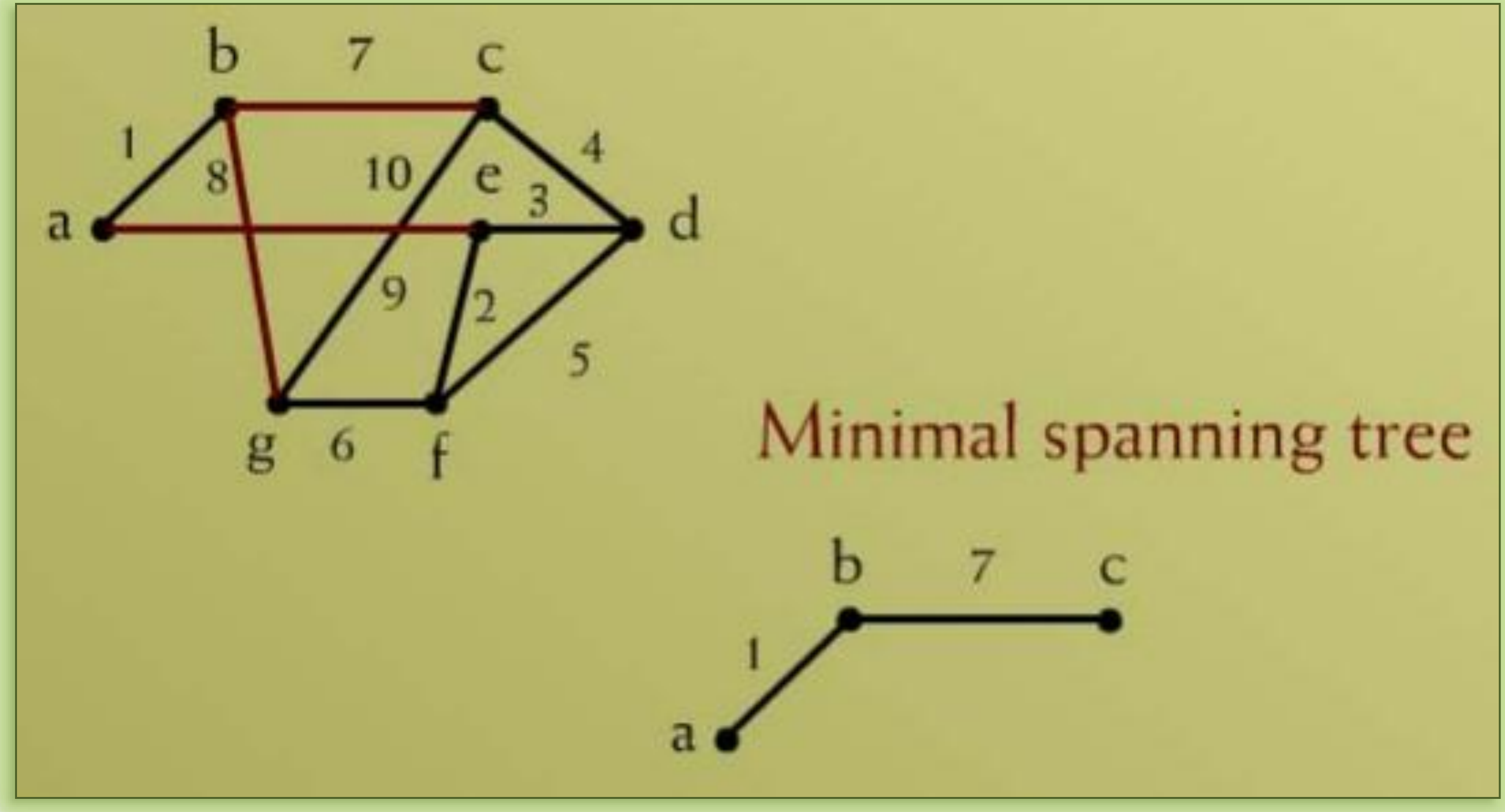
Use Prim's algorithm starting with vertex a to find a minimal spanning tree of the graph below. Indicate the order in which edges are added to form the tree.



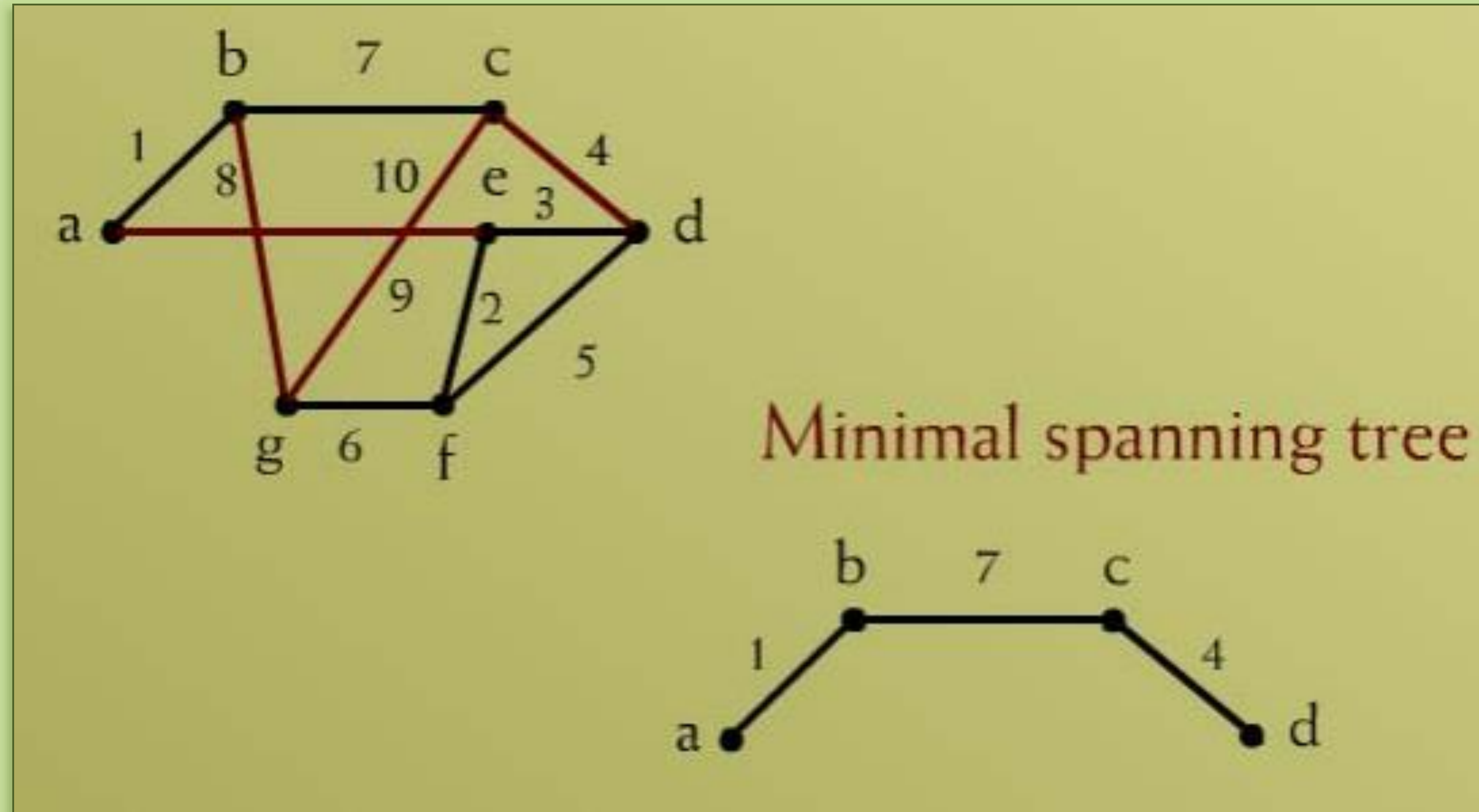
SOLUTION



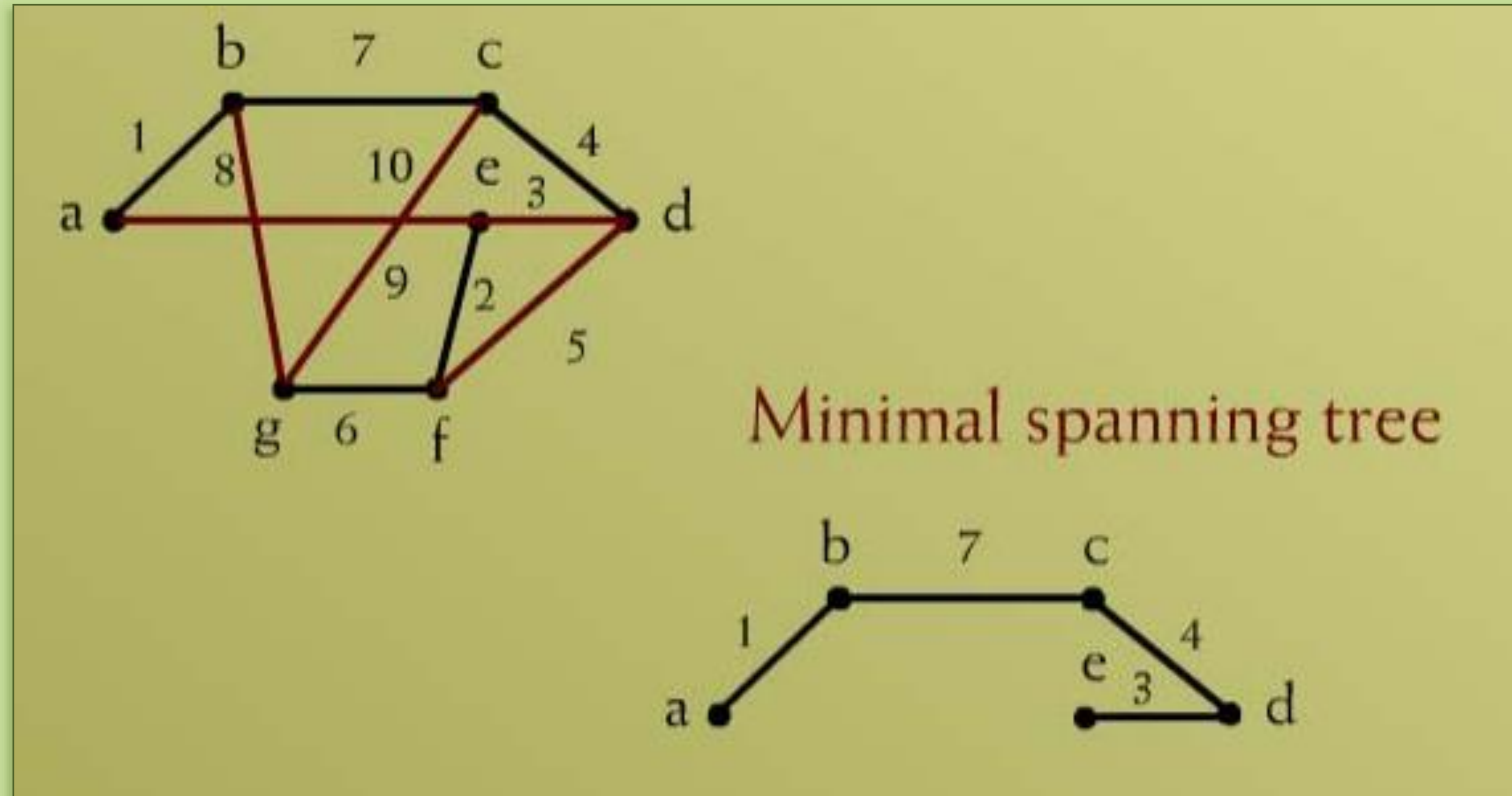
SOLUTION



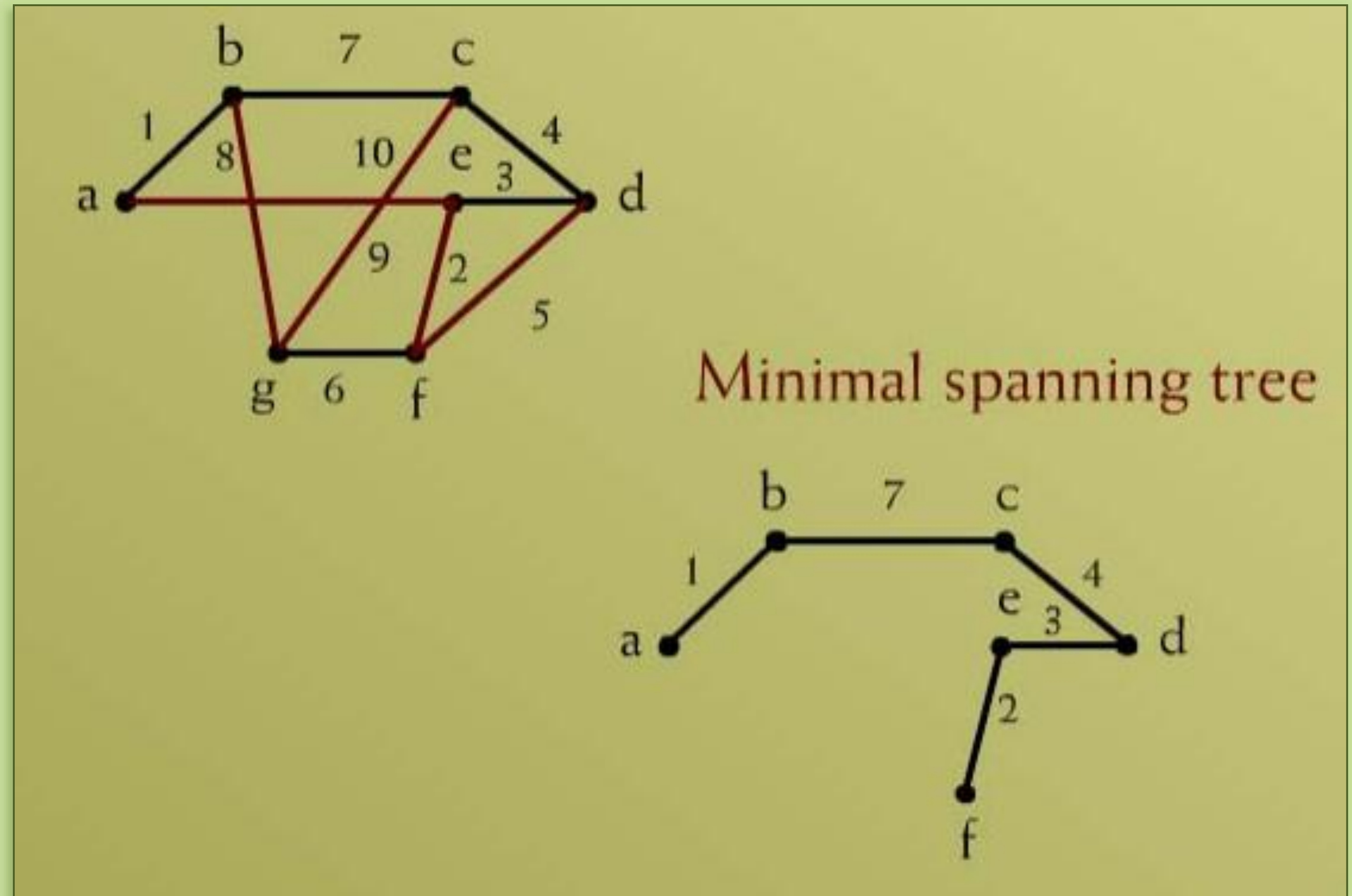
SOLUTION



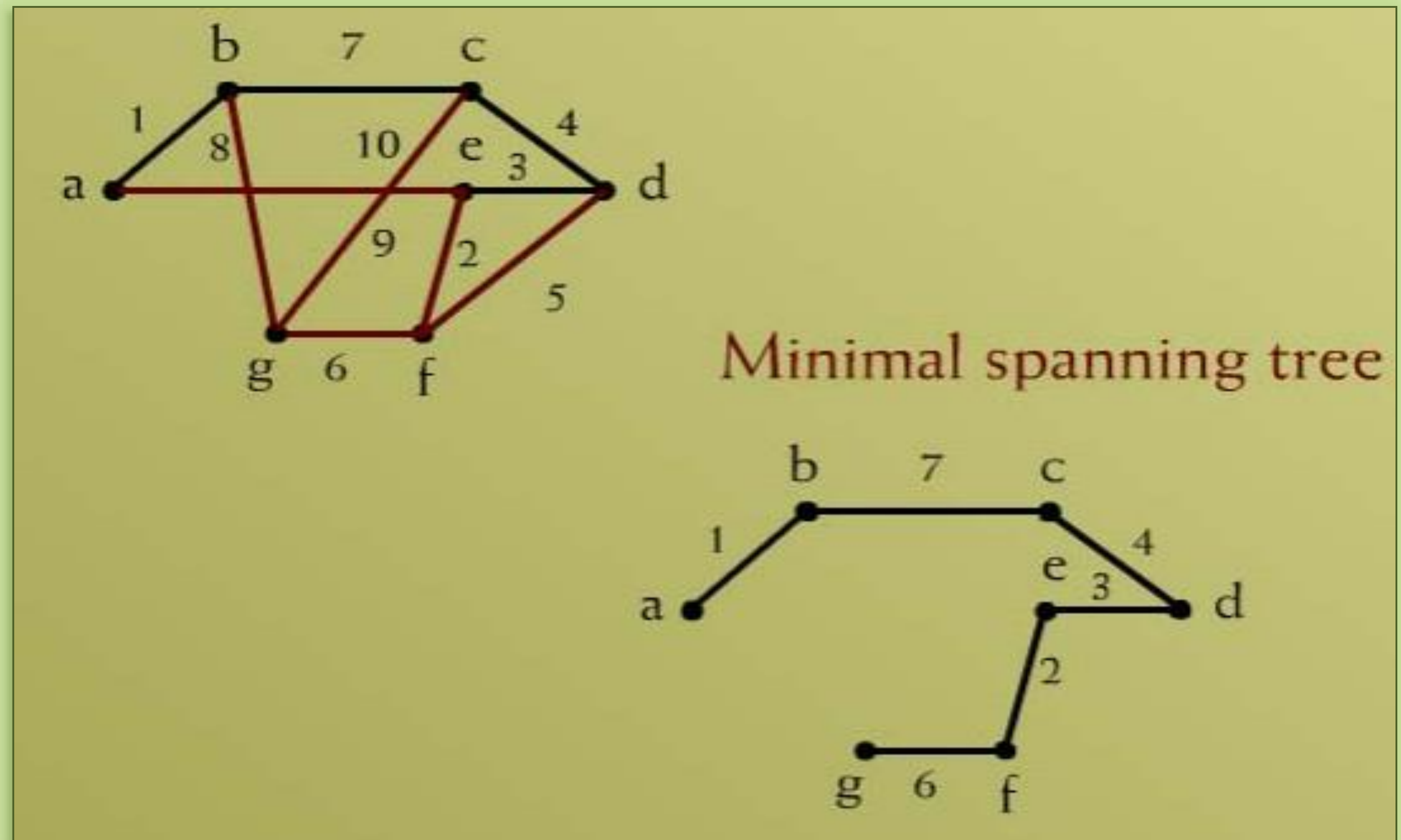
SOLUTION



SOLUTION



SOLUTION



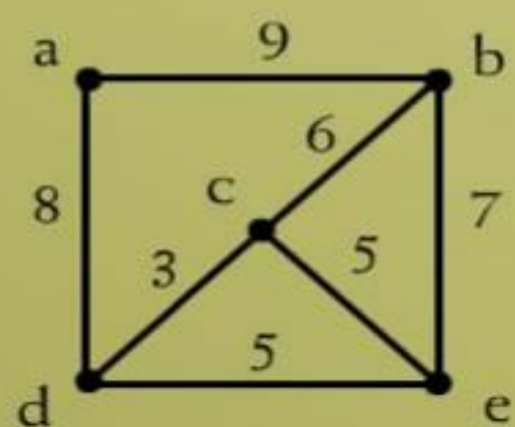
b 7 c
1 8 10 e 3 4
a d
9 2 5
g 6 f
Minimal spanning tree
b 7 c
1 e 3 4
a d
2
g 6 f

EXAMPLE

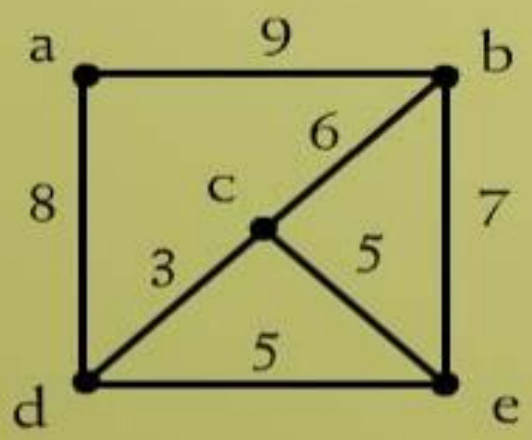
Find all minimal spanning trees that can be obtained using

(a) Kruskal's algorithm.

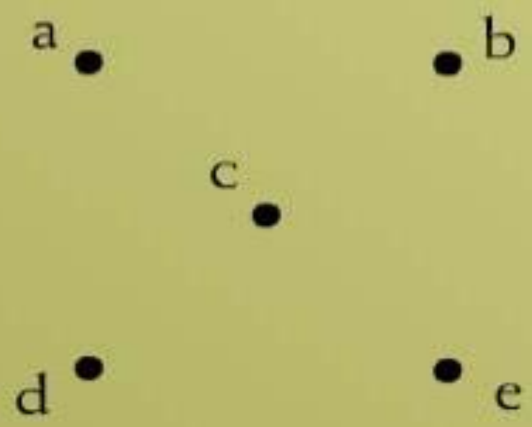
(b) Prim's algorithm starting with vertex a.



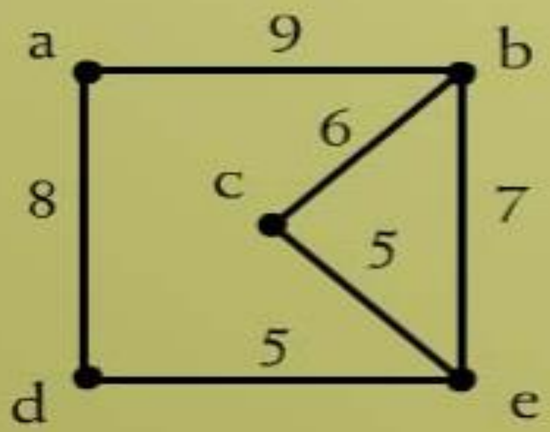
SOLUTION



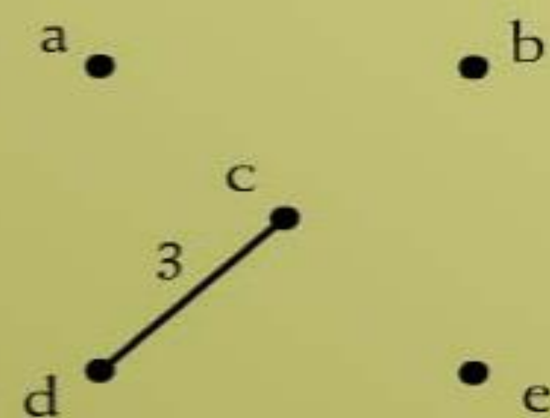
Minimal Spanning Tree



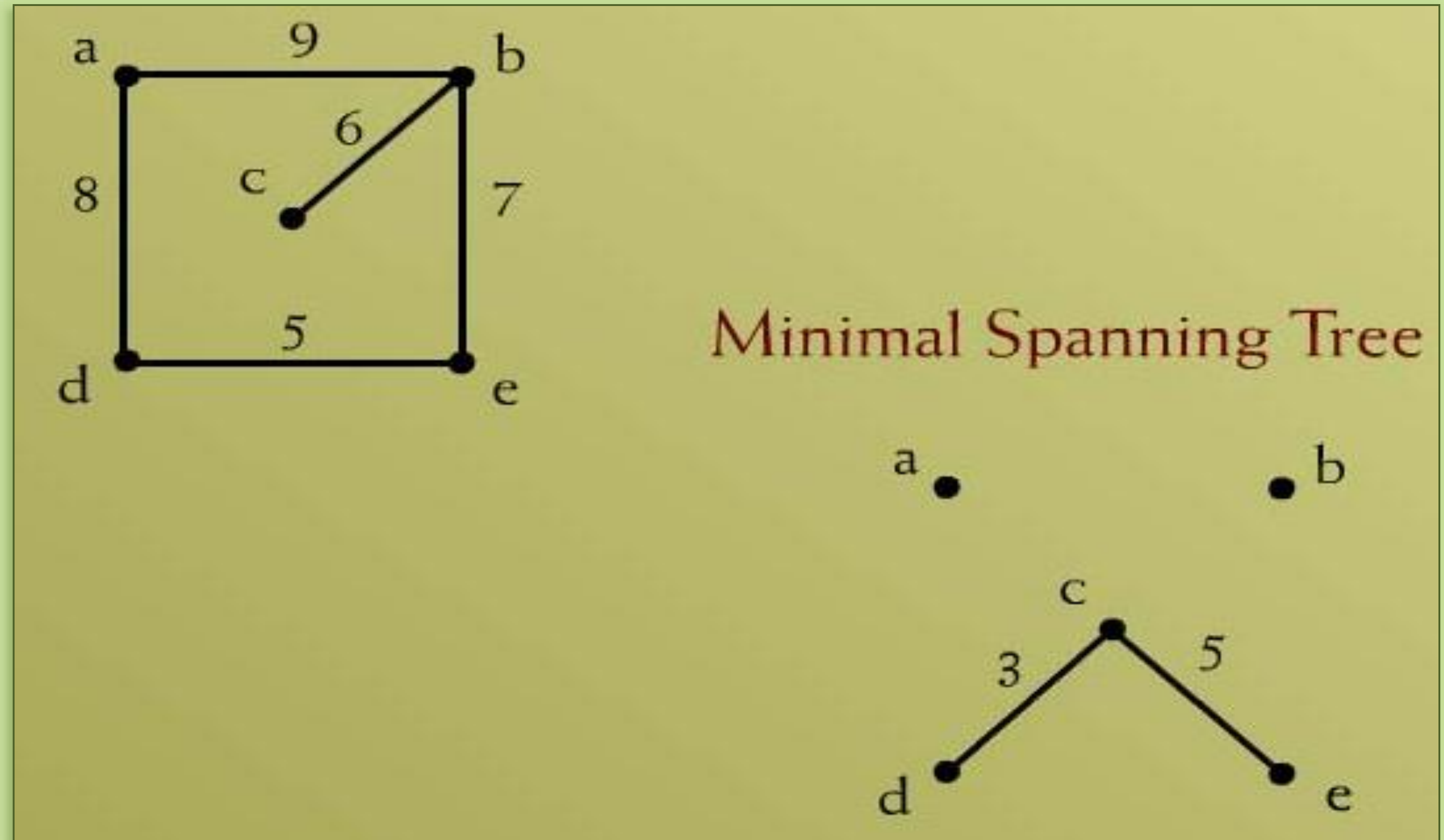
SOLUTION



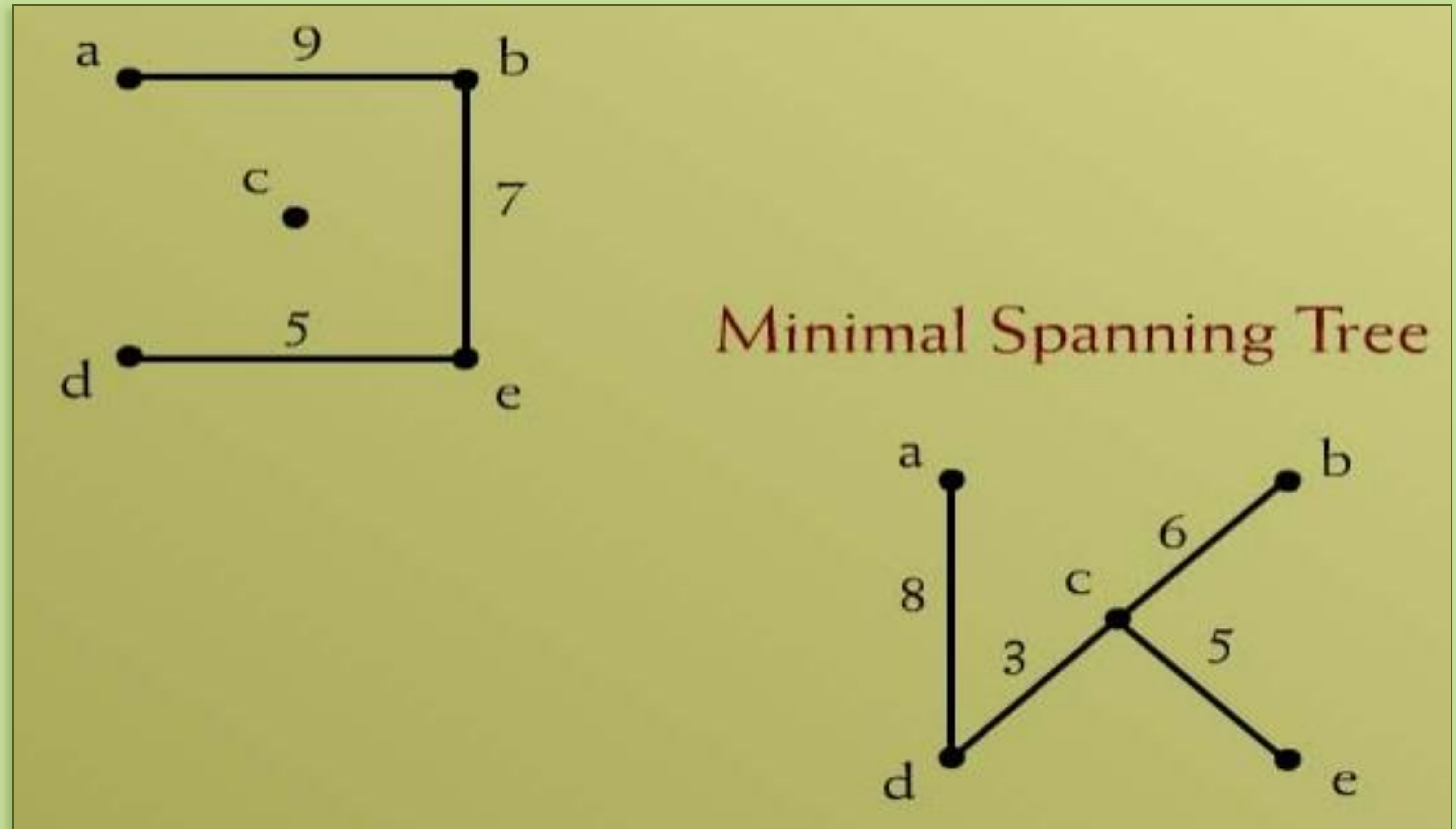
Minimal Spanning Tree



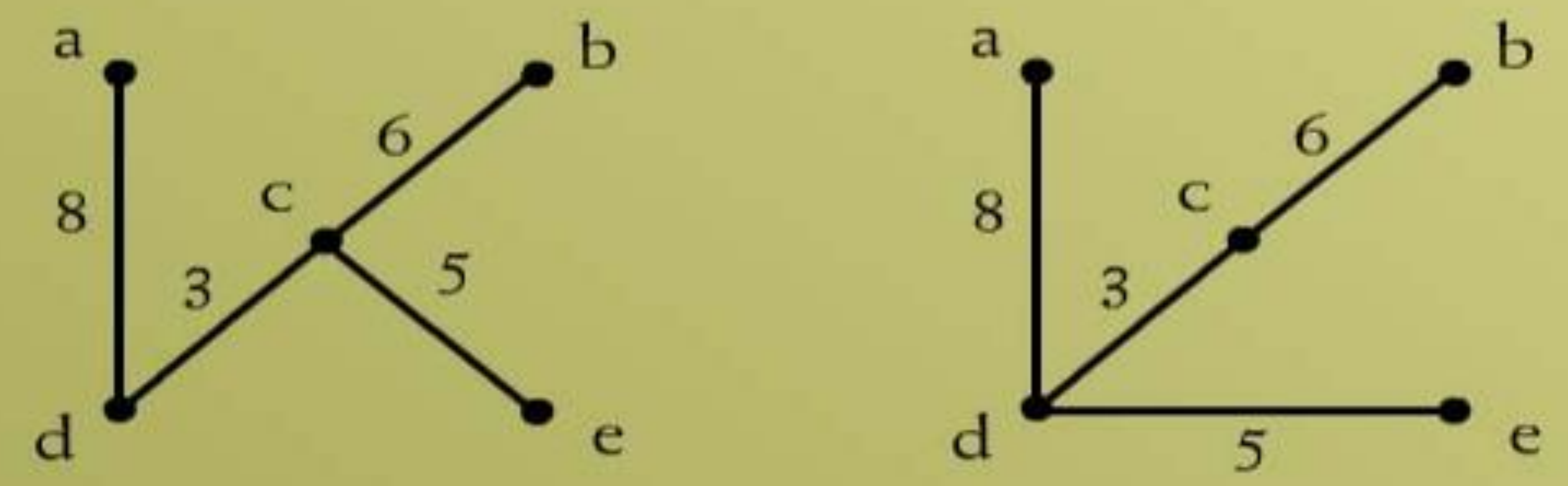
SOLUTION



SOLUTION

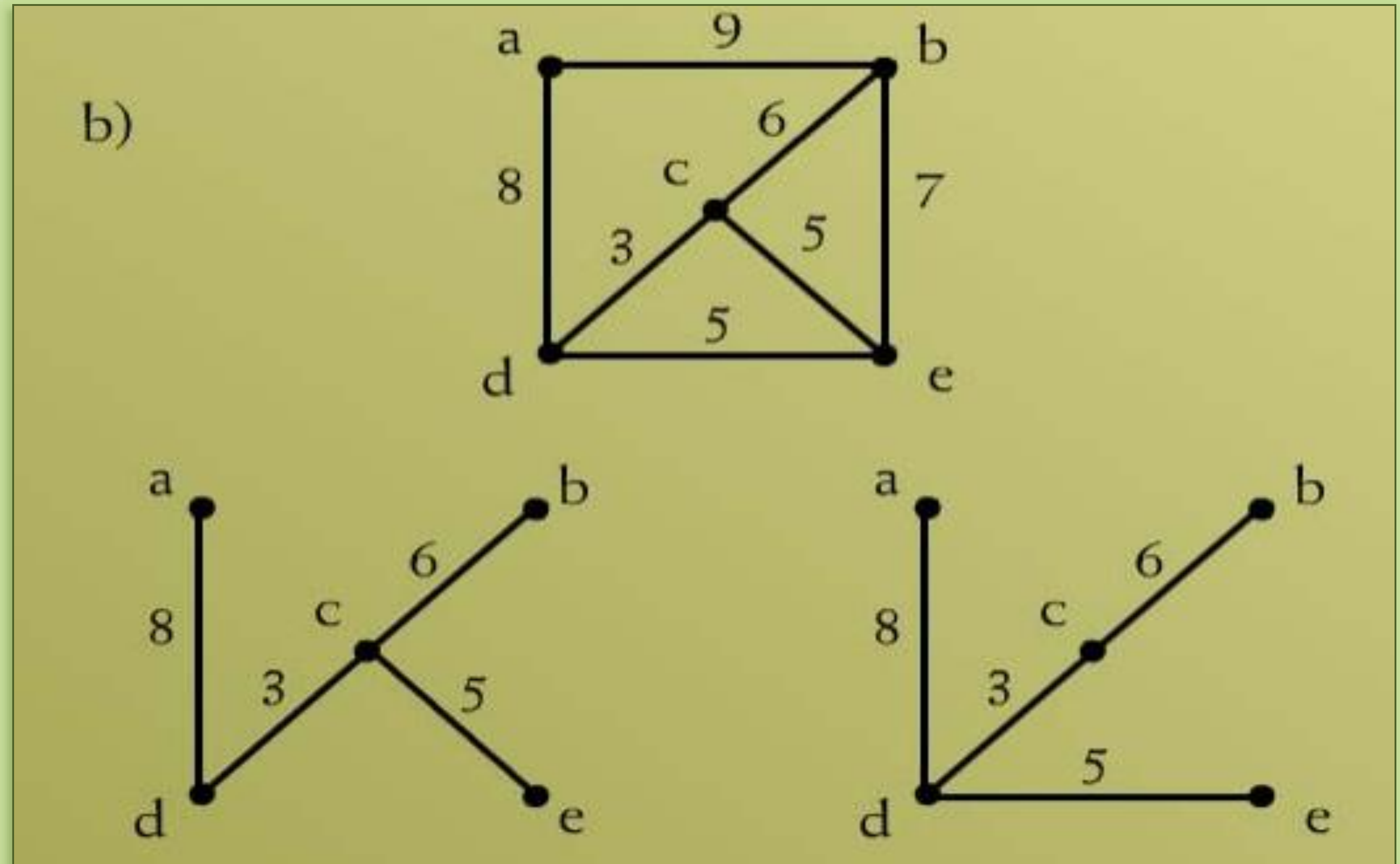


SOLUTION



Minimal Spanning trees

SOLUTION



Important Notes

1. If the edge weights in your graph are all different from each other, then your graph has a unique minimum spanning tree, so Kruskal's and Prim's algorithms are guaranteed to return the same tree.

2. Else, you can have different minimum spanning trees as well. However, all MSTs will have same (least) weight.