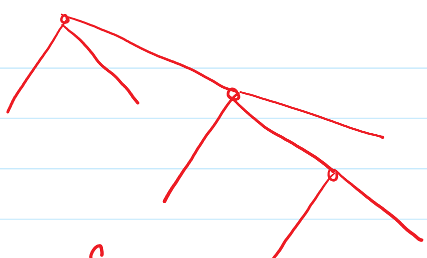
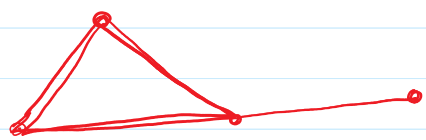
Tuesday, November 2, 2021 12:39 PM

- Introduction to Trees
- Definition of a Tree graph
- Theorem
- Rooted Tree
- m-*ary* tree
- Quiz

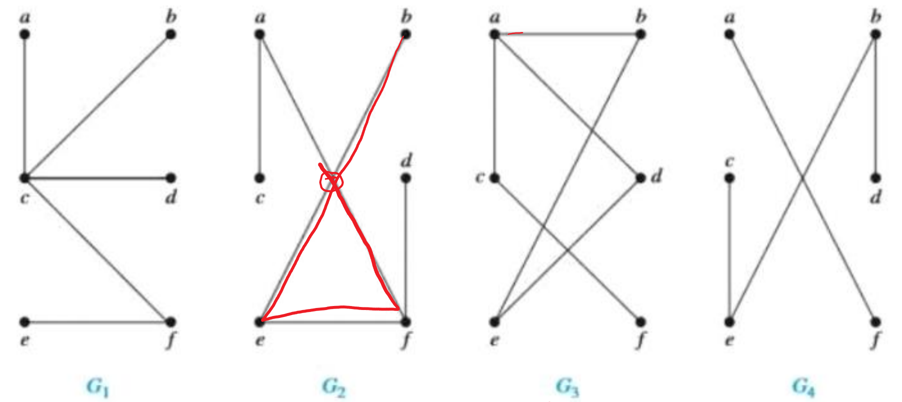
A connected graph that contains no simple circuits is called a tree.



G



Which of the following graphs are trees?

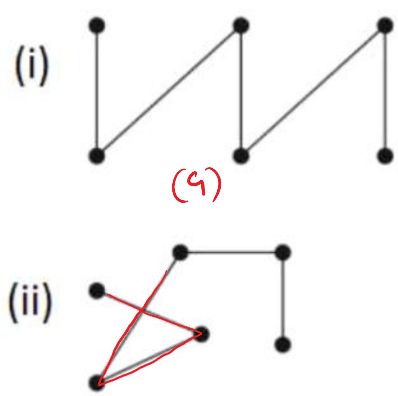


Tree Tree

Examples of trees and graphs that are not trees.

G_1 (Tree)

G_2 (Tree.)



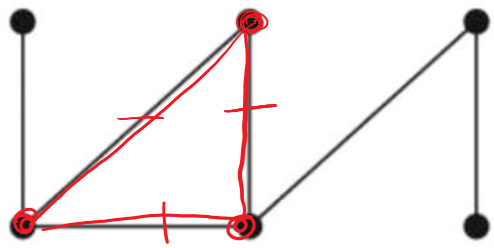
Which are trees?

A. only (i)

B. only (ii)

C. Both (i) and (ii)

D. Neither (i) nor (ii)



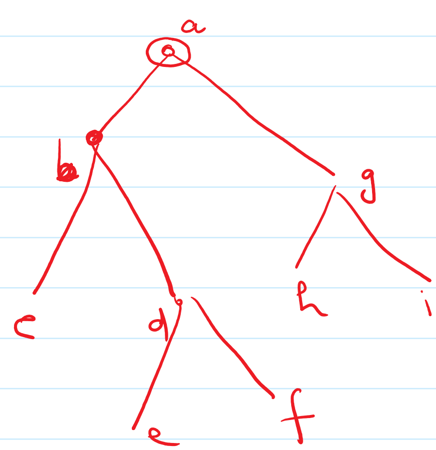
Is this graph a tree?

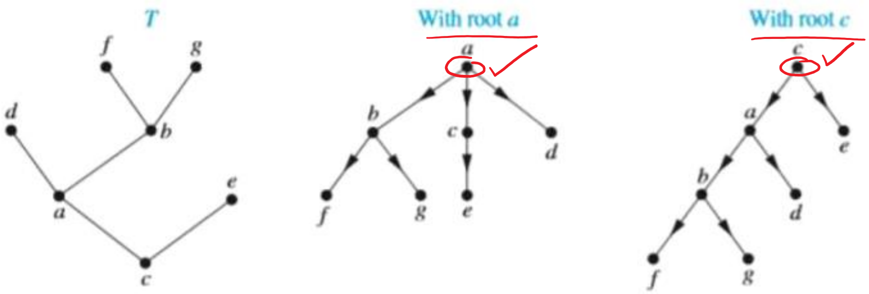
YES NO

As this Graph contains of circuit of length 3 ∴ It is not a tree.

Rooted tree.

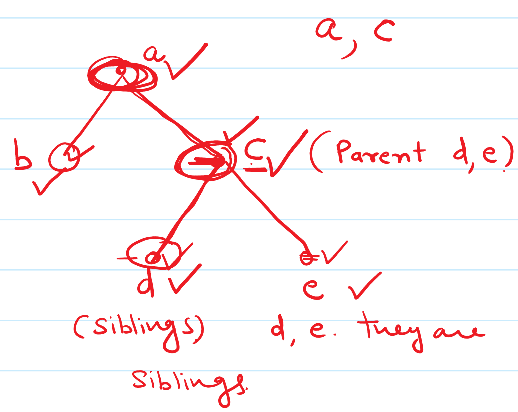
A *rooted tree* is a tree in which one vertex has been designated as the root and every edge is directed away from the root.



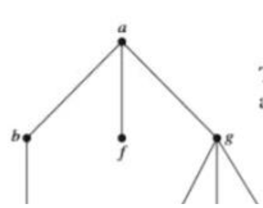


A tree and rooted trees formed by designating two different roots.

- If v is a vertex in T other than the root, the **parent** of v is the unique vertex u such that there is a directed edge from u to v. When u is a parent of v, v is called a child of u.
- Vertices with the same parent are called **siblings**.
- The **ancestors** of a vertex other than the root are the vertices in the path from the root to its vertex, excluding the vertex itself and including the root. The **descendants** of a vertex v are those vertices that have v as an ancestor.
- A vertex of a rooted tree is called a leaf if it has no children. Vertices that have children are called **internal vertices**.
- The root is an internal vertex unless it is the only vertex in the graph, in which case it is a leaf.

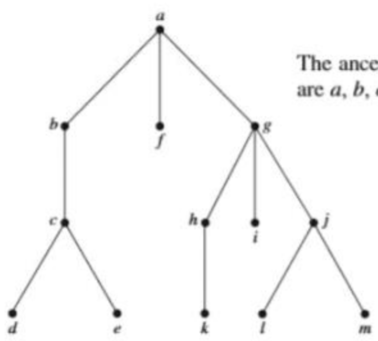


Internal vertices: Those vertices who have children in the rooted tree are called internal vertices.



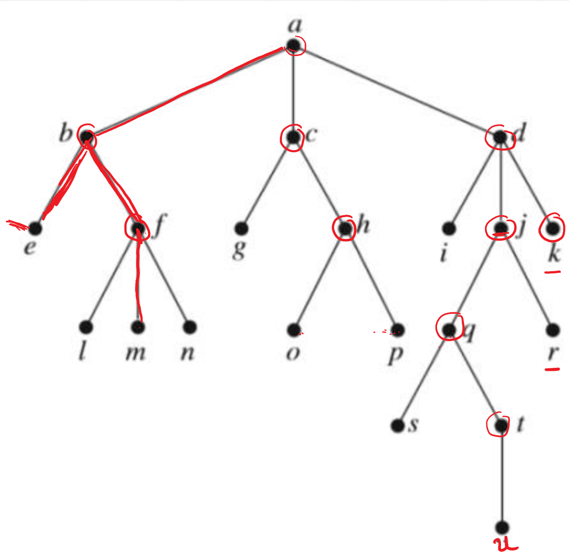
The parent of c is b. The children of g are h, i, and j. The siblings of h are i and j. The ancestors of e are c, b, and a. The descendants of b are c, d, and e. The internal vertices are a, b, c, g, h, and j. The leaves are d, e, f, i, k, l, and m.

rooted tree are called internal vertices.



A rooted tree T.

The parent of c is b. The children of g are h, i, and j. The siblings of h are i and j. The ancestors of e are c, b, and a. The descendants of b are c, d, and e. The internal vertices are a, b, c, g, h, and j. The leaves are d, e, f, i, k, l, and m.



1. What is the ~~vertex~~ root of the rooted tree? (a)
2. Which vertices are internal? a, b, f, c, h, d, j, q, t
3. Which vertices are leaves? e, l, m, n, g, o, p, s, u, r, k. i,
4. Which vertices are children of j? q, r.
5. Which vertex is the parent of k? (d)
6. Which vertices are siblings of o? (p)
7. Which vertices are ancestors of m? f, b, a
8. Which vertices are descendants of b? f, l, m, n, e

Subtree

If a is a vertex in a tree, the **subtree** with a as its root is the subgraph of the tree consisiting of a and its descendants and all edges incident to these descendants.

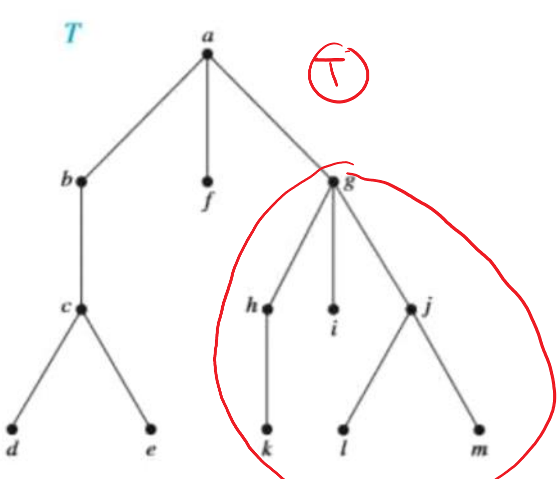


FIGURE 5 A rooted tree T.

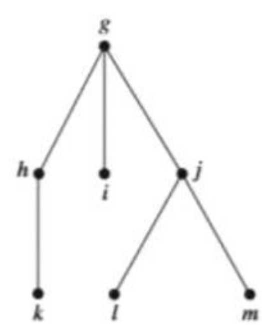
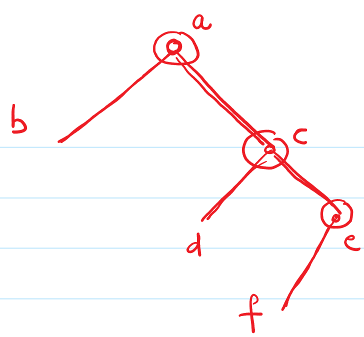
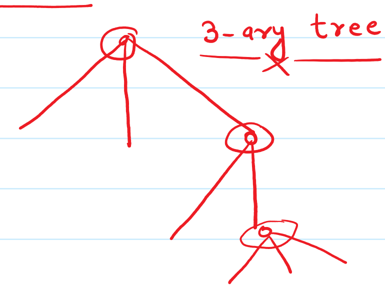


FIGURE 6 The subtree rooted at g.

A rooted tree is called an *m-ary* tree if every internal vertex has no more than m children.

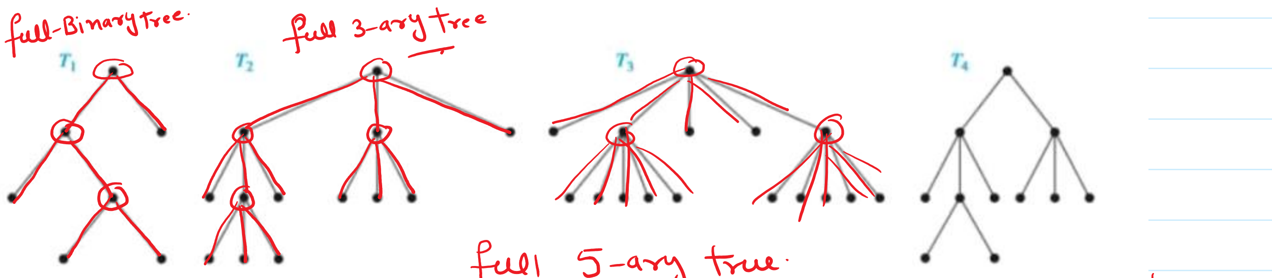


Binary tree.



A rooted tree is called an *m-ary* tree if every internal vertex has no more than m children.

The tree is called a full *m-ary* tree if every internal vertex has exactly m children. An *m-ary* tree with $m = 2$ is called a binary tree.



Four rooted trees.

Are the rooted trees in Figure full m-ary trees for some positive integer m?

Solution: T_1 is a full binary tree because each of its internal vertices has two children. T_2 is a full 3-ary tree because each of its internal vertices has three children. In T_3 each internal vertex has five children, so T_3 is a full 5-ary tree. T_4 is not a full m-ary tree for any m because some of its internal vertices have two children and others have three children.

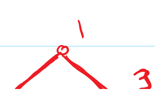
Minimum Spanning tree.

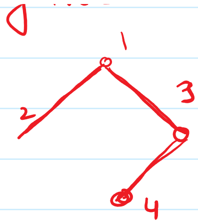
① Krushkal's Algorithms

② Prim's Algorithm.

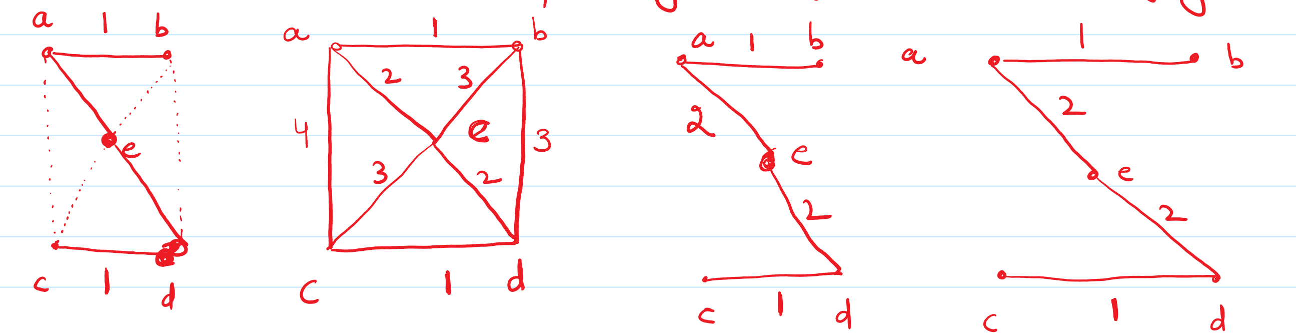
Theorem: If have n vertices in the graph, then only graph with $(n-1)$ edges is called a tree.

① Krushkal's Algorithm for finding Minimum Spanning tree.





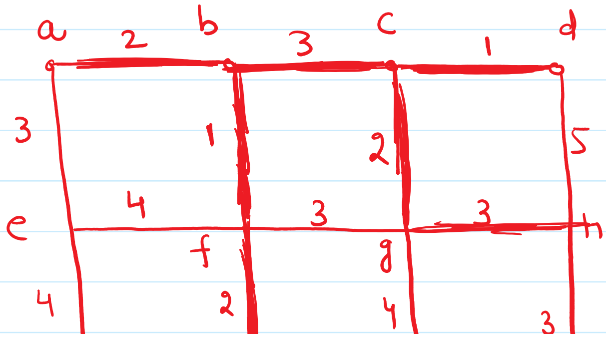
Q Find the minimum Spanning tree from the following graph.

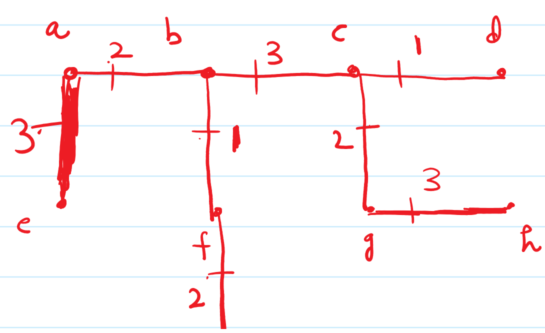


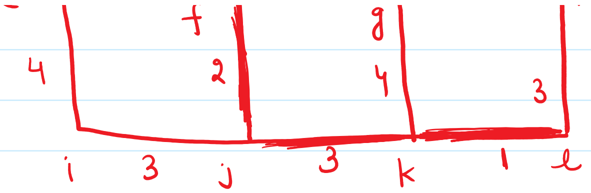
write these edges of the graph in ascending order

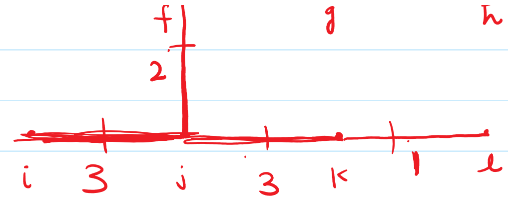
1 < ab ✓, cd ✓ | 2 < ae ✓, ed ✓ | 3 < eb ×, bd ×, ce × | 4 — ac × | weight of MST = 6.

Q Find the MST from the following graph.

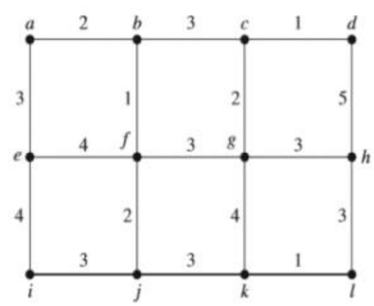






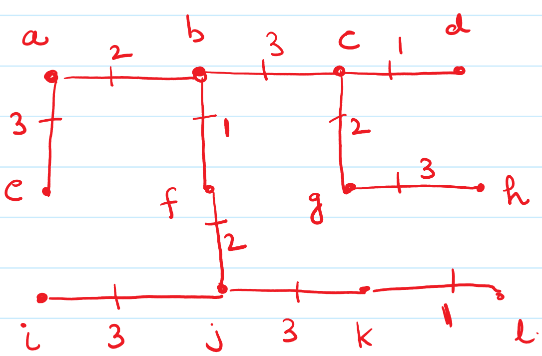


weight of M.S.T = 24



Find a minimum spanning tree for the given weighted graph using Prim's algorithm.

A weighted graph.



cd = 1 ✓

c — g = 2 ✓
c — b = 3 ✓

d → h = 5 ✗

g — f = 3 ✗
g — h = 3 ✓
g — k = 4 ✗

b — a = 2 ✓
b — f = 1 ✓

f — e = 4 ✗
f — j = 2 ✓
f — g = 3 ✗

a — e = 3 ✓

j — i = 3 ✓
j — k = 3 ✓

h — d = 5 ✗
h — l = 3 ✗

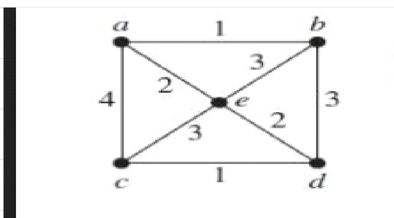
e — f = 4 ✗
e — i = 4 ✗

i — e = 4 ✗

k — g = 4 ✗
k — l = 1 ✓

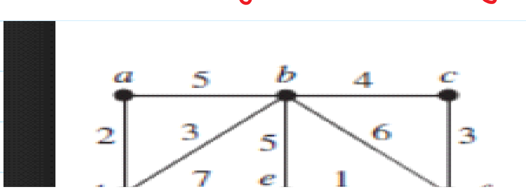
l → h = 3 ✗

weight of M.S.T = 24



a

Use Prim's Algorithm for finding the M.S.T.



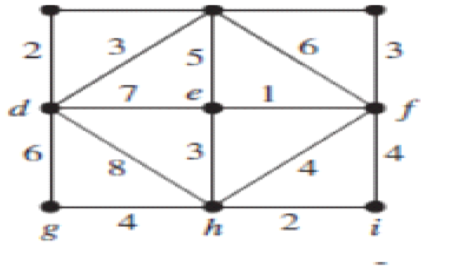
ef = 1 ✓

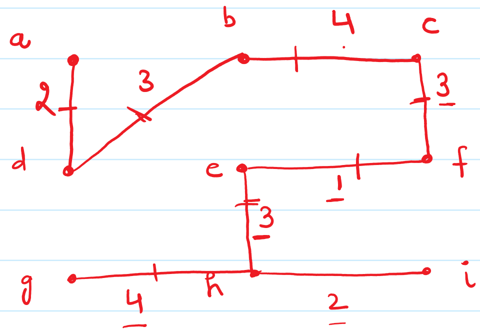
i → f = 4 X

e — b = 5 X

c — b = 4 ✓

[redacted]





weight of M.S.T = 22.

— ✗ —

e — b = 5 ✗, d = 7 ✗, h = 3 ✓

f — c = 3 ✓, b = 6 ✗, h = 4 ✗, i = 4 ✗

h — d = 8 ✗, g = 4 ✓, f = 4 ✗, i = 2 ✓

c — b = 4 ✓

g → d = 6 ✗

b — a = 5 ✗, d = 3 ✓, e = 5 ✗, f = 6 ✗

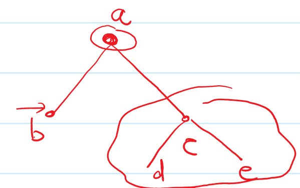
d — a = 2 ✓, e = 7 ✗, h = 8 ✗, g = 6 ✗

a → b = 5 ✗

Ordered Rooted Tree

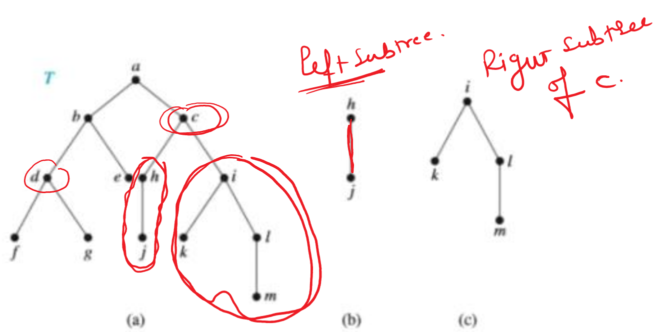
An **ordered rooted tree** is a rooted tree where the children of each internal vertex are ordered. Ordered rooted trees are drawn so that the children of each internal vertex are shown in order from left to right.

In an ordered binary tree, if an internal vertex has two children, the first child is called the **left child** and the second child is called the **right child**. The tree rooted at the left child of a vertex is called the **left subtree** and tree rooted at the right child of a vertex is called the **right subtree** of the vertex.



b is called left child of a.
and c is called Right child of a

What are the left and right children of d in the binary tree T shown in Figure (a) (where the order is that implied by the drawing)? What are the left and right subtrees of c?



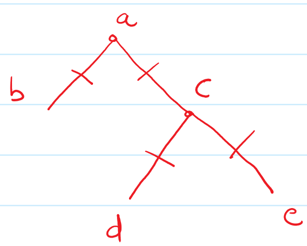
left child of d is → f
Right child of d is → g

FIGURE **A binary tree T and left and right subtrees of the vertex c.**

Theorem 1: A tree with n vertices has $n-1$ edges.

Theorem 2 :A full *m-ary* tree with i internal vertices contains $mi+1$ vertices.

$n = mi+1$ Total no. of vertices



How many edges does a tree with 10000 vertices have?

A. 99

B. 999

C. 9999

D. 99999

How many vertices does a full 5-ary tree with 100 internal vertices have?

A. 501
B. 5001
C. 4999
D. 499

$m = 5, \quad i = 100$

$n = mi + 1$

$= (5)(100) + 1$

$= 500 + 1 = 501$

$i = \frac{n-1}{m}, \quad l = \frac{(m-1)(n)+1}{m}$

Theorem 3 : A full *m-ary* tree with

(i) n vertices has $i = \frac{n-1}{m}$ internal vertices and $l = \frac{(m-1)n+1}{m}$ leaves,

(ii) i internal vertices has $n = mi + 1$ vertices and $l = (m-1)i + 1$ leaves,

(iii) l leaves has $n = \frac{ml-1}{m-1}$ vertices and $i = \frac{l-1}{m-1}$ internal vertices.

(i) no. of vertices
(ii) no of internal vertices
(iii) no. of leafs

How many leaves does a full *3-ary* tree with 100 vertices have?

A. 69

B. 67

C. 201

D. 33

$m = 3, \quad n = 100$

$$l = \frac{(m-1)\,n+1}{m} = \frac{(2)(100)+1}{3} = \frac{201}{3} = 67.$$

How many internal vertices does a full *4-ary* tree with 49 leaves have?

A. 48

B. 46

C. 18

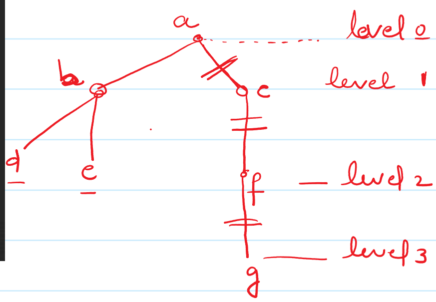
D. 16

$m = 4, \quad l = 49$

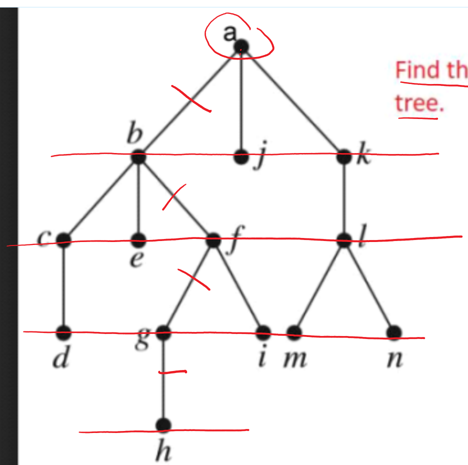
$$i = \frac{l-1}{m-1} = \frac{49-1}{4-1} = \frac{48}{3} = 16.$$

The **level** of a vertex v in a rooted tree is the length of the unique path from the root to this vertex.

The **height** of a rooted tree is the length of the longest path from the root to any vertex.



Height of this tree = 3.



Find the levels of all vertices and height of the tree.

Height of this tree = 4.

level 0 → a

level 1 → b, j, k.

level 2 → c, e, f, l.

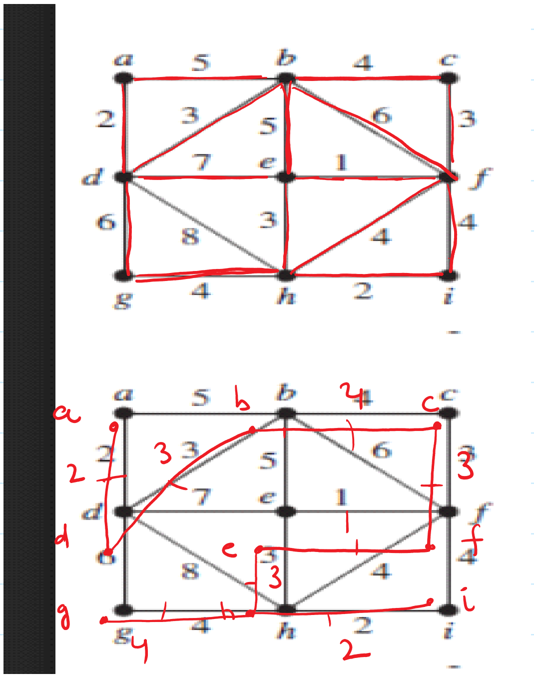
level 3 → d, g, i, m, n.

level 4 → h

The root a is at level 0. Vertices b, j, and k are at level 1. Vertices c, e, f, and l are at level 2. Vertices d, g, i, m, and n are at level 3. Finally, vertex h is at level 4. Because the largest level of any vertex is 4, this tree has height 4.

Arrange the edges of this graph in ascending order

Arrange the edges of this graph in ascending order



1 — ef ✓

2 — ad ✓, hi ✓

3 — db ✓, cf ✓, eh ✓

4 — bc ✗, gh ✓, fi ✗, fh ✗

5 — ab ✗, be ✗

6 → bf ✗, dg ✗

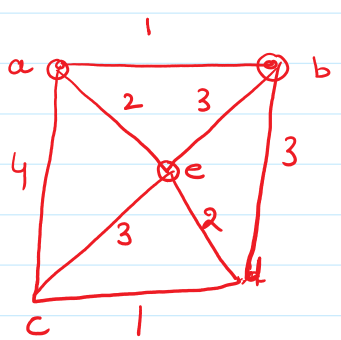
7 → de ✗

8 → dh ✗

weight of MST = 2+3+4+3+1+3+4+2

= 22

Prim's Algorithm for finding Minimum Spanning tree



(i) First of all select one edge from this graph whose weight is minimum.

ab = 1 ✓

a — e = 2 ✓, c = 4

b — e = 3 ✗, d = 3 ✗

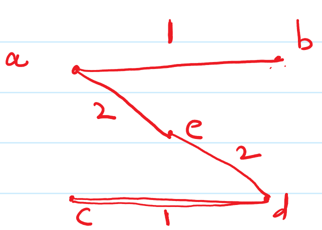
e — b = 3 ✗, d = 2 ✓, c = 3 ✗

d — b = 3 ✗, c = 1 ✓

c — a = 4 ✗, e = 3 ✗

weight of M.S.T = 1+2+2+1

= 6 Ans



$\underline{\underline{Q}}$ F