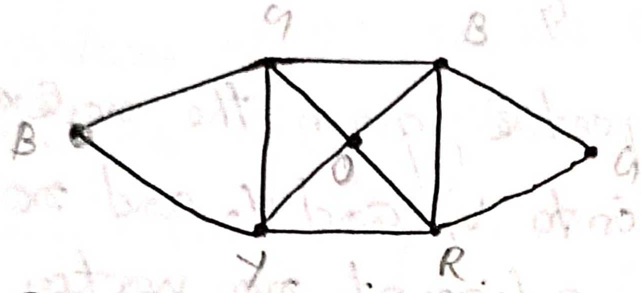
8-3-23

MODULE-5

GRAPH COLORING

→ Assignment of Colour ~~of~~ to each vertex of a graph so that no two adjacent vertices has the same colour.



⁘→ Chromatic Number . $\chi(G)$

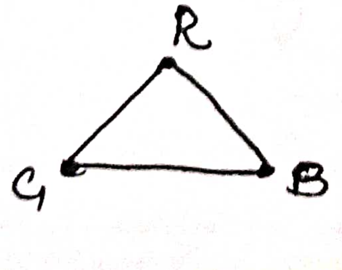
Least no: of colours needed to colour a graph is called Chromatic no: of the graph.

Find the Chromatic no: of graph.

(1) What is the Chromatic no: of K_n:

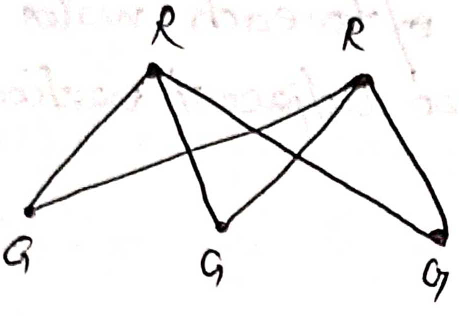
Since every vertex is adjacent to every other vertex. all vertices must be assigned different colours.

$n=3$.



→ What is the chromatic number of Complete Bipartite Graph $K_{m,n}$?

$K_{2,3}$

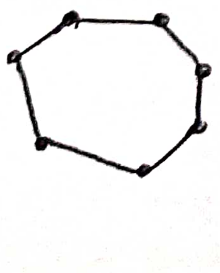


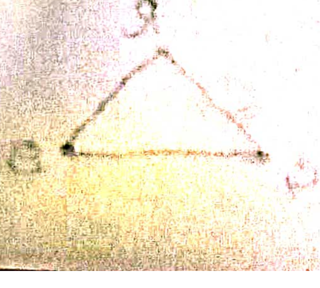
Since it is a bipartite graph the vertex set V is divided into V_1 and V_2. and no vertices in V_1 are adjacent, no vertices in V_2 are adjacent. So vertices in V_1 is coloured with one colour and vertices in V_2 are coloured with second colour. ∴ Chromatic no: is 2.

③ Chromatic no: of Cycle C_n?

$n = 6$.

When n is even $n \geq 4$
Chromatic number is 2.
When n is odd $n \geq 3$
Chromatic no: is 3.



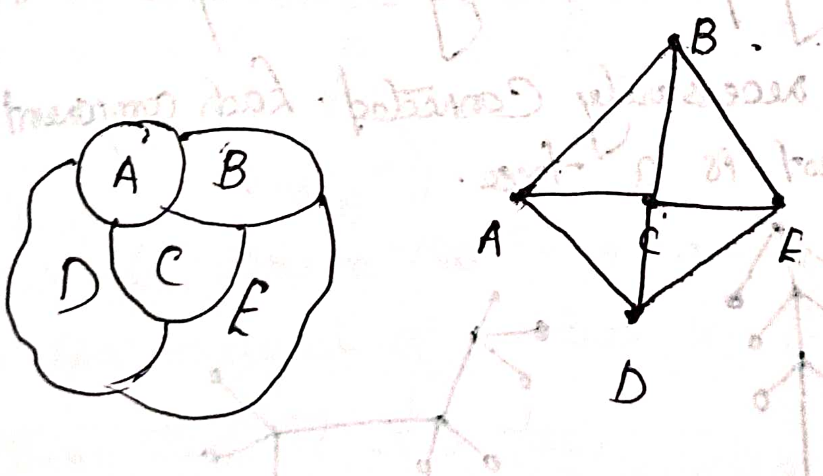


→ Application of Graph Colouring (Assignment 2).

*→ Dual Graph

A map in a plane can be represented by a graph where each region of the map is represented by a vertex. Edges connect two vertices of the regions represented by these vertices have common border. Two regions that touch only at one point is not considered adjacent. The resulting graph is called dual graph of the map.

Ex:



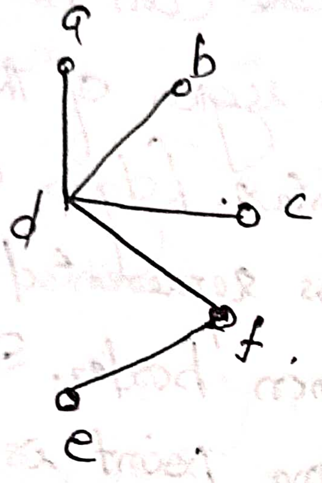
*→ The Four Colour Theorem

Chromatic no: of a planar graph is not greater than four.

→ **Trees**

Defn A tree is a connected graph with no simple circuit.

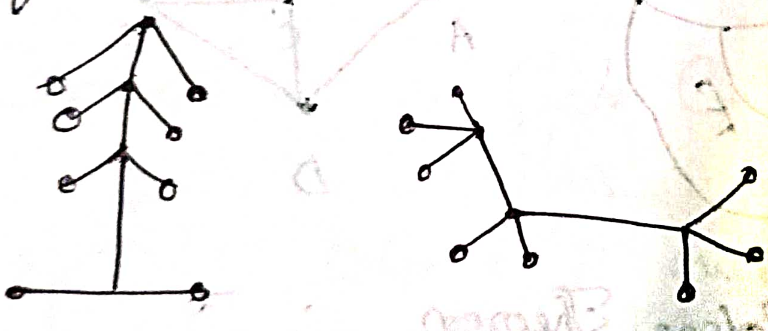
Ex :



→ **Forest**

A graph containing no simple circuit but not necessarily connected. Each component of a forest is a tree.

Eg:



*→ **Properties of a Tree**

- **Theorem**

An undirected graph is a tree if and only if there exists a unique simple path b/w any two of its vertices.

Proof

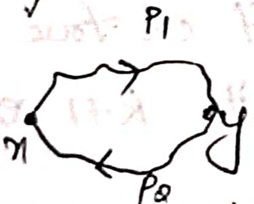
Assume that T is a tree by definition that implies T is connected.

Therefore by definition of connectedness there exist a simple path b/w every pair of vertices.

To show that this path is unique,

if possible the path is not unique.

Let x,y be two vertices and ∃ two paths b/w x and y but that will form a circuit in T. Which is not possible. Therefore the path is unique.



Conversely assume that ∃ a unique simple path b/w every pair of vertices in T.

→ To show that T is a tree,

① Clearly by assumption T is connected.

② Now to prove T has no circuit.

If possible ∃ a circuit from x to x. Let y be any vertex in this circuit. This will give two paths P_1 and P_2 between x and y.

which is a contradiction to our assumption. So T has no circuit. That implies T is a tree.

→ A tree with n vertices has $n-1$ edges.

<u>Proof</u>

We use induction on n.

① Result is true for $n = 1, 2$.

$n=1$ $n=2$.

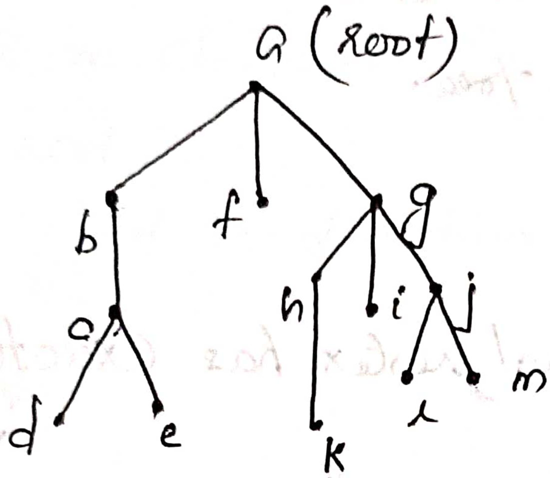
- Let us assume the result is true for $n=k$. i.e. a tree with k vertices has $k-1$ edges.
- Now to prove result is true for $n = k+1$. $(k+1)-1)$ i.e. to prove tree with $k+1$ vertices has k edges.

Let T be a tree with $k+1$ vertices. Let v be leaf of T and (v,w) be an edge in T. Remove vertex v and edge (v,w) from T which produces a new tree T'. T' has k vertices. So by our assumption it has $k-1$ edges. That implies T has $k-1+(1) = k$ edges.

→ Rooted trees

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

Ex :



→ Parent

A vertex u is a parent of v if there is a directed edge u → v. Then v is called child of u.

→ Siblings

vertices with same parents.

→ Ancestors of the vertex

its the vertices in the path from root to this vertex including the vertex and path.

→ A vertex with no children - leaf.

→ Internal vertices

• Vertices that have children.

→ <u>m-array tree</u>

If every internal vertex of a rooted tree has no more than m children. Then tree is called m-array tree.

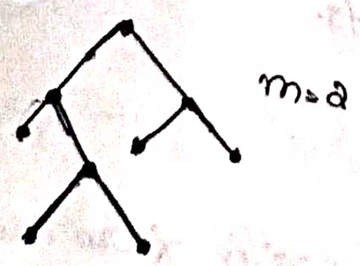
→ <u>Full m-array tree</u>

If every internal vertex has exactly m children.

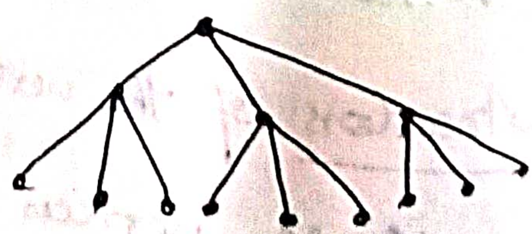
→ <u>Binary tree</u>

An m-array tree with m=2.

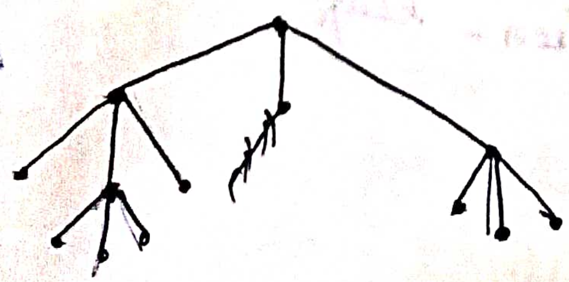
<u>Binary tree</u>



<u>Full 3-array tree</u>



<u>3-array tree.</u>



→ **Theorem** n = no: of vertices in the tree.

✓ A full m-array tree with i internal vertices contains $n = m \cdot i + 1$ vertices.

→ A full m-array tree has i internal vertices each with m children. So total mi children + one root.

ie, Tot. no: of vertices $= mi + 1$

✲ **Theorem**

✓ i) A full m-array tree with 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.

Proof

n = no: of vertices

l = no: of leaves

i = no: of internal vertices.

$$n = l + i \rightarrow (1)$$

By previous theorem,

$$n = mi + 1 \longrightarrow (2)$$

(i) $(2) \Rightarrow \frac{n-1}{m} = i$

Put i in eq (1),

$$n = l + i$$

$$n = l + \frac{n-1}{m}$$

$$n = \frac{lm + n - 1}{m}$$

$$nm = lm + n - 1$$

$$\frac{nm}{n-1} = \cancel{lm} \qquad lm = nm - (n-1)$$

$$l = \left(\frac{nm}{n-1}\right)m \qquad l = \boxed{\frac{(m-1)n+1}{m}}$$

(ii) $n = mi + 1 \to$ Clearly $n = mi + 1$.

$\to l = (m-1)i + 1$

Eq. Now equate with eqn (1);

$$n = l + i$$

$$mi + 1 = l + i$$

$$mi - i + 1 = l$$

$$i(m-1) + 1 = l$$

iii) Eq ① $\Rightarrow$ $l = n - i$. $(n = l + i)$ $n = \dfrac{ml-1}{m-1}$ $i = \dfrac{l-1}{m-1}$

Substitute i from eq ②.

$$n = mi + 1$$

$$\frac{n-1}{m} = i \rightarrow ③$$

Sub ③ in ① $\Rightarrow$ $l = n - \dfrac{(n-1)}{m}$

$$l = \frac{nm - (n-1)}{m}$$

$$lm = nm - (n-1).$$

$$lm = nm - n + 1$$

$$lm = n(m-1) + 1$$

$$lm - 1 = n(m-1)$$

$$\frac{lm-1}{m-1} = n \rightarrow ④.$$

Eq ① $\Rightarrow$ $i = \cancel{l-n}\ n - l.$

Eq ④ in ① $\Rightarrow$ $i = \dfrac{lm-1}{m-1} - l$

$$i = \frac{lm - 1 - l(m-1)}{m-1}$$

$$i(m-1) = \cancel{lm} - 1 - \cancel{lm} + l \quad \frac{l-1}{m-1}$$

$$i = \frac{l-1}{m-1}$$

→ <u>level of a vertex</u>

level of a vertex in a rooted tree is the length of the unique path from root to this vertex. Level of root is defined to be zero.

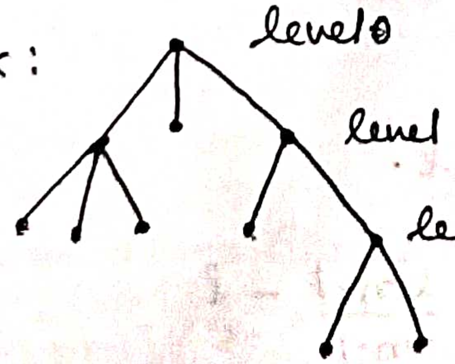
→ <u>Height of rooted tree</u>

It is the maximum level of a vertex in a rooted tree.

→ <u>Balanced m-array tree.</u>

Rooted m-array tree of height h is balanced if all leaves are atleast at levels h or h-1.

Ex:



m 3 array tree
height of this tree = 3
not balanced.

→ Theorem

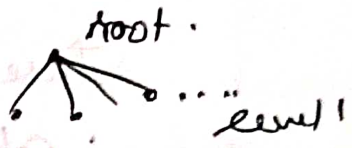
There are atmost m^h leaves in an m-array tree of height h.

Proof

Induction on h.

We use induction on h.

- for $h=1$



Consider m-array tree of height $h=1$. It has a root and leaves not more than m because it is an m-array tree. Therefore total no: of leaves $= m$. ie m^1 ~~(m)~~. So result is true for $h=1$.

- ~~For $h = h+1$~~

Assume that the result is true for an m-array tree of ~~[illegible]~~ height less than h.

To prove that the result is true for an m-array tree of height h. ie is to prove it has atmost m^h ~~trees~~ leaves.

Let T be an m-array tree of height h deleting all the edges from root to all the vertices at level 1. We can get subtrees of height $\leq h-1$. But the result is true for these subtrees

∴ Each subtree has atmost m^{h-1} leaves.

There are atmost m such subtrees.

∴ Total atmost $m(m^{h-1})$ leaves.

subtree m^{h-1} atmost
total $= m \cdot m^{h-1}$
$= m^h$

$$= m^{h-1+1} = \underline{\underline{m^h}}$$

i.e atmost m^h leaves.

These leaves are the same for original tree

∴ atmost m^h leaves for a tree of height h.

→ **Corollary**

If an m-array tree of height h has l leaves then $h \geq \lceil \log_m l \rceil$.

If the m-array tree is full and balanced then $h = \lceil \log_m l \rceil$

Proof

- By using our theorem,

$$l \leq m^h.$$

$$\Rightarrow \log_m l \leq h.$$

$$\left[\begin{array}{l} n = b^a \\ \log_b n = a \end{array} \right]$$

or $h \geq \log_m l$

$$h \geq \lceil \log_m l \rceil$$

→ Suppose tree is balanced and full, then each leaf is at level h or $h-1$. because height is h ∃ atleast one leafe at level h.

$\Rightarrow l > m^{h-1}$

Also by previous theorem,

$$l \leq m^h.$$

So combining both inequalities,

$$m^{h-1} < l \leq m^h$$

Applying log on both sides,

$$h-1 < \log_m l \leq h.$$

$$\Rightarrow h = \lceil \log_m l \rceil$$

Spanning tree

→