Unit-5

Graph Theory

Graph:

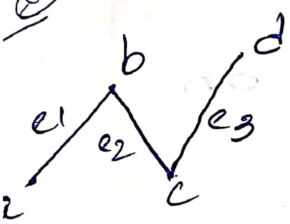
A graph G is defined as an ordered pair, $G = (V, E)$

where V is a set of vertices (or) nodes (or) points

E is called as set of edges (or) line.

Ex:-

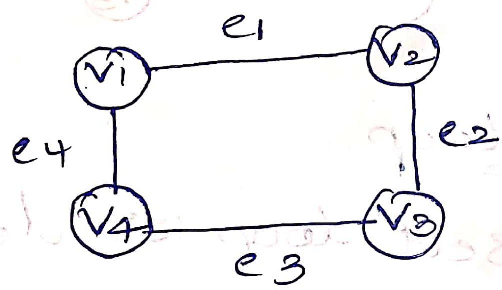
Ex



$e_1 = (a, b)$

$e_2 = (b, c)$

$e_3 = (c, d)$



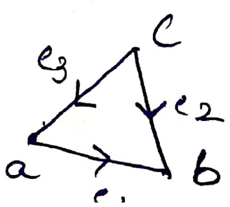
$V = \{v_1, v_2, v_3, v_4\}$

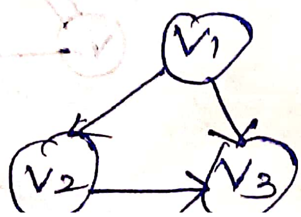
$E = \{e_1, e_2, e_3, e_4\}$

$= \{(v_1, v_2), (v_2, v_3), (v_3, v_4), (v_1, v_4)\}$

Directed Graph:

If every edge in the graph has directions.



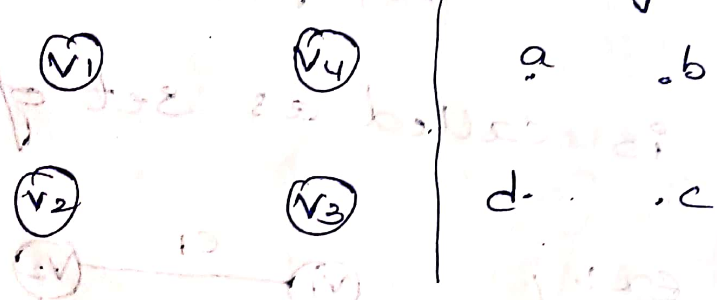


Undirected Graph:-

If every edge in the graph have no direction [no arrows]

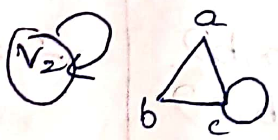
Null Graph (or) [Isolated Graph]

It is a graph with a set of vertices, but no edges



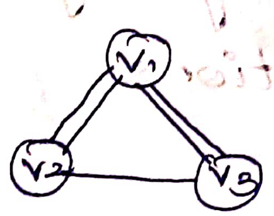
Self-loop

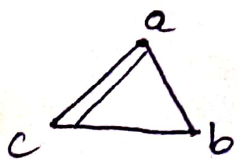
A self-loop (or) loop is an edge that connects a vertex to itself.



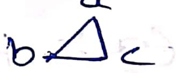
Parallel (or) multiple edges

If there is more than one edge between a pair of vertices

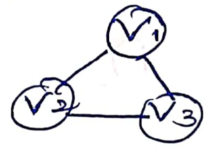




Simple Graph :-



A graph which have no parallel edges with no self loops.



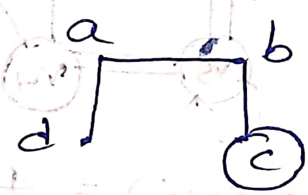
Multi-Graph:

It is a graph with parallel edges but no self loops.

[need not all to be ll'el 1 is enough].

Pseudograph:

It is a graph that allows parallel edges and self-loops



Degree of a vertex :-

It is the number of edges connected to the vertex.

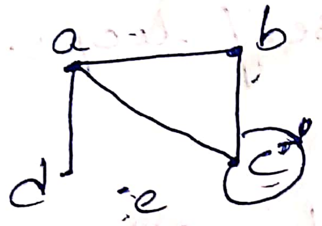
(i) undirected Graph

deg(v) = number of edges.

* → If a vertex has a self loop, it adds 2 to the degree.

Ex:- Find the degree of given vertices.

1)

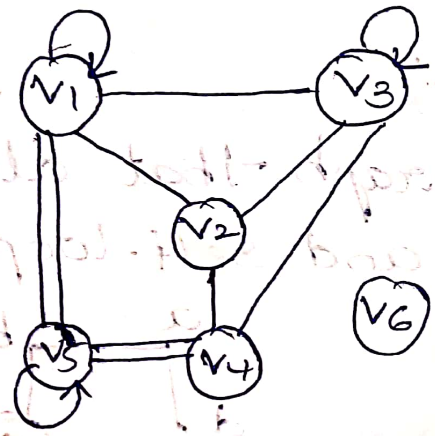


deg (a) = 3 , deg (d) = 1

deg (b) = 2 , deg (c) = 2+2 = 4.

deg (e) = 0

2)



deg (V1) = 4 + 2 = 6

deg (V2) = 2

deg (V3) = 3 + 2 = 5

deg (V4) = 4

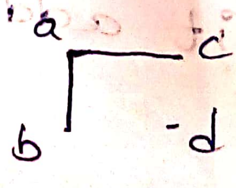
deg (V5) = 4 + 2 = 6

deg (V6) = 0

Isolated Graph

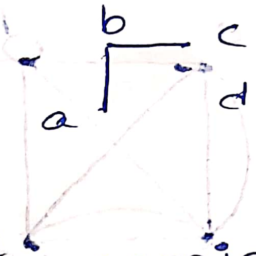
If deg (V) = 0 then V is called Isolated Graph.

deg (d) = 0 →



Pendant Graph

If $\deg(v) = 1$ then 'v' is called pendant graph

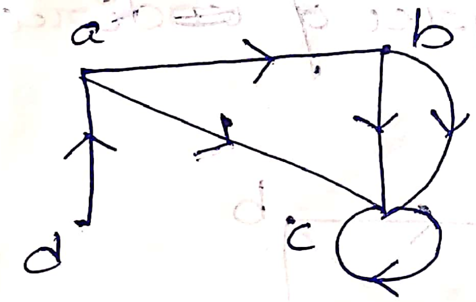


(ii) Directed Graph :-

In-degree :- No. of edges coming ($\deg v^+$) into the vertex.

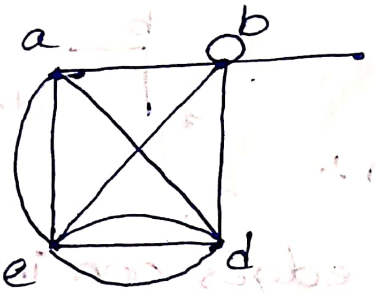
out-degree ($\deg v^-$) No. of edges going out of the vertex.

Ex:- A loop is added only once. for both indegree & outdegree



Indegree (v^+)	Outdegree. (v^-)
$\deg(a) = 1$	$\deg(a) = 2$
$\deg(b) = 1$	$\deg(b) = 2$
$\deg(c) = 3+1 = 4$	$\deg(c) = 1$ [loop]
$\deg(d) = 0$	$\deg(d) = 1$

Ex of undirected Graph :



deg (a) = 4

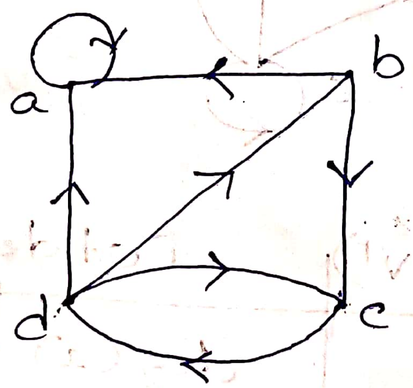
deg (b) = 4+2=6

deg (c) = 1

deg (d) = 5

deg (e) = 6

Find the degree of ~~un~~directed graph.



Indegree (V^+)	Outdegree (V^-)
deg(a) = 2+1 = 3	deg (a) = 1 [loop]
deg (b) = 1	deg (b) = 2
deg (c) = 2	deg (c) = 1
deg (d) = 1	deg (d) = 3

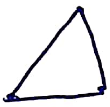
Scanned with OKEN Scanner

cycle Graph:-

A graph that consists of a single cycle is called cycle graph.

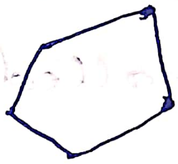
→ It is denoted by C_n

[$n \to$ no. of vertices], $n \geq 3$.



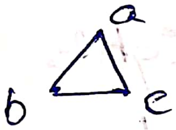
C_3

C_4



C_5

$\boxed{\text{vertices = edges}}$



→ a → b →

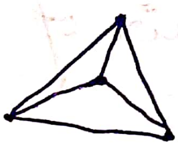
$a \to c \to b \to a$

start & end vertex must be same

wheel Graph

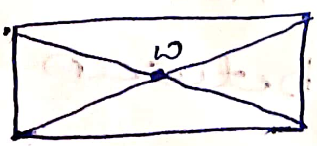
A wheel graph of order 'n' is a graph obtained by joining a single new vertex to each vertex of cycle graph. C_{n-1}

It is denoted by W_n



W_4

$C_{4-1} = C_3$



W_5

$[W_n = C_{n-1}]$

$= C_4$

odd vertex

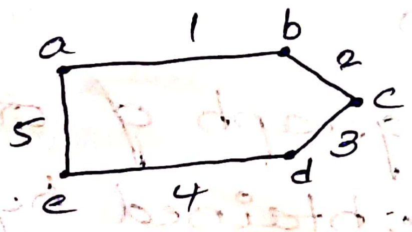
A vertex with odd degree is called odd vertex.

Even vertex

A vertex with even degree is called even vertex.

weighted Graph

A graph in which weights are assigned to every edge is called a weighted Graph.



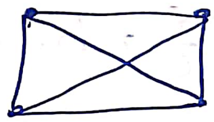
Here 1, 2, 3, 4, 5 are weights assigned to each edge.

complete Graph

It is a simple graph in which there is exactly one edge between every pair of vertices.

Denoted by K_n

K_3



K_4

No. of edges in $K_n = \frac{n(n-1)}{2}$

$n \to$ no. of vertices

$\Rightarrow K_3 = \frac{3\cdot(3-1)}{2} = \frac{3\times 2}{2}$

$\Rightarrow$ 3 edges

$n=4$

let $K_4 = \frac{4\times(4-1)}{2} \Rightarrow \frac{4\times 3}{2} \Rightarrow$ 6 edges

---

K_{50}

for 50 vertices $\to$ 1225 edges $\frac{50(50-1)}{2}$

$\cdot v_1$ 0 edge.

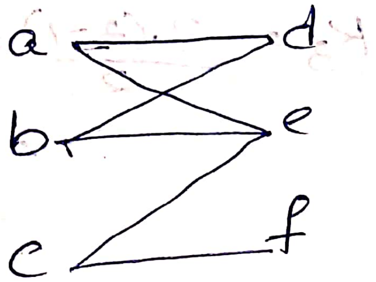
$n=1 \therefore K_n = \frac{1(1-1)}{2} \Rightarrow \frac{1\times 0}{2} = 0$ edge

---

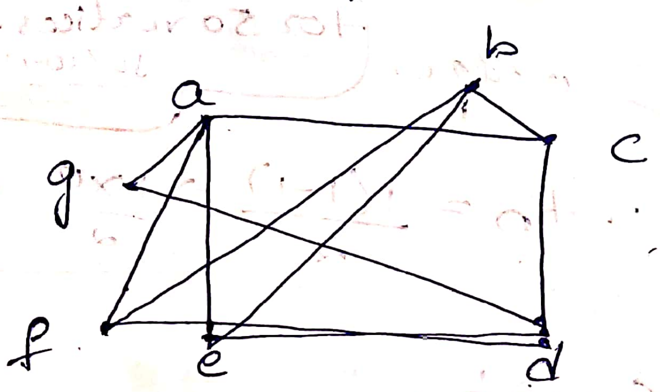
Bipartite Graph:

A simple graph is bipartite if its vertex set V can be partitioned into 2 disjoint set V_1 & V_2 such that each edge of graph connects a vertex of V_1 to a vertex in V_2.

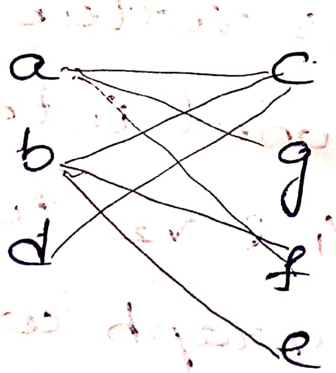
$V = \{a, b, c, d, e, f\}$



Is the graph bipartite?

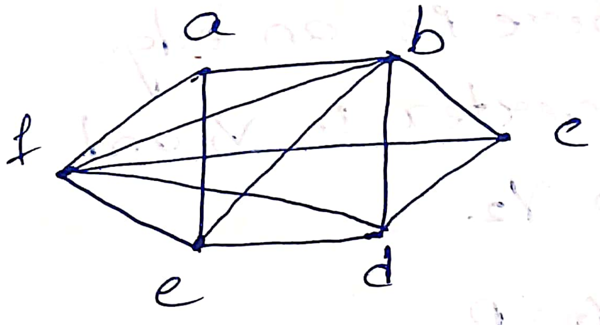


$V = \{a, b, c, d, e, f, g\}$

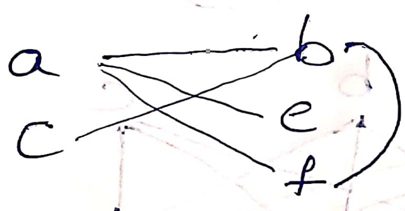


∴ It is bipartite

Is the graph bipartite.

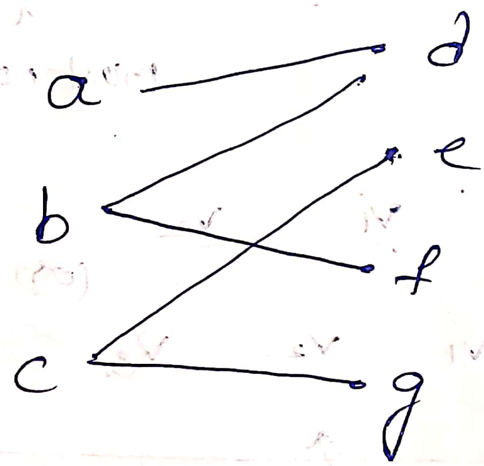


$V = \{a, b, c, d, e, f\}$



b e f are on same side

so it is not bipartite.



bipartite

complete bipartite Graph :

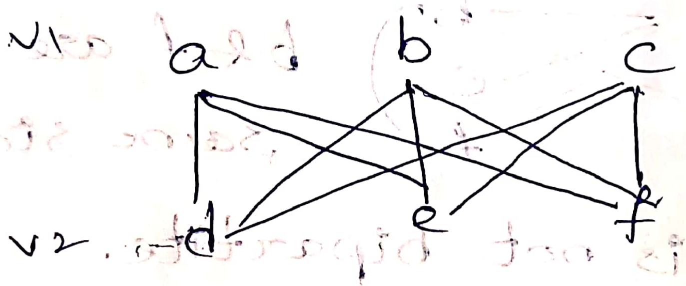
combination of complete &
bipartite graph.

No vertex must be
alone, all must be connected

A Bipartite Graph is complete if there is an edge between every vertex in V_1 and every vertex in V_2.

If $V_1 = m$, $V_2 = n$

[Denoted by $K_{m,n}$.]

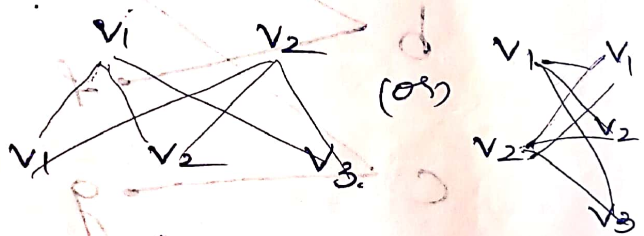


all must be connected to all :

must not be alone.

$K_{3,3}$

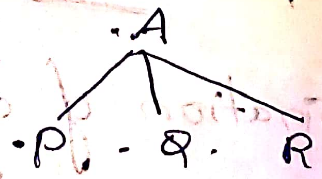
$K_{2,3}$ ⇒



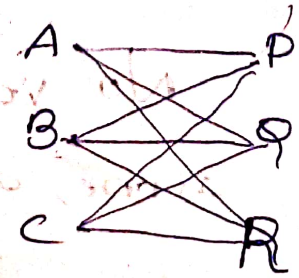
1) Draw $K_{2,3}$

P
Q
R

2) Draw $K_{1,3}$



3) Draw $K_{3,3}$



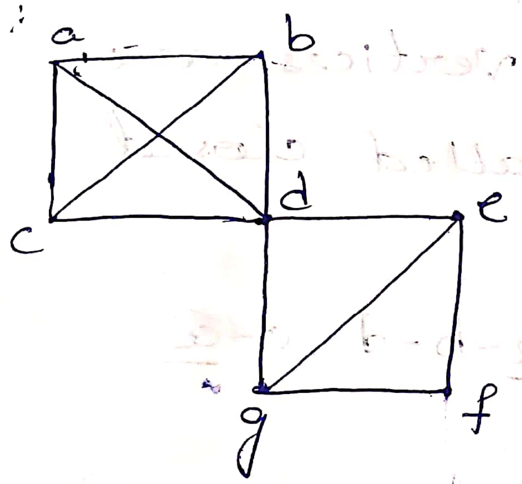
Path/walk :-

It is a sequence of vertices and edges of a graph.

→ vertex and edges repitition is allowed.

→ length of path = No. of edges in the path.

Ex :



Q) Sequence	path or not	length → edges
1) a-d-e-f-g	yes	4
2) a-e-f-g	No.	
3) a-d-c-a-b-c	yes	5
4) a-d-c-b-d-c	yes	5

Trivial path [· a]

A path of length 0 is called trivial path.

open path

A path in which initial and terminal vertices are distinct is called open path.

Ex:- a-d-e-f-g is open path.

closed path / circuit

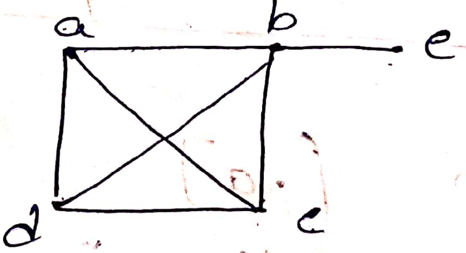
A path in which initial and terminal vertices are ~~not~~ same is called closed path.

Ex:- a-b-a, a-b-d-c-a

simple path

A path is said to be simple if all the edges and vertices on path are different except possibly of end points.

Ex:-



i) a-b-c-a
ii) a-b-c-d-a

are simple paths.

3) a-b-c-d-c-a
4) a-b-c-a-b } not simple paths.

Representation of Graphs

3 types

1) Adjacency matrix
2) Incidence matrix.
3) path matrix of a graph.

1) Adjacency matrix

undirected graph   directed graph.

Undirected Graph

Procedure

→ 1) write the vertices is row & colomn wise.

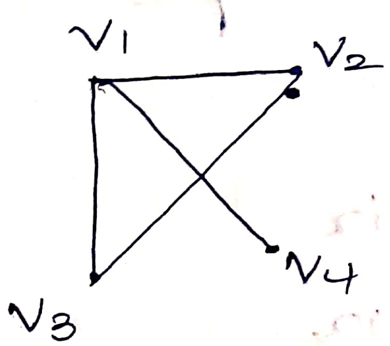
2) If there are edges → 1
no edges → 0.

$$A = (a_{ij})_{m \times n} = \begin{cases} 1 & \text{if } (a_i, a_j) \in E \\ 0 & \text{if } (a_i, a_j) \notin E. \end{cases}$$

Find the adjacency matrix of the undirected graph.

symmetric.

1)

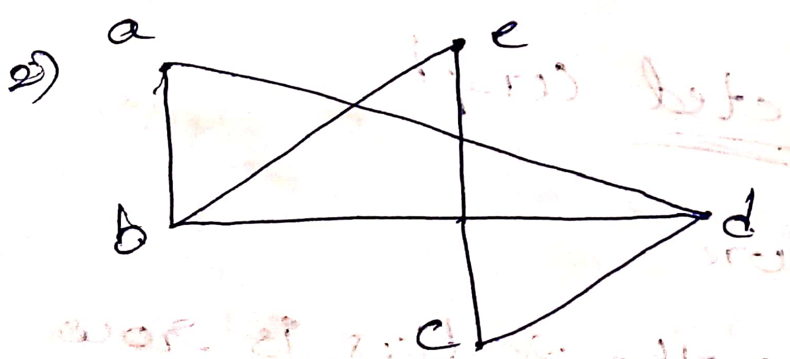


[row wise]

$v_1 \left< \begin{matrix} v_3 \\ v_2 \\ v_4 \end{matrix} \right\} \to 1$

$$
\begin{array}{c|cccc}
 & v_1 & v_2 & v_3 & v_4 \\
\hline
v_1 & 0 & 1 & 1 & 1 \\
v_2 & 1 & 0 & 1 & 0 \\
v_3 & 1 & 1 & 0 & 0 \\
v_4 & 1 & 0 & 0 & 0
\end{array}
$$

2)

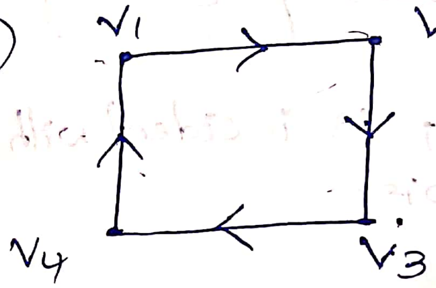


$$
\begin{array}{c|ccccc}
 & a & b & c & d & e \\
\hline
a & 0 & 1 & 0 & 1 & 0 \\
b & 1 & 0 & 0 & 1 & 1 \\
c & 0 & 0 & 0 & 1 & 1 \\
d & 1 & 1 & 1 & 0 & 0 \\
e & 0 & 1 & 1 & 0 & 0
\end{array}
$$

Scanned with OKEN Scanner

Adjancency matrix in Directed Graph

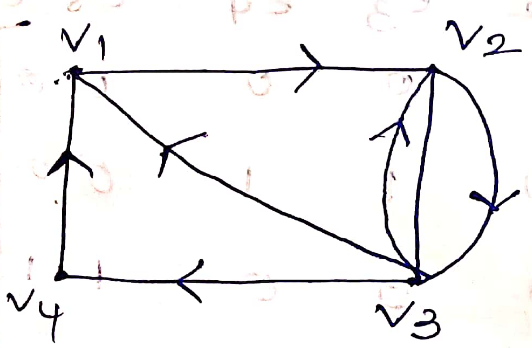
If there is a direction → 1
no directions → 0.
↓
not symmetric.

1) 

[Row wise]

$V_1 \to V_2$ ✓
V_2 to V_2 ✗

$$
\begin{array}{c|cccc}
 & V_1 & V_2 & V_3 & V_4 \\
\hline
V_1 & 0 & 1 & 0 & 0 \\
V_2 & 0 & 0 & 1 & 0 \\
V_3 & 0 & 0 & 0 & 1 \\
V_4 & 1 & 0 & 0 & 0.
\end{array}
$$

2) 

$$
\begin{array}{c|cccc}
 & V_1 & V_2 & V_3 & V_4 \\
\hline
V_1 & 0 & 1 & 0 & 0 \\
V_2 & 0 & 0 & 1 & 0 \\
V_3 & 01 & 01 & 0 & 1 \\
 & 1 & 0 & 0 & 0
\end{array}
$$

2) Incidence matrix

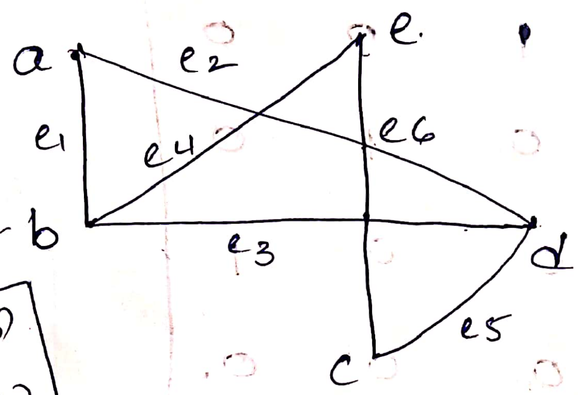
Undirected — Directed

1) Undirected.

{ e_1 is near a, b so put 1, remaining 0 }

$$m_{ij} = \begin{cases} 1, & \text{when } v_i \text{ is incident with } e_j \\ 0, & \text{otherwise.} \end{cases}$$

Ex 1)



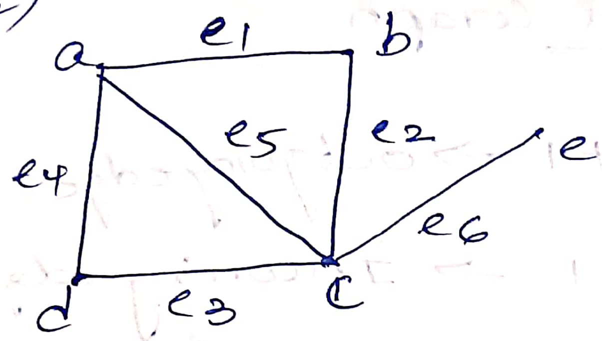
{ column wise }

$e_1 = (a, b)$
1 1

$e_2 = (a, d)$
1 1

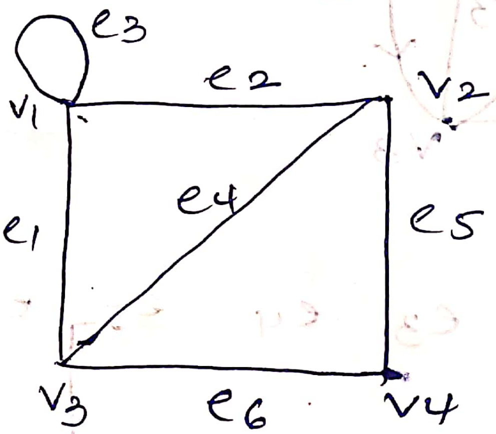
$$\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array}
\begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\
\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

Ex 2)



$$
\begin{array}{c|cccccc}
 & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\
\hline
a & 1 & 0 & 0 & 1 & 1 & 0 \\
b & 1 & 1 & 0 & 0 & 0 & 0 \\
c & 0 & 1 & 1 & 0 & 1 & 1 \\
d & 0 & 0 & 1 & 1 & 0 & 0 \\
e & 0 & 0 & 0 & 0 & 0 & 1
\end{array}
$$

3)



[loop → +2]

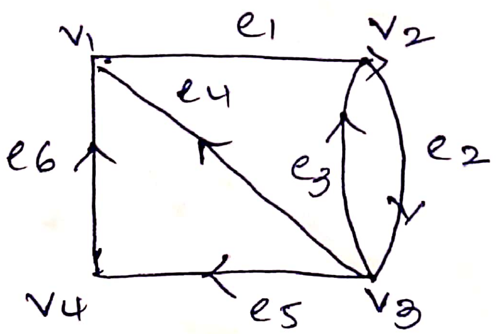
$$
\begin{array}{c|cccccc}
 & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\
\hline
v_1 & 1 & 1 & 2 & 0 & 0 & 0 \\
v_2 & 0 & 1 & 0 & 1 & 1 & 0 \\
v_3 & 1 & 0 & 0 & 1 & 0 & 1 \\
v_4 & 0 & 0 & 0 & 0 & 1 & 1
\end{array}
$$

Incidence matrix of Directed Graph

$$b_{ij} = \begin{cases} 1, & \text{if } v_i \text{ is initial vertex of edge } e_j \\ -1, & \text{if } v_i \text{ is final vertex of edge } e_j \\ 0, & \text{otherwise} \end{cases}$$

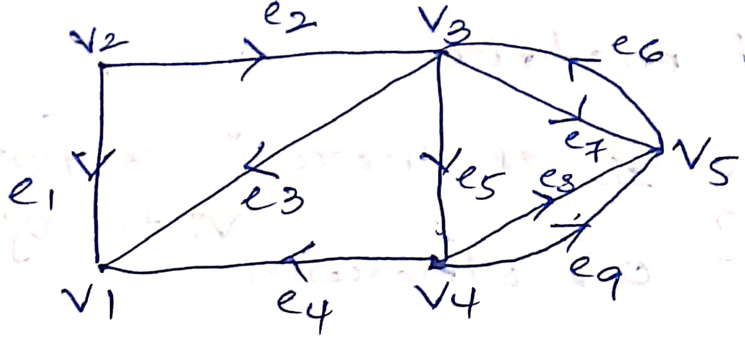
$$\left\{ \begin{array}{l} \text{Incoming} \rightarrow -1 \\ \text{Outgoing} \rightarrow +1 \\ \text{otherwise} \rightarrow 0 \end{array} \right\} \boxed{\text{row wise}}$$

Ex 1)



$$\begin{array}{cc} & \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{array}{cc} 1 & v_1 \\ 2 & v_2 \\ 3 & v_3 \\ 4 & v_4 \end{array} & \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \end{array}$$

2)



$$
\begin{array}{c|c}
 & \begin{array}{ccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 \end{array} \\
\begin{array}{c} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \end{array} &
\left[\begin{array}{ccccccccc}
-1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 1 & 0 & 1 & -1 & 1 & 0 & 0 \\
0 & 0 & 0 & +1 & -1 & 0 & 0 & +1 & +1 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1
\end{array}\right]
\end{array}
$$

3) path matrix

$$P_{ij} = \begin{cases} 1, & \text{if there is a path from } v_i \text{ to } v_j \\ 0, & \text{otherwise} \end{cases}$$

$A \rightarrow$ Adjacency matrix

$P = [P_{ij}]$ = path matrix

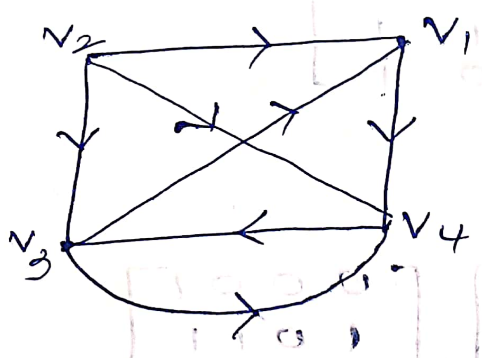
Then $P_{ij} = 1$ iff there is non-zero number in the entry of the matrix B_n

where $\boxed{B_n = A + A^2 + A^3 + \dots + A^n}$

$A \rightarrow$ Adjacency

$n \rightarrow$ no of vertices.

Find the path matrix of



Sol:- need to find

$$B_n = A + A^2 + \dots + A^n$$

$\because$ no. of vertices $= 4$

$$\therefore B_n = A + A^2 + A^3 + A^4$$

$A \rightarrow$ Adjacency matrix.

$$A = \begin{array}{c} \\ V_1 \\ V_2 \\ V_3 \\ V_4 \end{array} \begin{array}{c} \begin{array}{cccc} V_1 & V_2 & V_3 & V_4 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$A^2 = A \cdot A$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0+0+0+0 & 0+0+0+0 & 0+0+0+1 & 0+0+0+0 \\ 0+0+1+0 & 0+0+0+0 & 0+0+0+1 & 1+0+1+0 \\ 0+0+0+0 & 0+0+0+0 & 0+0+0+1 & 0+0+1+0 \\ 0+0+1+0 & 0+0+0+0 & 0+0+0+0 & 0+0+1+0 \end{bmatrix}$$

$$A^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

$$A^3 = A^2 \cdot A$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A^3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 2 & 2 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$A^4 = A^3 \cdot A$$

$$= \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 2 & 2 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A^4 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 2 & 0 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\because B_n = A + A^2 + A^3 + A^4$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 2 & 2 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 1 & 1 \\ 2 & 0 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ \ldots \end{bmatrix}$$

$\therefore B_n =$

$\because P_{ij} =$

num

other

$\therefore B_n =$ v_1 v_2

$= P \Rightarrow$ v_3 v_4

2) Find

Sol:

$A =$ v_1 v_2 v_3 v_4

$A^2 =$

$$\therefore B_n = \begin{bmatrix} 1 & 0 & 2 & 3 \\ 5 & 0 & 6 & 8 \\ 3 & 0 & 3 & 5 \\ 2 & 0 & 3 & 3 \end{bmatrix}$$

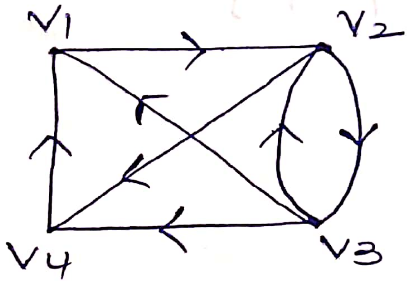
$\because$ $P_{ij} = 1$ if there is a non-zero number in B_n

otherwise 0.

$$\therefore B_n = P \Rightarrow \begin{array}{c|c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \end{array}$$

---

2) Find the path matrix of:



Sol:

$$B_n = A + A^2 + A^3 + A^4$$

$$A = \begin{array}{c|c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$A^2 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad A^3 = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 1 \\ 2 & 2 & 1 & 2 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$A^4 = \begin{bmatrix} 1 & 2 & 1 & 1 \\ 2 & 2 & 2 & 3 \\ 3 & 3 & 2 & 3 \\ 2 & 1 & 0 & 1 \end{bmatrix}$$

$\therefore B = A + A^2 + A^3 + A^4$

$$= \begin{bmatrix} 3 & 4 & 2 & 3 \\ 5 & 5 & 4 & 6 \\ 7 & 7 & 4 & 7 \\ 3 & 2 & 1 & 2 \end{bmatrix}$$

$\therefore$ $p_{ij} = 1 \rightarrow$ non-zero in B

otherwise 0.

$$\therefore P = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$\therefore$ There is a path

Isomorphism :-

2 Graphs are said to be isomorphic, if they satisfy the following conditions

(1) Both graphs must contain same no. of vertices

(2) " " same no. of edges

(3) " " same degree sequence

(4) one-to-one correspondence between the vertices of 2 graphs must be same.

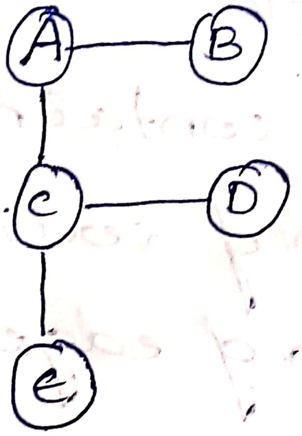
(5) Edge preserving should be satisfied i.e, each edge in graph 1 is equivalent to an edge in graph 2.

(6) Adjacency matrix of both graphs must be same.

Determine whether the following graphs are isomorphic.

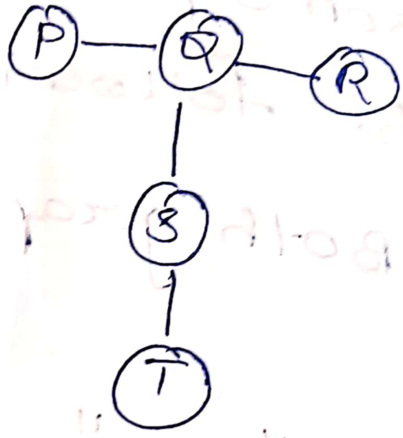
G_1

1)



G_2

2)



must satisfy 6 conditions

$\underline{G_1}$

1) No. of vertices = 5

2) No. of edges = 4

3) Degree of G_1

$= \{A, B, C, D, E\}$

$= \{2, 1, 3, 1, 1\}$

$\underline{G_2}$

1) No. of vertices = 5

2) No. of edges = 4

3) Degree of G_2

$\Rightarrow \{P, Q, R, S, T\}$

$\Rightarrow \{1, 3, 1, 2, 1\}$

If Ascending order both degrees are same

$\{1, 1, 1, 2, 3\}$

$\{1, 1, 1, 2, 3\}$

4) one-to-one correspondence

write vertices for all in G_1 & G_2

(2) A — B (1), C (3)

(1) B — A (2)

(3) C — A (2), D (1), E (1)

(1) D — C (3)

(1) E — C (3)

(1) P — Q (3)

(3) Q — P (1), R (1), S (2)

(1) R — Q (3)

(2) S — Q (3), T (1)

(1) T — S (2)

C <=> Q

A <=> S

B <=> T

D <=> P (or) R also

E <=> R.

5) edge preserving

A - B <=> S - T

A - C <=> S - Q

C - D <=> Q - P

C - E <=> Q - R

Edges of G_1 must be equal to edges of G_2

Scanned with OKEN Scanner

(6) Adjacency matrix

G_1

$$\begin{array}{c|c} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

G_2

$$\begin{array}{c|c} & \begin{matrix} S & T & Q & P & R \end{matrix} \\ \begin{matrix} S \\ T \\ Q \\ P \\ R \end{matrix} & \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

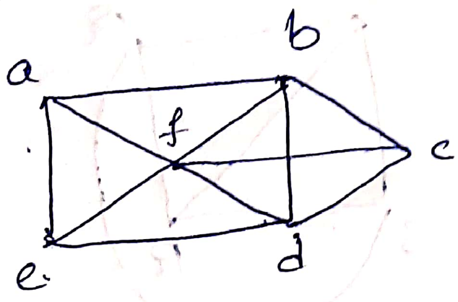
$P \rightarrow D$
$Q \rightarrow C$
$R \rightarrow E$
$S \rightarrow A$
$T \rightarrow B$

$\because G_1 = G_2$

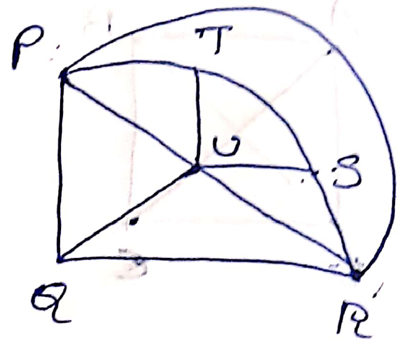
$\therefore$ The Given Graphs are isomorphic.

(2Q)

G1



G2



G1	G2
Sol: 1) vertices: 6	vertices: 6
2) Edges: 11	Edges: 11
3) degree	degree
{a, b, c, d, e, f}	{P, Q, R, S, T, U}
{3, 4, 3, 4, 3, 5}	{4, 3, 4, 3, 3, 5}

G1

4) 1-1

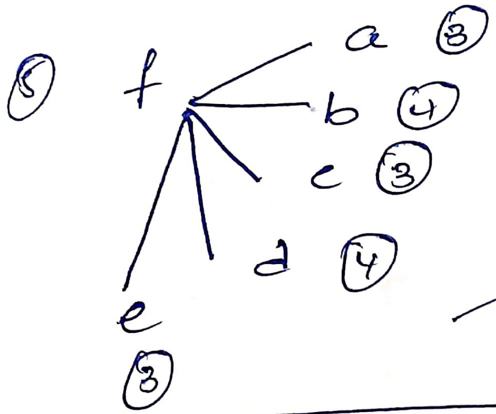
(3) a — b (4), f (5), e (3)

(4) b — a (3), c (3), d (4), f (5)

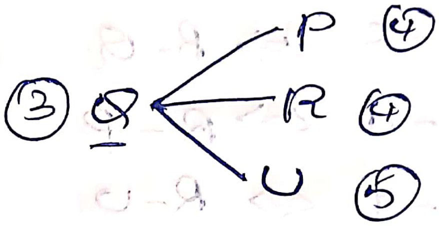
(3) c — b (4), f (5), d (4)

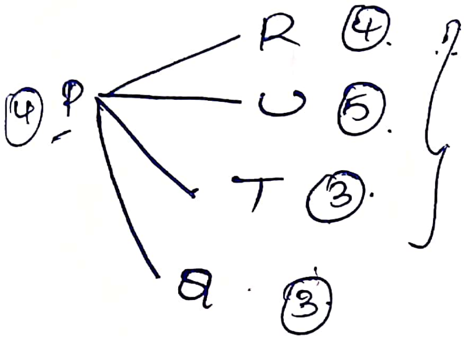
(4) d — c (3), f (5), b (4), e (3)

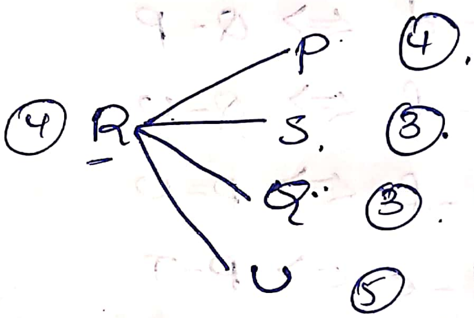
(3) e — f (5), a (3), d (4)

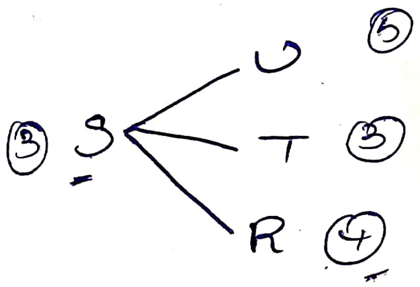


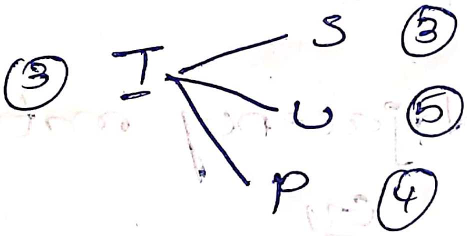
Ge

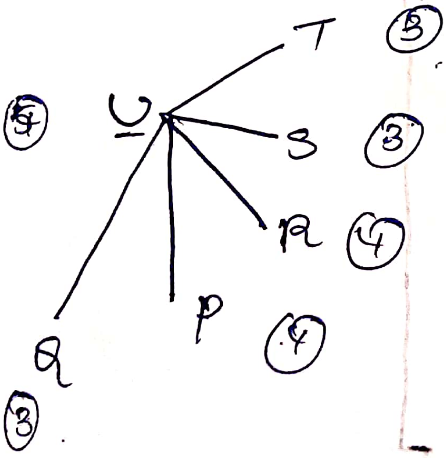












d $\Leftrightarrow$ P

b $\Leftrightarrow$ R

a $\Leftrightarrow$ S

e $\Leftrightarrow$ T

f $\Leftrightarrow$ U

c $\Leftrightarrow$ Q

5) Edge preference

a - b ⇔ S - R

a - f ⇔ S - U

a - e ⇔ S - T

b - c ⇔ R - Q

b - d ⇔ R - P

b - f ⇔ R - U

c - d ⇔ Q - P

c - f ⇔ Q - U

d - f ⇔ P - U

d - e ⇔ P - T

e - f ⇔ T - U

6) Adjacency matrix.

G_1

$$\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array}
\begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

G2

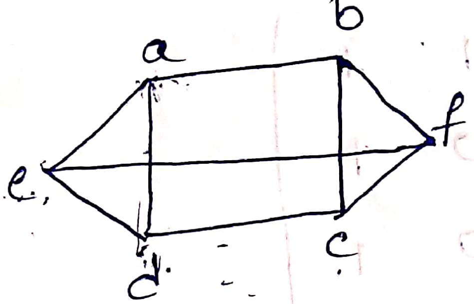
$$
\begin{array}{c|c}
 & \begin{matrix} S & R & Q & P & T & U \end{matrix} \\
\begin{matrix} S \\ R \\ Q \\ P \\ T \\ U \end{matrix} &
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 \\
0 & 01 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 0
\end{bmatrix}
\end{array}
$$

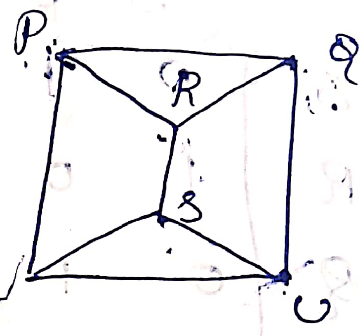
$\because G_1 = G_2$

$\because$ All the 6 properpties satisfied.

$\therefore$ The Given Graphs are Isomorphic//.

(3Q) G1 G2

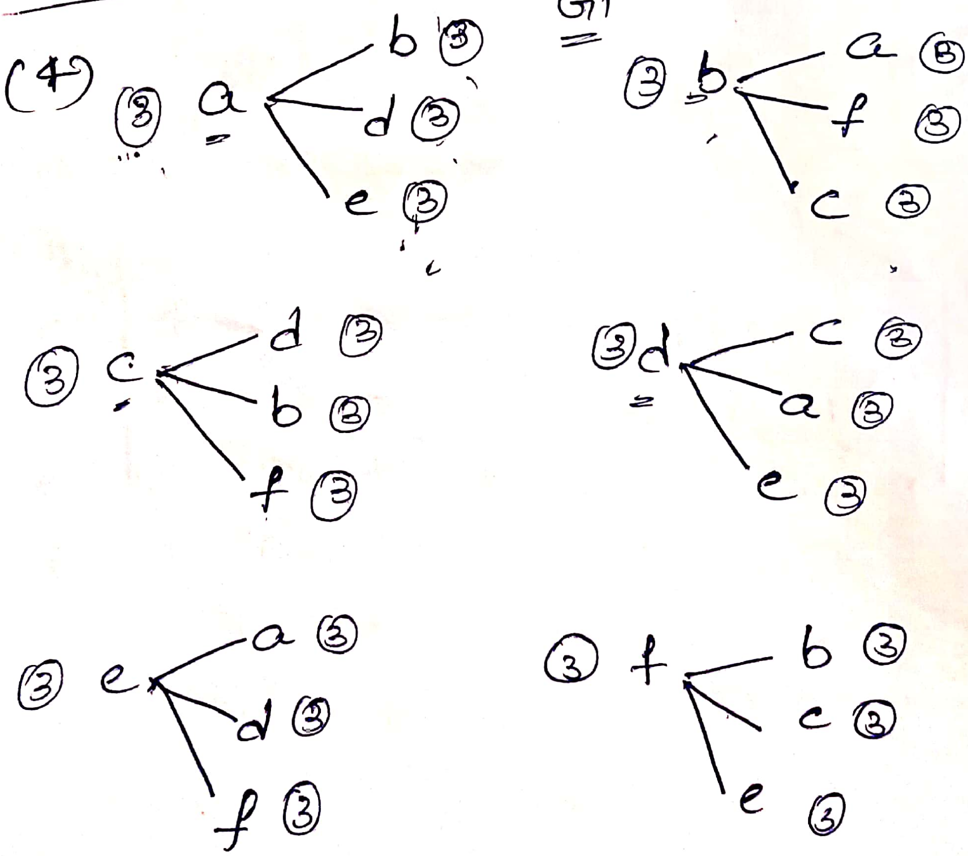




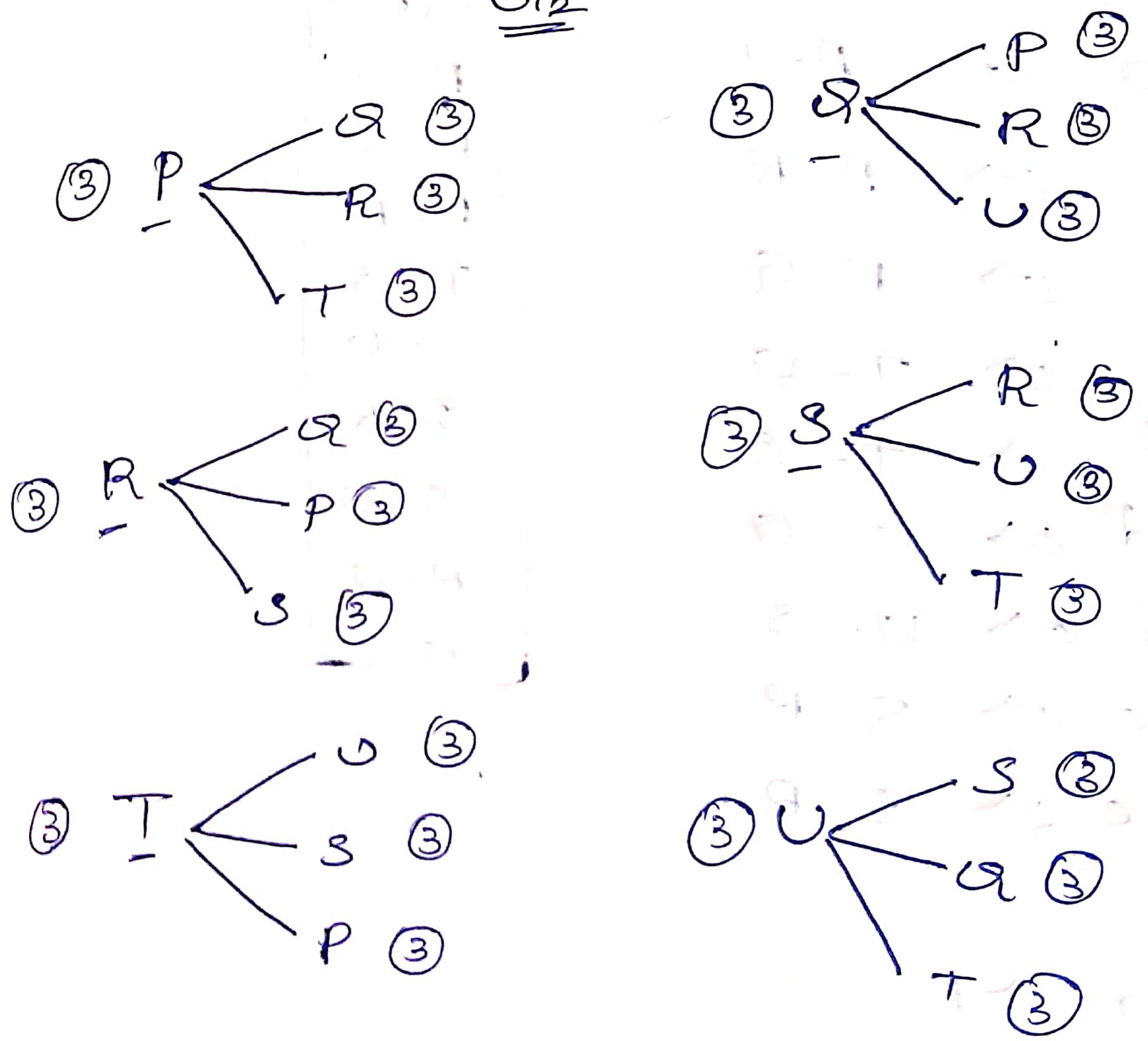
Sol:

G1	G2
(1) vertices = 6	1) vertices = 6
2) Edges = 9	2) Edges = 9
3) Degree =	3) Degree =
{a, b, c, d, e, f}	{P, Q, R, S, T, U}
{3, 3, 3, 3, 3, 3}	{3, 3, 3, 3, 3, 3}.

(4) G1



G_2



[using diagram]

$a-b \Leftrightarrow P-T$

$a-e \Leftrightarrow P-R$

$a-d \Leftrightarrow P-Q$

$b-c \Leftrightarrow T-U$

$b-f \Leftrightarrow T-S$

$c-d \Leftrightarrow U-Q$

$c-f \Leftrightarrow U-S$

$d-e \Leftrightarrow Q-R$

$f-e \Leftrightarrow S-R$

(or)

P − T
P − Q
P − R
T − S
T − U
S − R
S − U
R − Q
U − Q

$\therefore a \Leftrightarrow P$

$b \Leftrightarrow T$

$e \Leftrightarrow R$

$c \Leftrightarrow U$

$d \Leftrightarrow Q$

$f \Leftrightarrow S$

(6) Adjacency matrix

G_1

$$\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

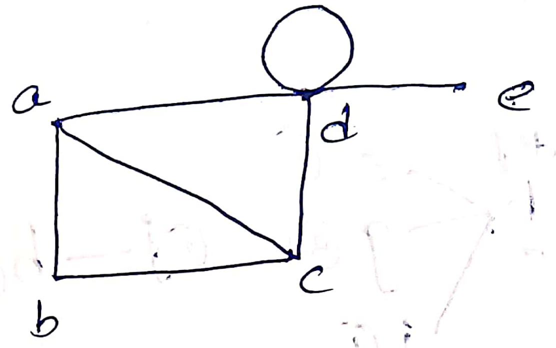
G_2

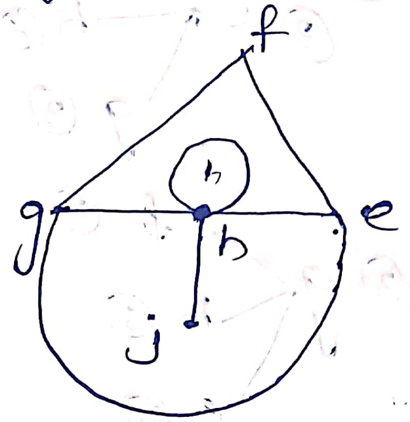
$$\begin{array}{c} \\ P \\ T \\ U \\ Q \\ R \\ S \end{array} \begin{array}{c} \begin{array}{cccccc} P & T & U & Q & R & S \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$\because G_1 = G_2$

$\therefore$ Isomorphic.

4) verify the following graphs are isomorphic or not.

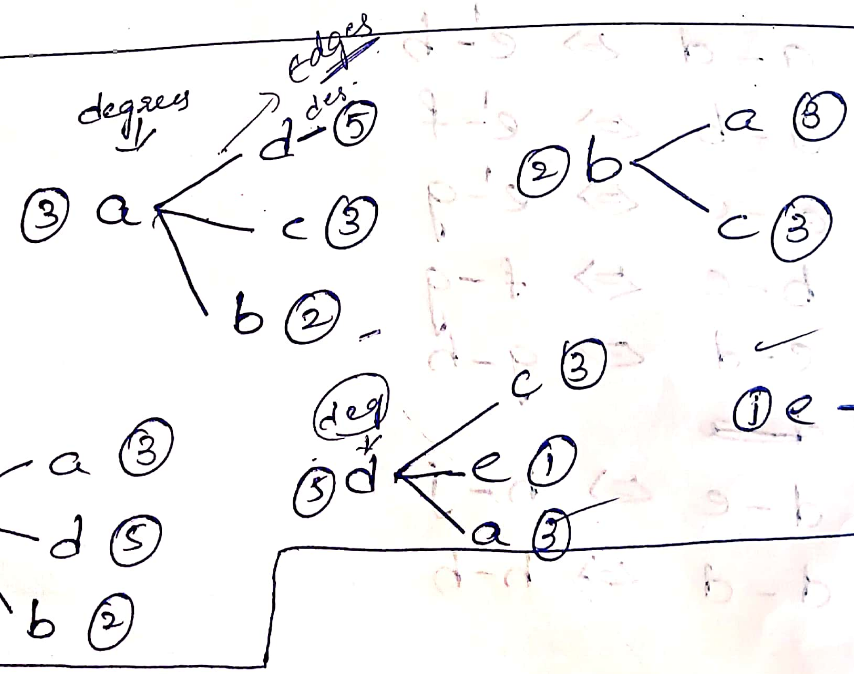




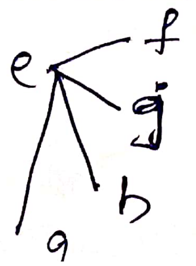
Sol:- 1) vertices = 5	Vertices = 5
2) edges = 7	edges = 7
deg(d)=3+2	~~A~~ loop [+2] ↑ degree
3) degree	
{a, b, c, d, e}	{e', f, g, h, j}
{3, 2, 3, 5, 1}	{3, 2, 3, 5, 1}

4)

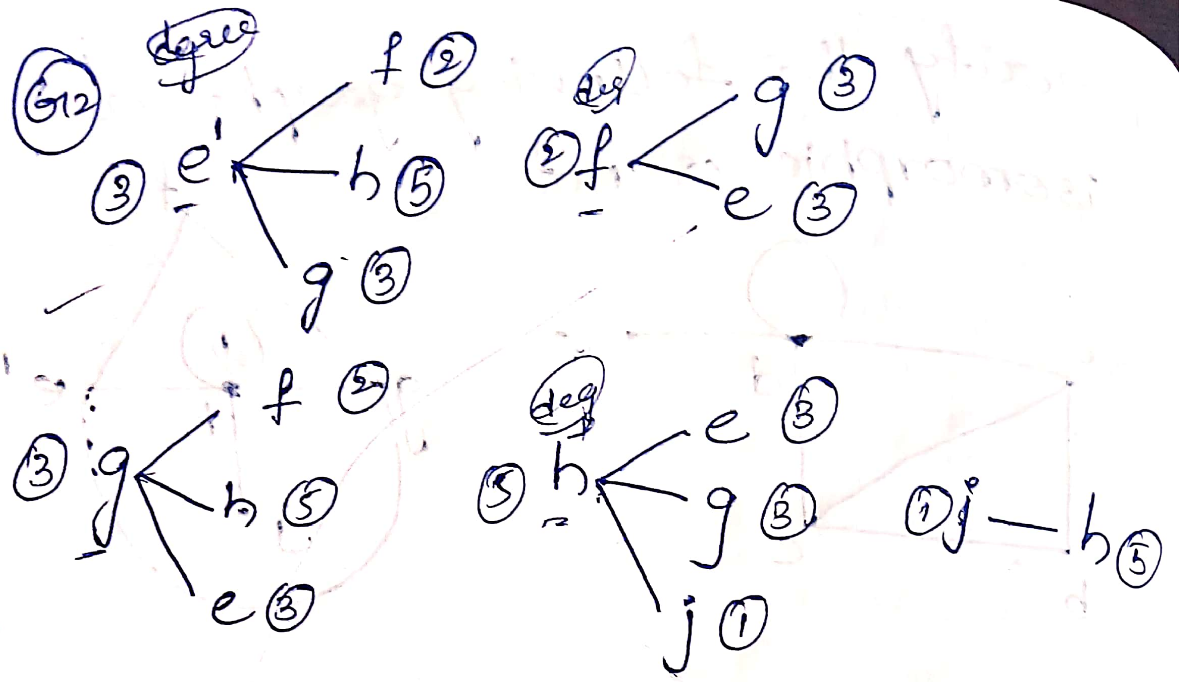
G1



G2



P.T.O



$a \Leftrightarrow e'$

$b \Leftrightarrow f$

$e \Leftrightarrow j$

$c \Leftrightarrow g$

$d \Leftrightarrow h$

(5)

$a-d \Leftrightarrow e'-h$

$a-b \Leftrightarrow e'-f$

$a-c \Leftrightarrow e'-g$

$b-c \Leftrightarrow f-g$

$c-d \Leftrightarrow g-h$

~~d-e~~

$d-e \Leftrightarrow h-j$

$d-d \Leftrightarrow h-h$

Adjacency matrix

$$
\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array}
\begin{array}{c}
\begin{array}{ccccc} a & b & c & d & e \end{array} \\
\begin{bmatrix}
0 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0
\end{bmatrix}
\end{array}
\qquad
\begin{array}{c} \\ e' \\ f \\ g \\ h \\ j \end{array}
\begin{array}{c}
\begin{array}{ccccc} e' & f & g & h & j \end{array} \\
\begin{bmatrix}
0 & 1 & 01 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 \\
01 & 1 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

$\because G_1 = G_2$ //

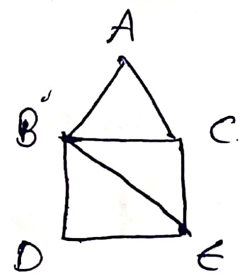
$\therefore$ Isomorphic //

Subgraphs

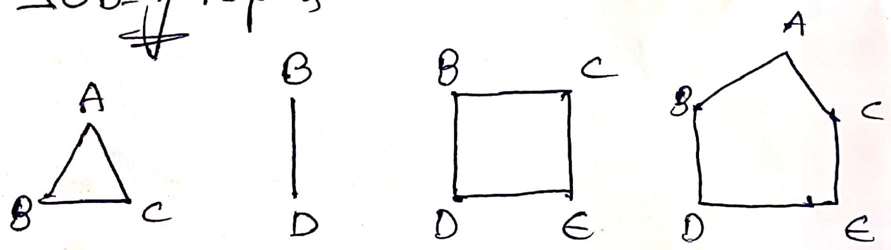
A graph $G' = (V', E')$ is called a subgraph of the given graph $G = (V, E)$ if

$V' \subseteq V$, $E' \subseteq E$

Given Graph



Sub-graphs



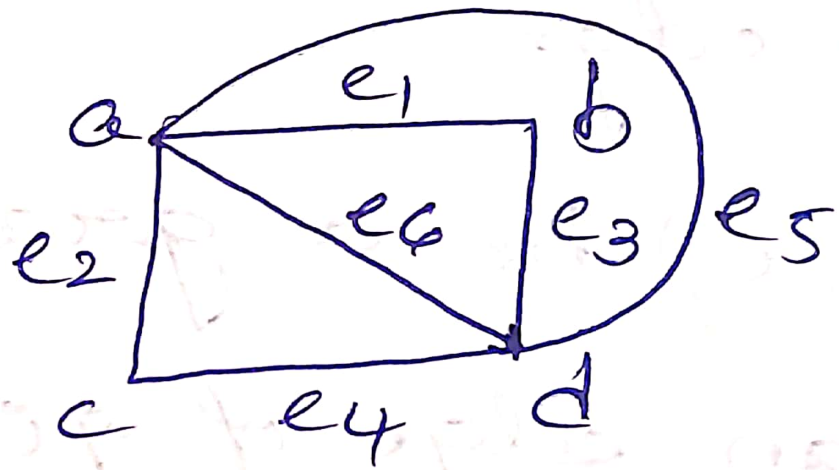
Deleting a vertex

let $G = (V, E)$ be a graph

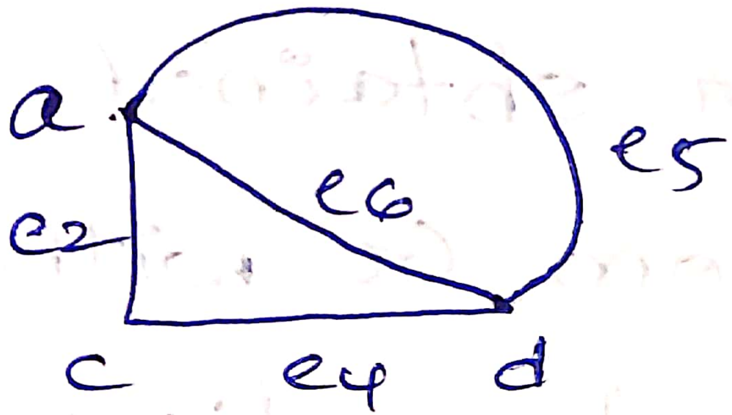
If v_i is a vertex of given graph G then $G - v_i$ is a subgraph of G, obtained by deleting the vertex v_i and all the edges incident with v_i

Ex:

1)



G - b

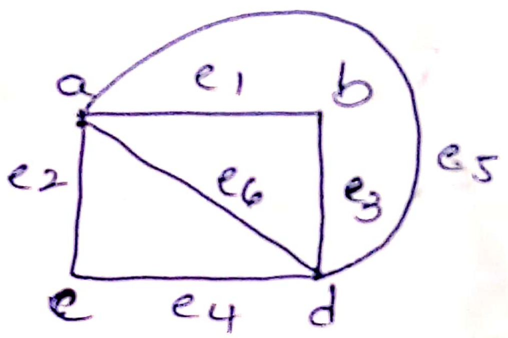


Deleting an edge

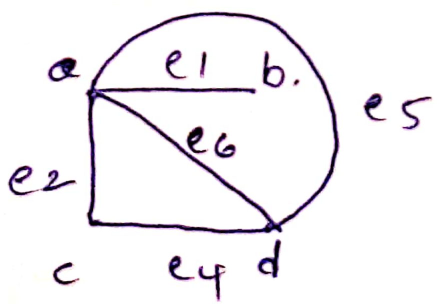
If $G = (V, E)$ be a graph

If e_i is an edge of given graph G, then $G - e_i$ is a subgraph of G, obtained by deleting e_i from G without deleting the end vertices of e_i

Ex:



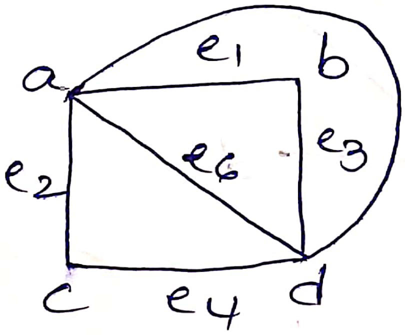
$G - e_3$



complement of a subgraph

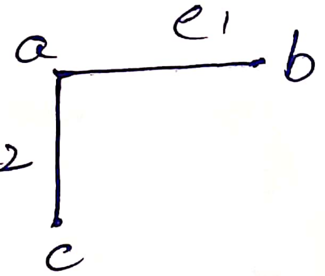
If H is a subgraph of a graph G, then the complement of H is defined as the graph with those edges between vertices of H that are in G but not in H.

Ex.:

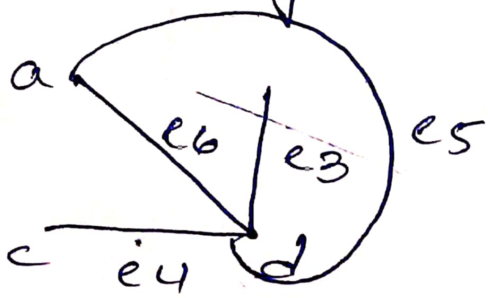


$\rightarrow G = (V, E)$

$G' = (V', E') \rightarrow$ subgraph



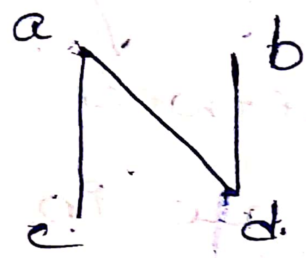
complement of $G' = G - G'$



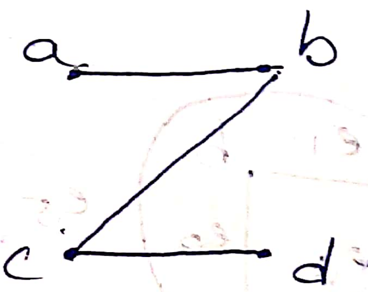
complement of a simple Graph

$G = (V, E)$ is $K_n - G$, where $|V| = n$.

Ex:



$\overline{G}$ =



Euler path (or) Euler walk (or) Trial

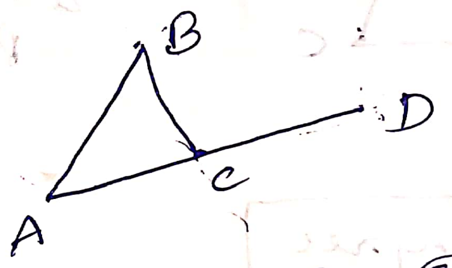
- It is a path in a graph that visits every edge exactly once.

→ start & end vertex are different

→ vertex can be repeated but edge can't be repeated.

→ A graph contains a Euler path iff it ~~is~~ exactly has 0 (or) 2 odd vertices.

Ex



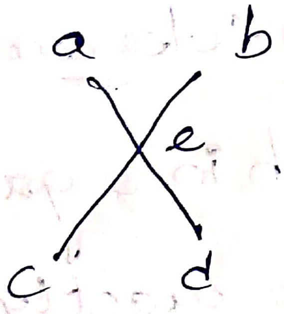
D - C - A - B - C

C - B - A - C - D

Vertices	Degree
A	2
B	2
C	3
D	1

2 odd degree

Ex:- 2)



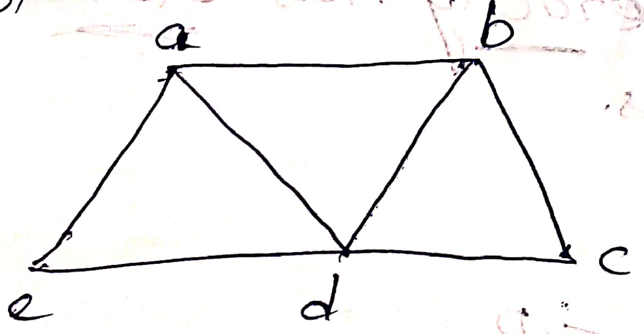
V	D
a	1
b	1
c	1
d	1
e	4

This Graph is not a euler path

must have exactly 2 odd vertices.

---

3)



→ It is a euler path

Vertex	degree
a	3
b	3
c	2
d	4
e	2

2 odd degree

Path

a - e - d - a - b - d - c - b

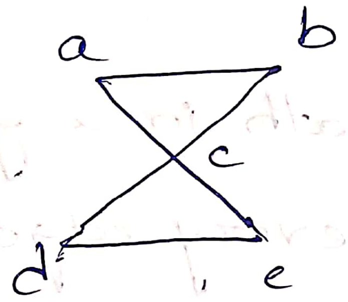
Edges

Euler circuit or Euler cycle

A Euler circuit is a euler path that starts and ends at same vertex.

→ A graph has a Euler circuit if each vertex degree is even

Ex



Vertices	degree
a	2
b	2
c	4
d	2
e	2

→ all even

a - b - c - e - d - c - a   same.

✓ a - b - c - e - d - c - a

∴ It is a Euler circuit

Euler Graph (or) Eulerian Graph

A graph is called as Euler graph if it contains Euler circuit

For Directed Graphs

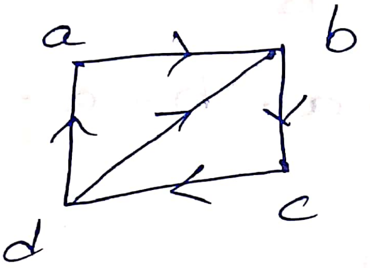
Euler path

It is a path in a graph that visits every edge exactly once.

→ A directed graph contains an Euler path iff it is connected and the in-degree of each vertex is the same as its out degree except at 2 vertices

For these 2 vertices, the indegree of 1 vertex is one larger than its out degree and the in-degree of the other vertex is one less than its out degree.

Ex:-



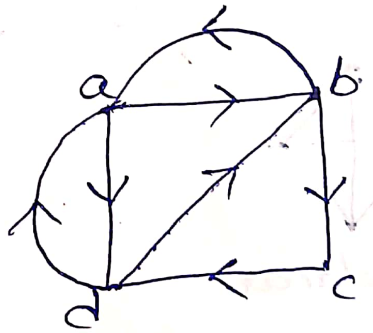
vertex	Indegree	Out-degree
a	1	1
b	2	1
c	1	1
d	1	2

Euler path is

d - b - c - d - a - b

For Euler circuit, the in-degree of each vertex in a connected graph is equal to its out-degree.

Ex:-



Vertex	In-degree	Out-degree
a	2	2
b	2	2
c	1	1
d	2	2

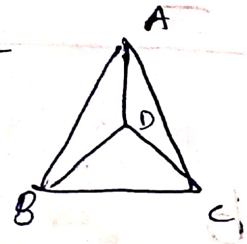
Euler circuit is

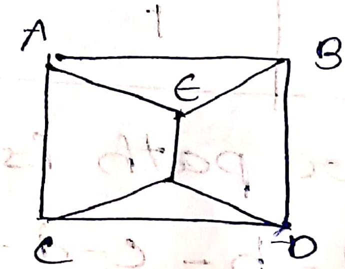
d - b - c - d - a - b - a - d

Planar Graph

It is a graph that can be drawn on a plane without crossing any of its edges, otherwise non-planar.

Ex

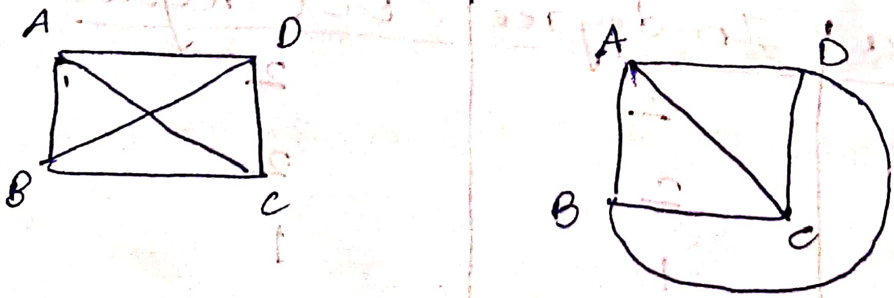




Some times, a non-planar graph can be converted to planar graph.

non-planar

planar



Euler's formula (or) Euler's theorem

A connected planar Graph G with $|V|$ vertices and $|E|$ edges has exactly $|E| - |V| + 2$ regions.

Region is a cycle that forms the boundary.

$$|R| = |E| - |V| + 2$$

(or)

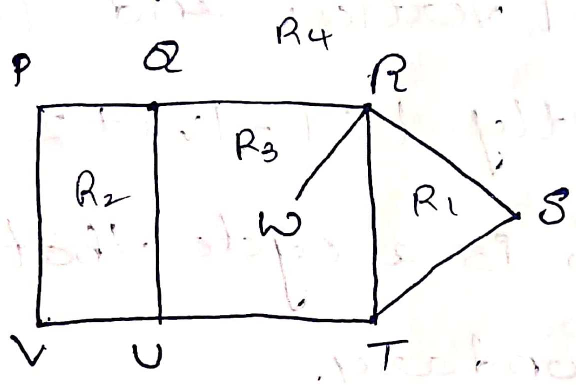
$$|V| - |E| + |R| = 2$$

→ Euler's formula

connected → must have edges for every vertex

$|R|$ = no. of regions
$|E|$ = no. of edges
$|V|$ = no. of vertices.

S.T the following graph G is verified by Euler's formula.



Sol: Vertices $= |V| = 8$

Edges $= |E| = 10$

Euler's formula:

$$|R| = |E| - |V| + 2$$

$$= 10 - 8 + 2 = 4$$

$\therefore$ Regions $\Rightarrow |R| = 4$

$\therefore$ No. of regions

$R_1 = R - S - T - R$

$R_2 = P - Q - U - V - P$

$R_3 = Q - R - W - R - T - U - Q$

$R_4 = P - Q - R - S - T - U - V - P$

$|V| - |E| + |R| = 2$

$8 - 10 + 4 = 2$

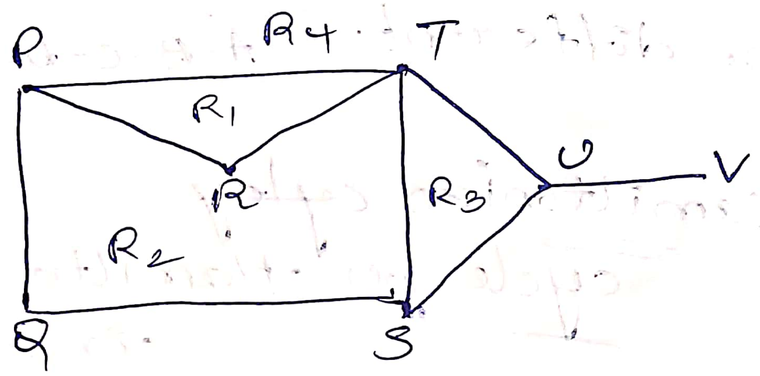
$-2 + 4 = 2$

$2 = 2$ [L.H.S = R.H.S]

∴ G satisfies Euler's formula.

---

2)



$|V| = 7, \ |E| = 9$

$|R| = |E| - |V| + 2$

$= 9 - 7 + 2 = 4$

∴ regions = 4

verify $|V| - |E| + |R| = 2$

$7 - 9 + 4 = 2$

$2 = 2$

∴ verified.

Hamiltonian path [vertex]

→ Each vertex of the graph will be visited exactly once.

→ edges can be repeated

→ start and end vertex must be different. A-B-C-D-E

Hamiltonian ~~cyley~~ cycle (or) Hamiltonian circuit

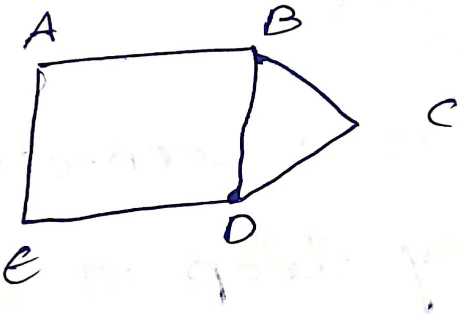
Each vertex of the graph will be visited exactly once except start vertex. A-B-C-D-A

→ start & End vertex must be same.

Hamiltonian Graph:

If the graph contains hamiltonian cycle (or) circuit then the corresponding graph is called as hamiltonian graph.

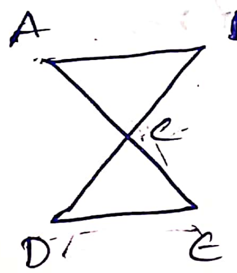
Ex:



A - B - C - D - E - A } → Hamilt cycle

C - D - E - A - B - C

Ex



A - B - C - D - E - C
vertex repeated

∴ It is not a hamiltonian cycle.

But It is a hamiltonian path

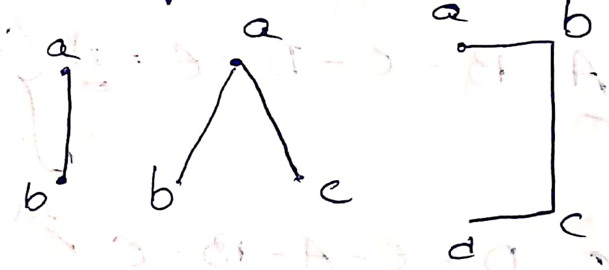
A - B - C - D - E

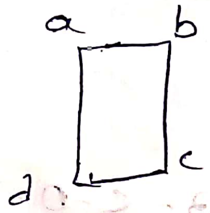
B.coz it covers all the vertices exactly once

Trees

A tree is a connected graph without any loop or circuits [closed]

Ex





→ not a tree.

Spanning tree

Let G be a connected Graph

→ If T is a subgraph of G

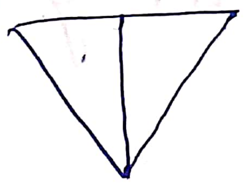
→ T is a tree [connected + no cycle]

→ T includes all vertices of G.

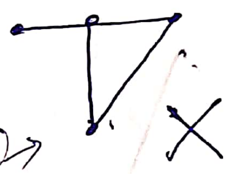
then T is a spanning tree.

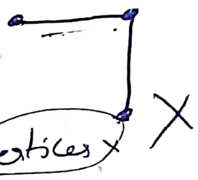
G

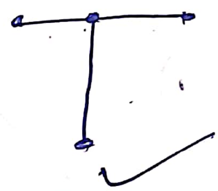
Ex:

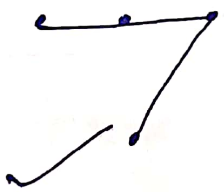


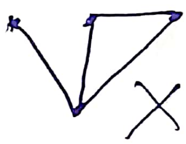
which of the following is a spanning tree of G.?

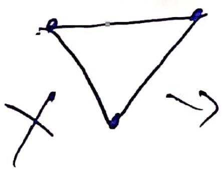
1)  cycle

2)  vertices

3) 

4) 

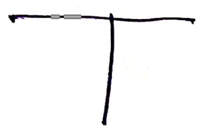
5)  cycle

6)  → subgraph → cycle

Note:- If a graph has 'n' vertices and 'm' edges then

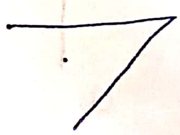
→ Its spanning tree T has a 'n' vertices and n-1 edges.

3 & 4 are spanning tree.



vertices = n = 4

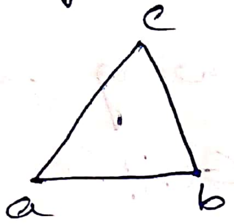
edges = n - 1 = 3



vert = 3

edges = 2

1) Find all the spanning trees of the graph.



remove 1-1 edge

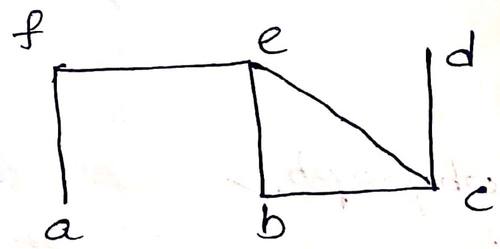
i)

3-v 2-e

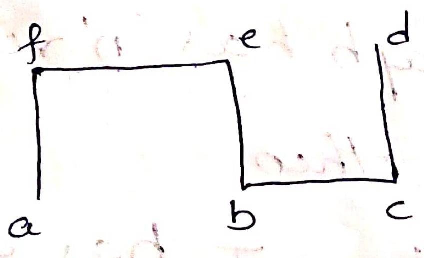
3-v 2-e

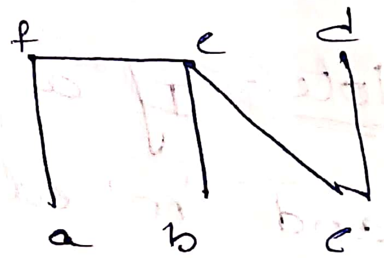
3-v 2-e.

2)

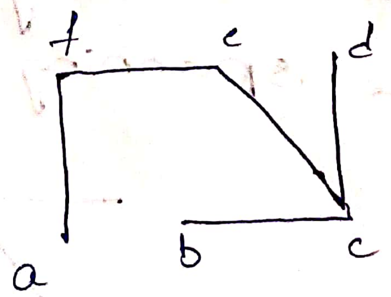


Sol:

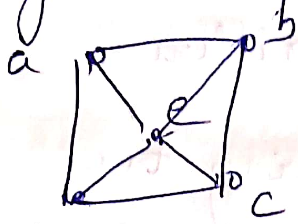




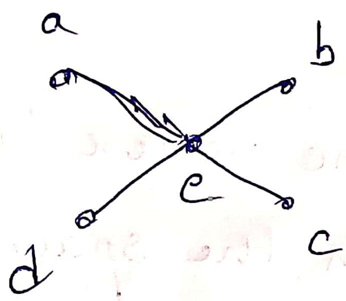
6-v
5-e

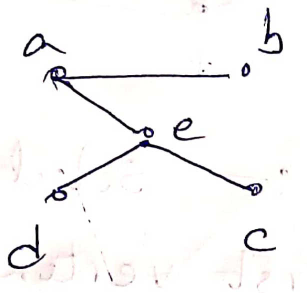


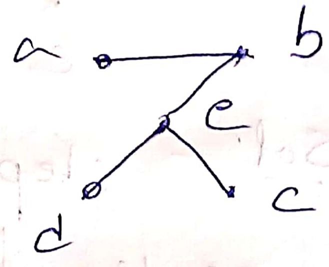
Q: Find all spanning trees of the following graph.

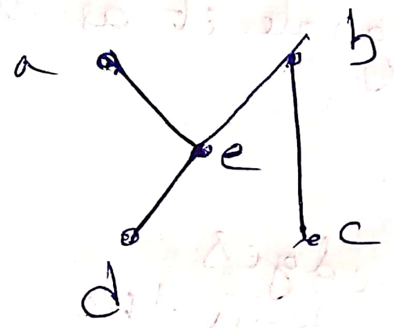


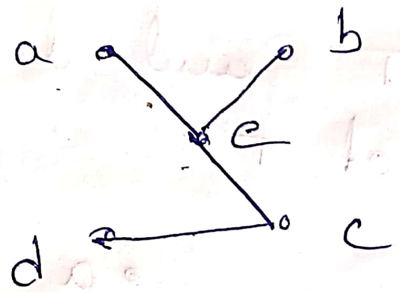
Sol The given graph has 5 vertices. So each spanning tree must have 5 vertices and $5-1=4$ edges.

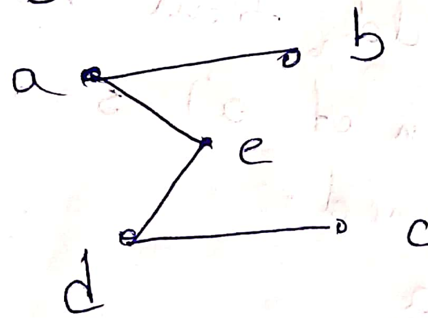


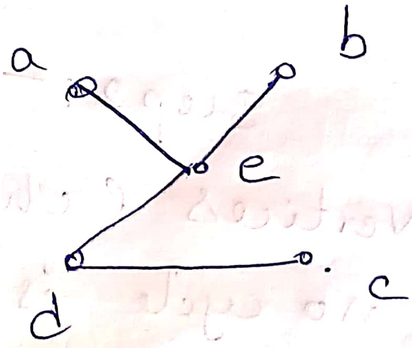


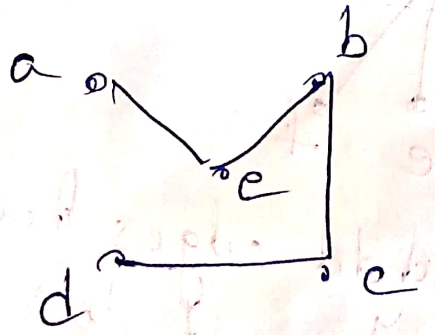












Properties of Trees

→ A tree with only one vertex is called a trivial tree. otherwise it is called a non-trivial tree.

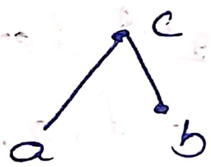
→ Edges of a Tree are called branches

→ There exists unique path between each pair of vertices in a tree.

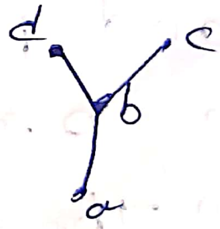
→ Tree with one vertex

→ Tree with 2 vertices

→ Tree with 3 vertices



→ Tree with 4 vertices

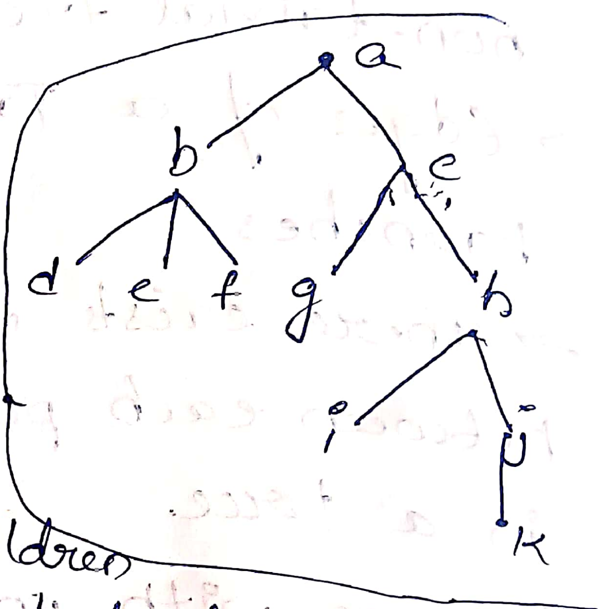


→ Vertices with degree 1 are called leaves of the tree.

Rooted tree

It is a tree in which a particular vertex is distinguished from the others and is called a root.

Ex :-



Here (i) a is a root and it at level 0.

(ii) b & c are children of a. They are at level 1.

(iii) a is said to be parent of b & c

(iv) 2nd level vertices are d, e, f, g, h. parent of d, e, f is b and parent of g, h is c.

(v) b and c are sibblings

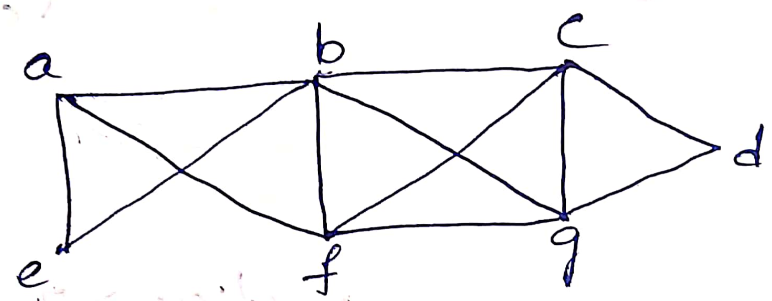
(vi) The ancestors of k are a, c, h, j

(vii) The descendants of c are g, h, i, j, k.

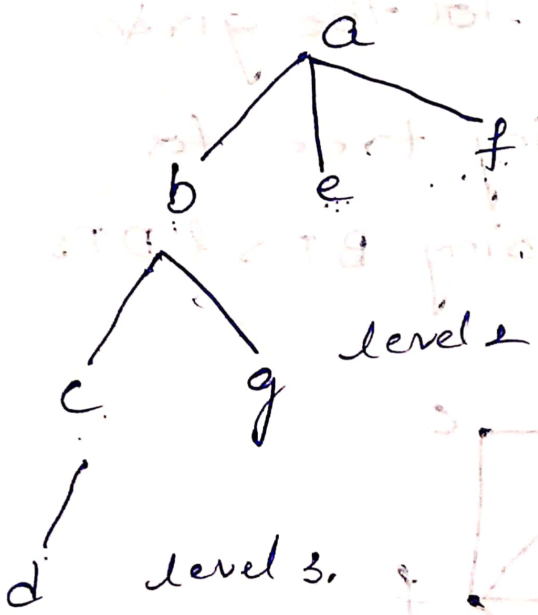
Breadth - First search

BFS Algorithm :

Find spanning tree for the following graph by using BFS & DFS.



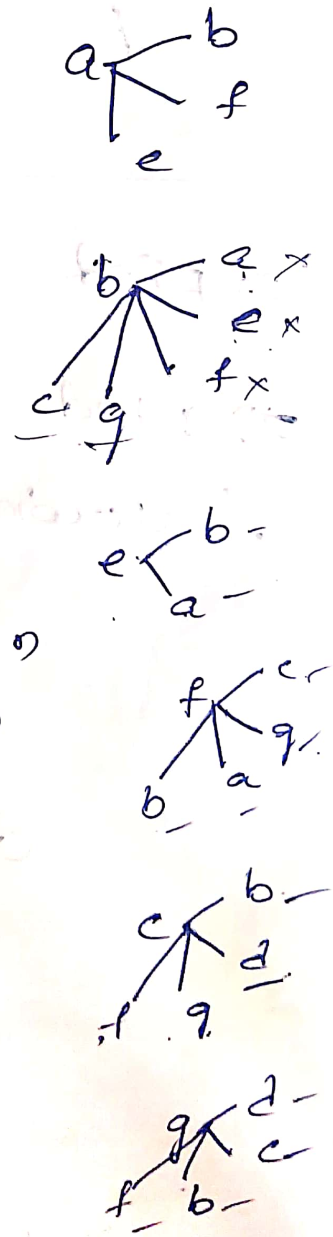
B.FS [for all]



vertices → 7 → n

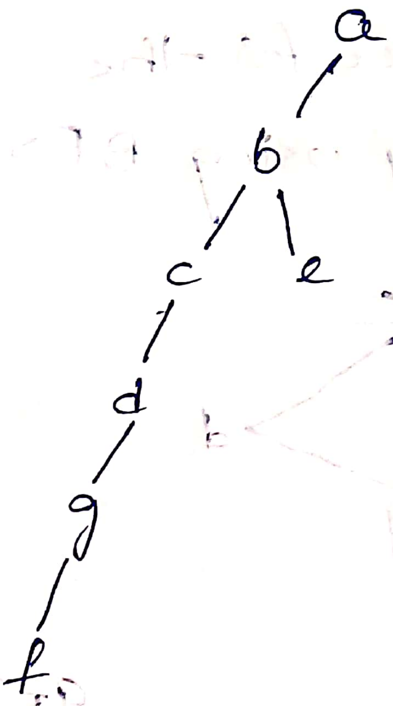
edges → (n-1)

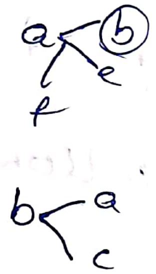
→ 6



Depth-first Search (DFS)

single-single





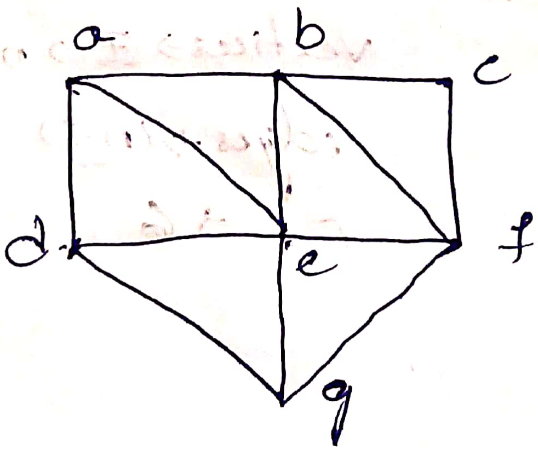
vertices → 7

Edges → 6

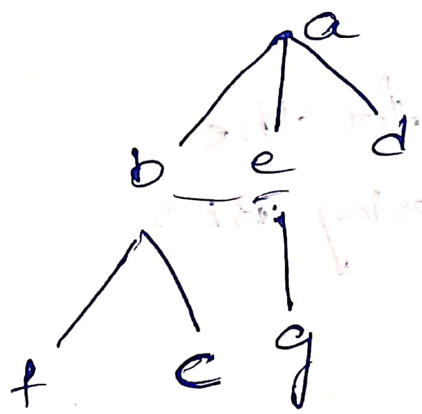
---

2) ~~Find~~ BFS & DFS ~~for the~~ given

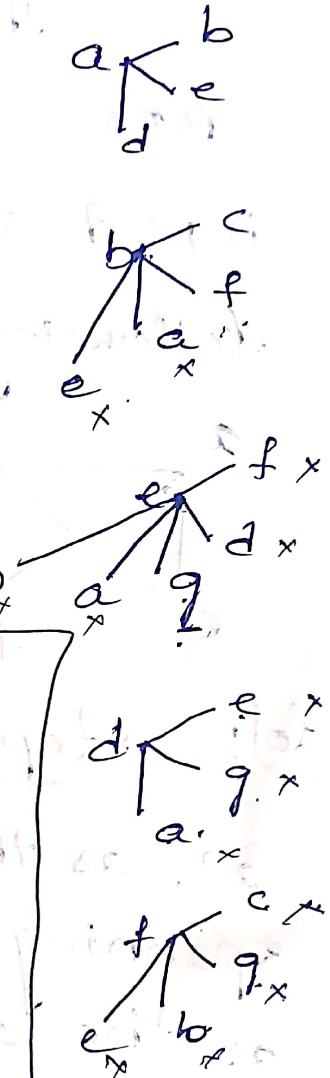
2) Find the spanning tree for following graph using BFS & DFS



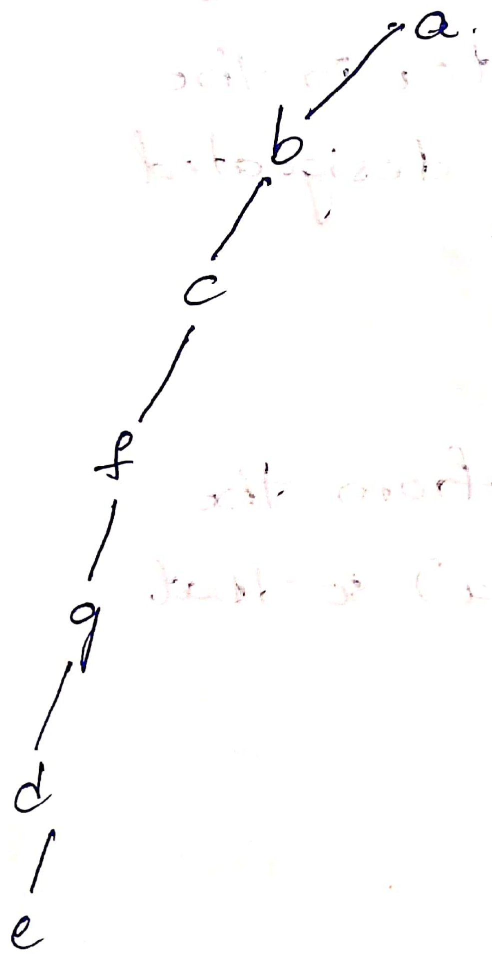
BFS



vertices → 7
edges → 6



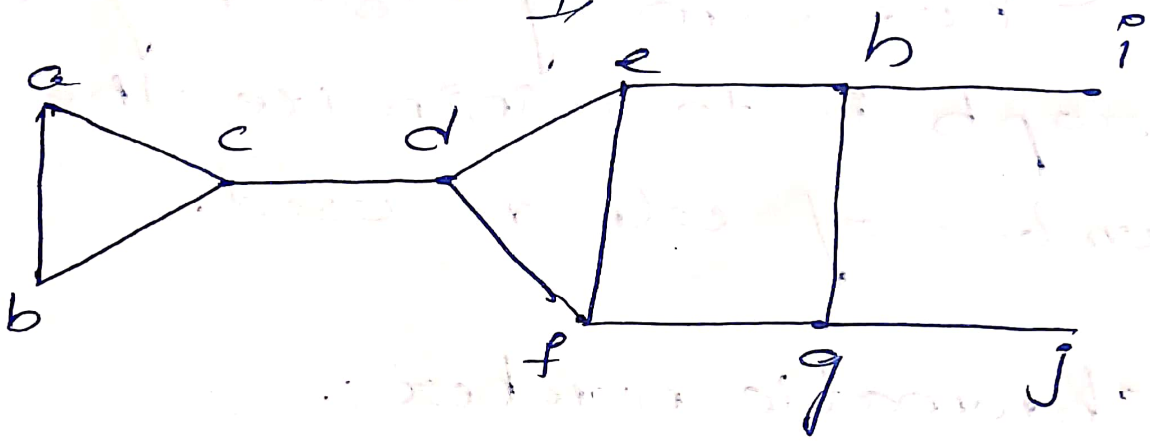
DFS



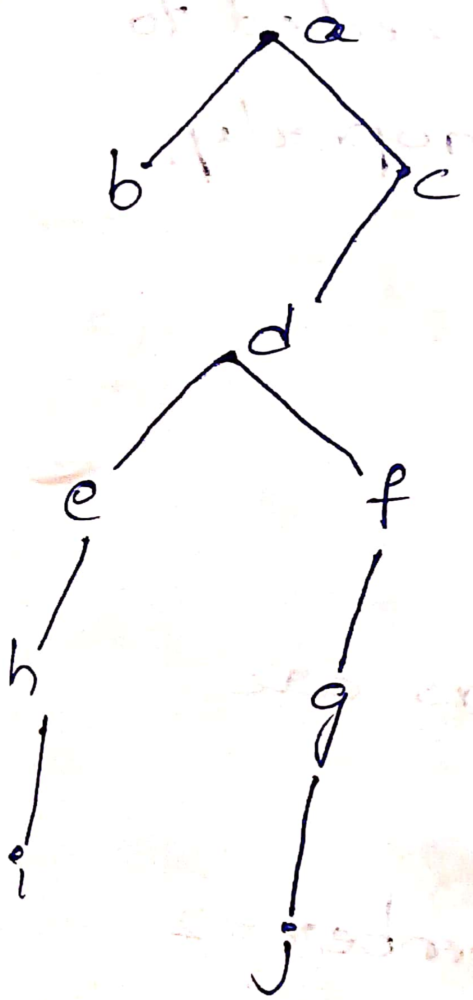
Vertices → 7
edges → 6

$1+4+9+3+17+23 = 57$

BFS & DFS of Graph



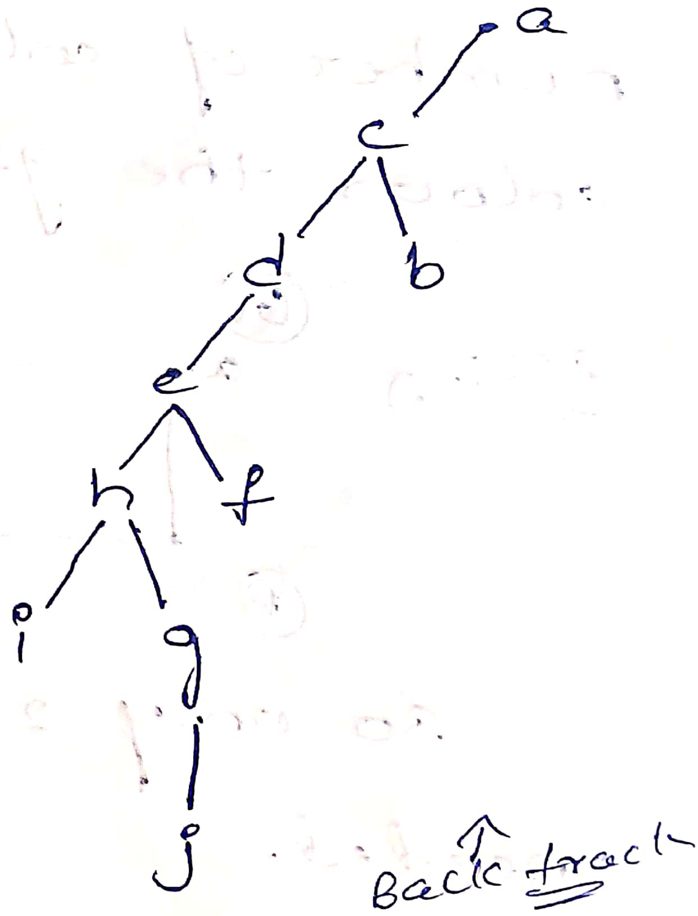
B.F.S.



Vertices → $n = 10$

Edges → $n-1 = 9$

DFS



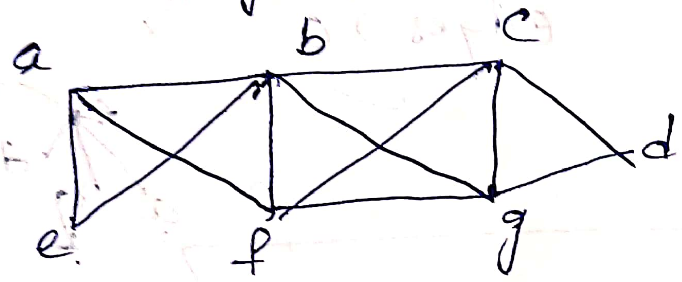
Vertices → $n = 10$

Edges → $n-1 = 9$

Theory Part

BFS

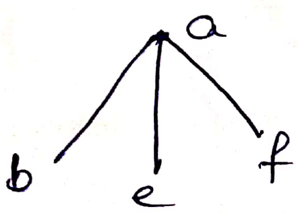
1) Find spanning tree for the following graph by using BFS



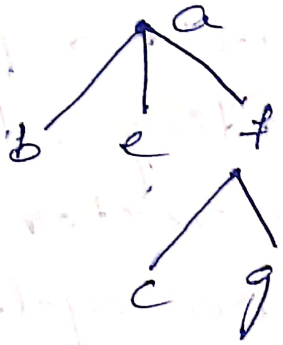
Sol:- step:- 1 select one vertex "a", as the first vertex in the spanning tree T and designated as the root of T.

• a.

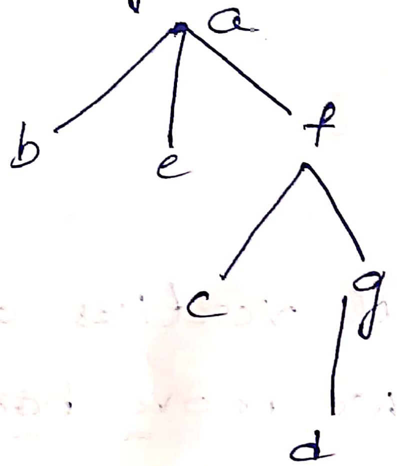
step: 2 Add edges from the vertices (children of a) so that no cycle is formed.



step:3:- Add Edges from the vertices (children of b, e & f) so that no cycle is formed.



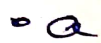
step-4:- Add edges from the vertices (children of c & g) so that no cycle is formed.



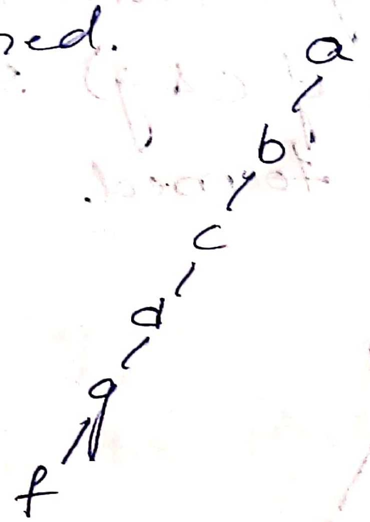
Since all vertices are visited, this is the required spanning tree.

DFS [Theory part for same Graph]

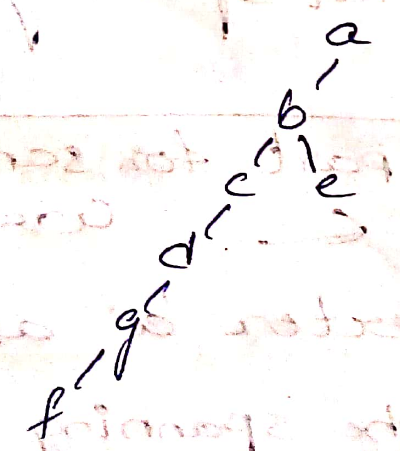
Sol:- Select one vertex 'a' as the first vertex in the spanning tree T and designate it as the root of T.



Step 2:: Now build a path by successively adding edges from 'a' so that no vertex is repeated and no cycle is formed.



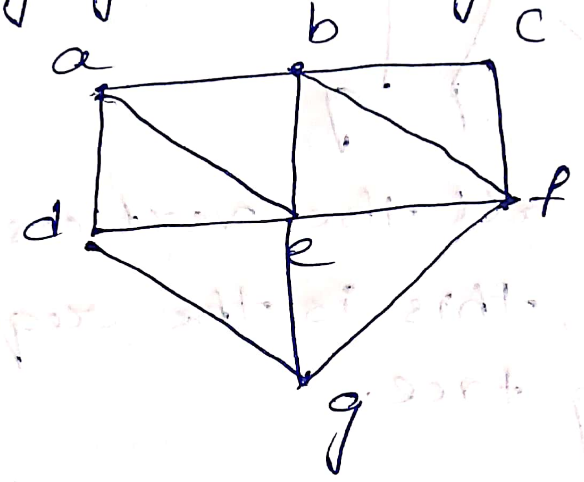
Step:-3 As all vertices are not visited, we move back along the path, and at b, add the edge e.



Now all vertices are visited and hence it is the required spanning tree.

Find the spanning tree for the following graph using BFS & DFS.

Sol:-

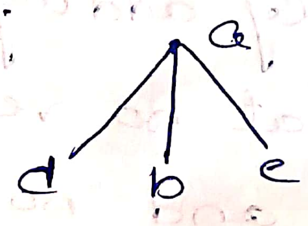


BFS

Step:-1: Select one vertex 'a' as the first vertex in the spanning tree T and designate it as the root of T.

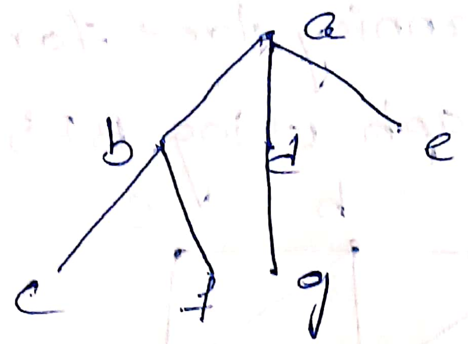
∴ a

Step:2:- Add edges from vertices (children of a), so that no cycle is formed and no vertex is repeated.



Step:-2:-

Add edges from vertices (children of b,d,e) so that no cycle is formed and no vertex is repeated.



Since all the vertices are visited, this is the required spanning tree.

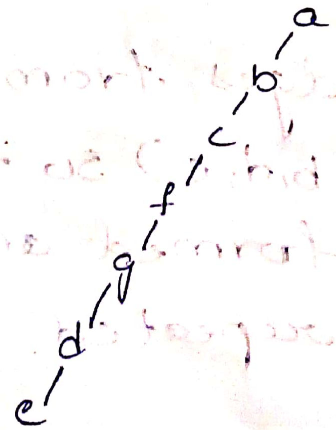
<u>DFS</u> :- [for same Graph].

<u>step</u>:-1 select one vertex 'a' as the first vertex in the spanning tree 'T' and designate it as the root of the tree 'T'

a

<u>step</u>:-2 :-

Now build a path by successively adding edges from 'a' so that no vertex is repeated and no cycle is formed.



Since all vertices are visited, this is the required spanning tree.

Minimal's spanning Tree.

Defn:- Let 'G' be a weighted Graph. A spanning tree of G with the smallest weight is called a minimal spanning tree.

Kruskal's Algorithm is used to calculate minimal spanning tree.

Procedure:

Setp:-1

The edges of the given Graph 'G', are arranged in the order of increasing weights [small first]

Step:-2

An edge ~~of G~~' with minimum weight is selected as an edge of the required

Spanning tree.

Step:3:

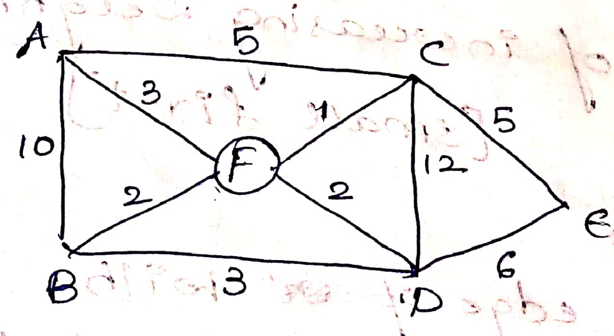
Edges with minimum weight are added to the spanning tree. So that no cycle is formed.

Step:4

Pick the next smallest edge and check if it forms a cycle or not. If doesnot forms cycle add it, otherwise discard the edge

---

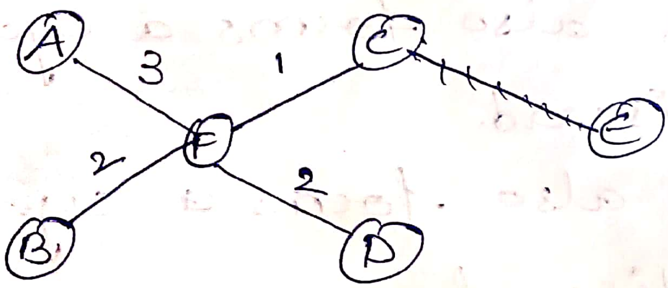
Q) use kruskal's algorith to find a minimal spanning tree for the following weighted graph.



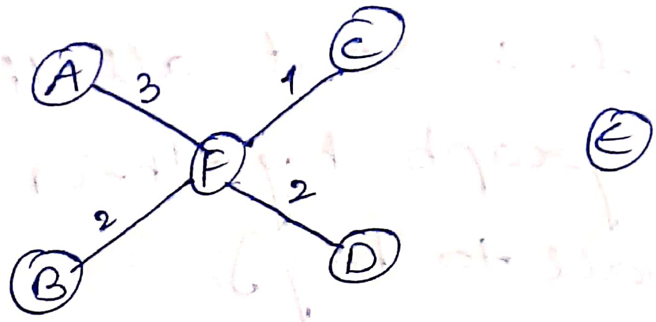
sol:- step:- 1. :- sort all the edges of the graph by their weights [small to big].

Edge	weight
C - F	1
F - D	2
F - B	2
A - F	3 , B-D → 3.x
A - C	5 x
C - E	5
D - E	6 x
A - B	10 x
C - D	12 x

Next write the vertices accordingly given in the Question



C-F, F-D, A-F, can be drawn because they don't form a cycle.

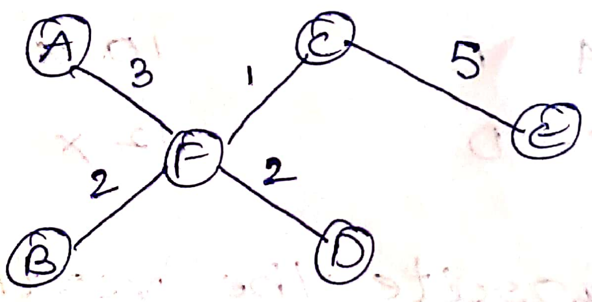


B-D can't be drawn, because it forms a cycle, so discard that edge.

Next go for next one

A-C also forms a cycle.

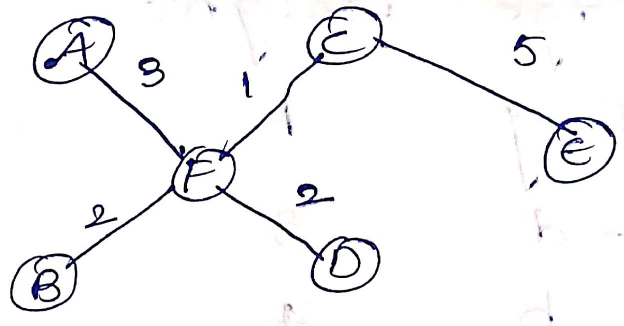
so next one is C-E [doesn't form cycle]



D-E also forms a cycle, so discard

A-B also forms a cycle, so discard.

C-D also forms a cycle, so discard.

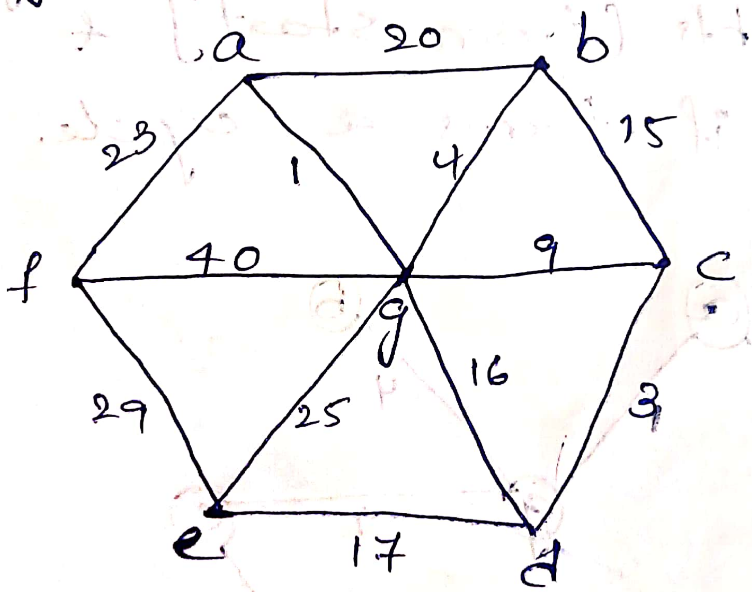


which is the required minimal spanning tree, weight of this tree is

$1+3+2+2+5 = 13.$ //

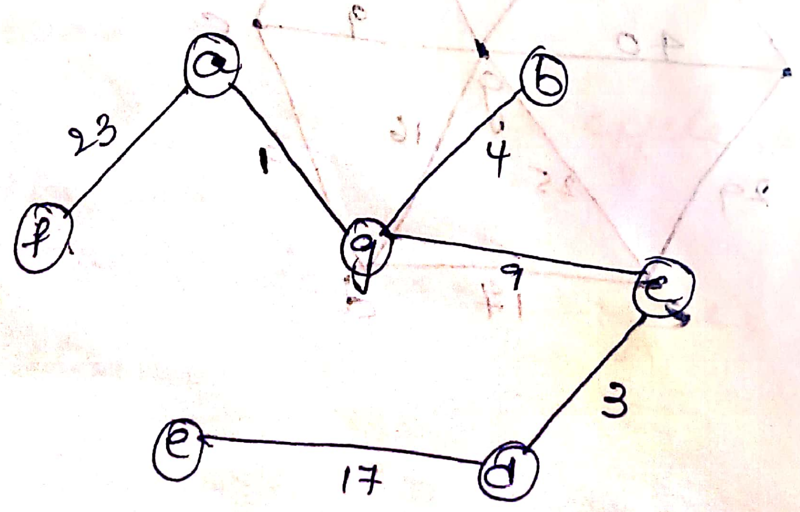
---

2) Find minimal spanning tree for the following weighted graph using Kruskal's algorithm



edge	weight
a-g	1
c-d	3
g-b	4
g-c	9
c-b	15 x
g-d	16 x
e-d	17
a-b	20 x
a-f	23, g-e - 25 x
f-e	29 x
f-g	40

Draw the graph according to weights [from start] & discard if forms a cycle.



discardes edges

c-b, g-d, a-b, g-e, f-g, f-e

∴ weight of this tree is

1 + 4 + 9 + 3 + 17 + 23 = 57 //

Graph colouring

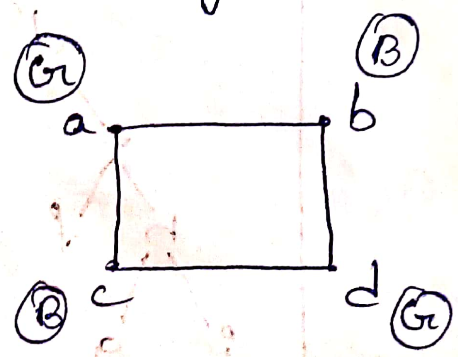
It means assigning colours to all the vertices of a graph so that no 2 adjacent vertices have the same colour.

→ Our aim of colouring a graph is to minimize the number of colours used.

chromatic number :-

It is the smallest number of colours needed to colour the graph properly.

ex:- 1)



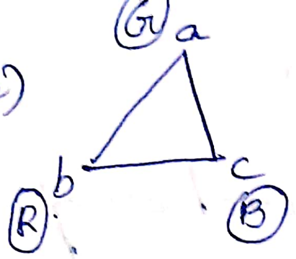
so only 2 colours are needed.

∴ chromatic number = 2.

[Denoted by $\chi(G)$]

R → Red
B → Blue
G → Green
V → violet.

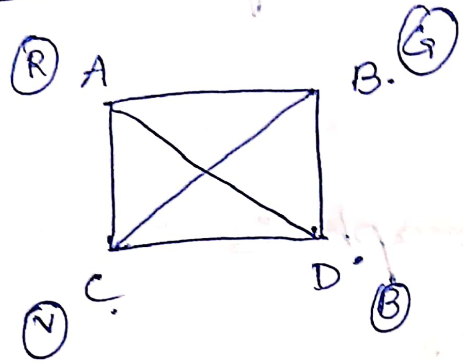
2)



chromatic no. = 3.

---

3)

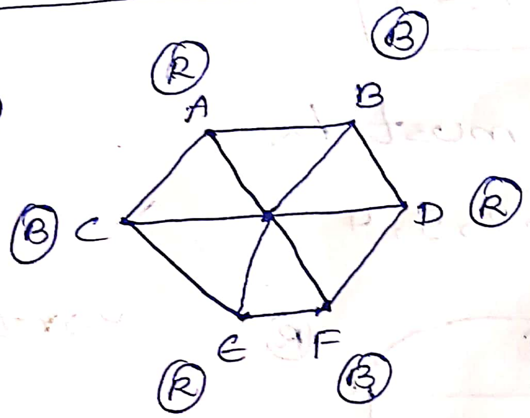


χ(Graph) = 4.

chromatic no. = 4.

---

4)

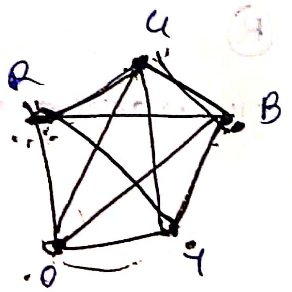


Chromatic number = 2.

---

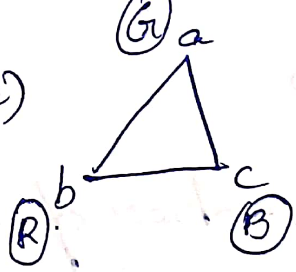
5) K_5.

K_6
↓
chromatic → 6



chromatic number = 5.

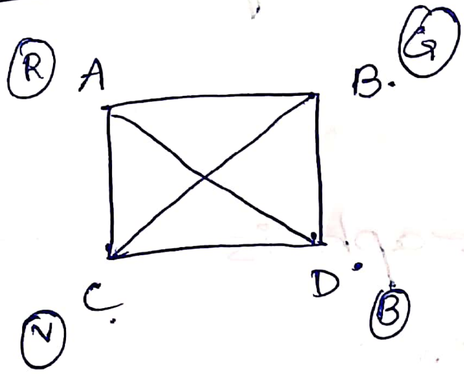
2)



R → Red
B → Blue
G → Green
V → violet.

chromatic no. = 3.

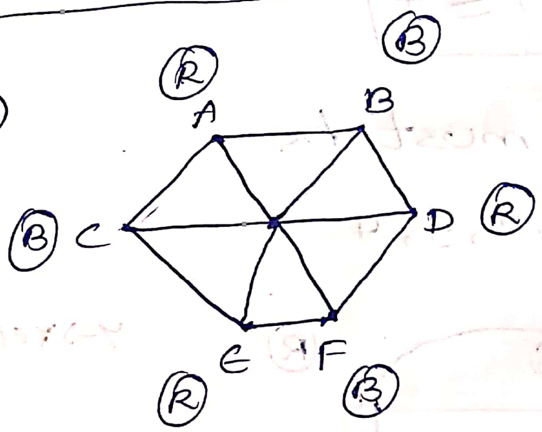
3)



χ(Graph) = 4.

chromatic no. = 4.

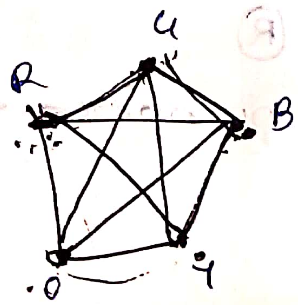
4)



chromatic number = 2.

5) K_5.

K_6
↓
chromatic → 6.



chromatic number = 5.

Four color Theorem

The chromatic number of a planar graph is no greater than 4

(or)

Every planar graph is four-colorable.

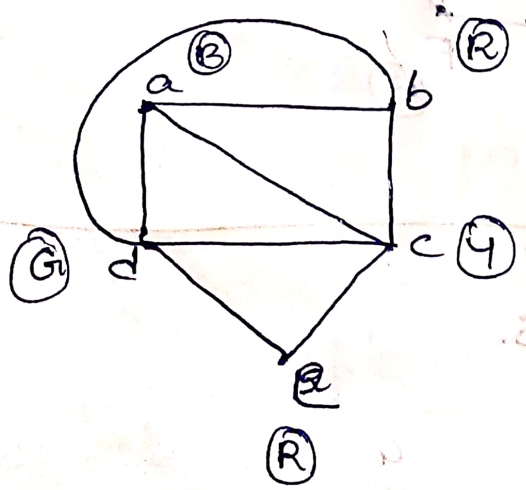
$$\boxed{\chi(G) \le 4}$$

means $\chi(G)$ must be

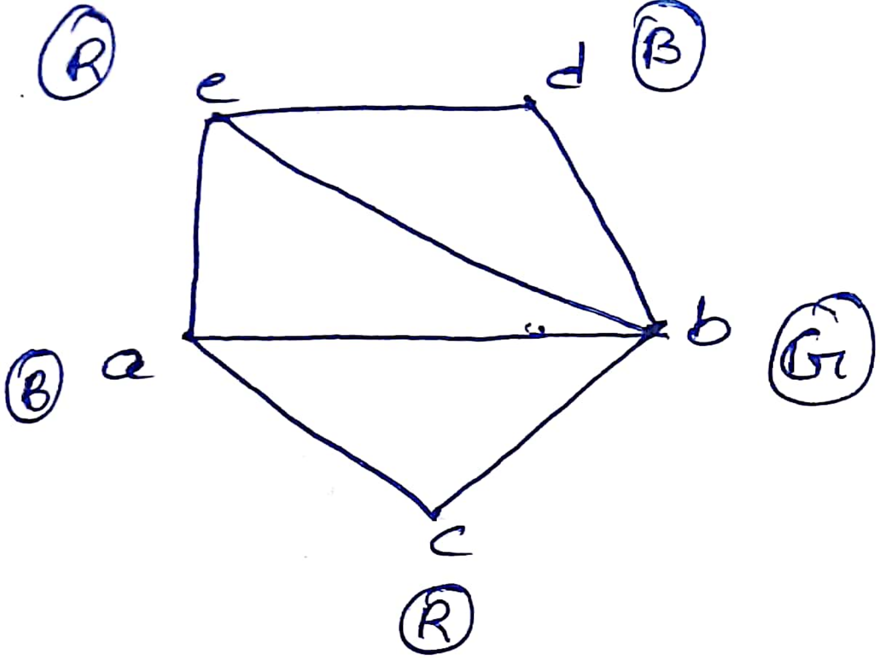
$\chi(G) = 1, 2, 3, 4$

Ex:-

Y → Yellow



chromatic number = 4.



R → Red

G → Green

B → Blue

chromatic number = 3