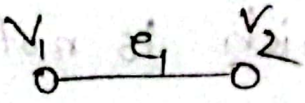
Graphs

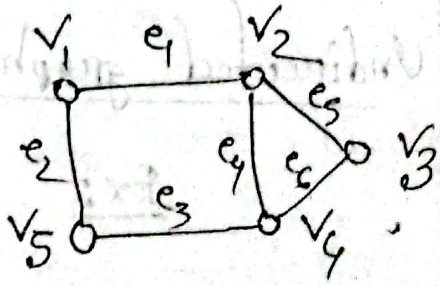
Def^n: A graph G is mathematical structure consisting of two sets V and E where V is a non-empty set of vertices and E is non-empty set of edges.

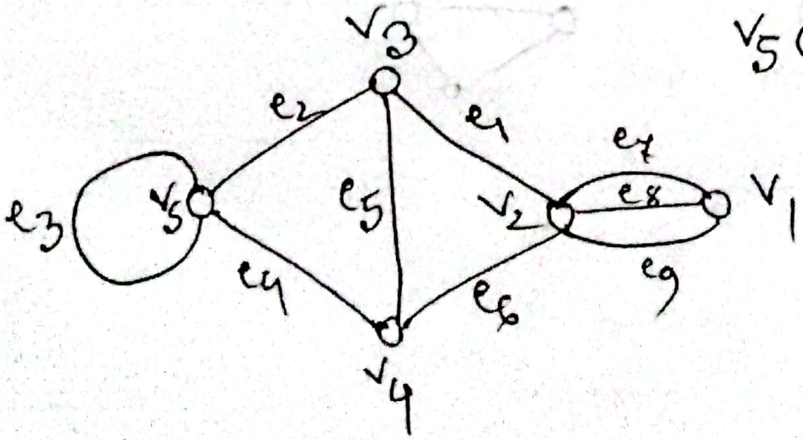
→ Each has either one or two vertices associated with it, called it's endpoints.

→ An edge is said to connect it's end-points.

Example:







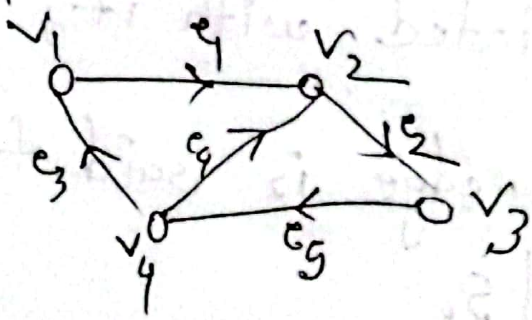
Trivial Graph: Only one vertex, no edge

Ex: 

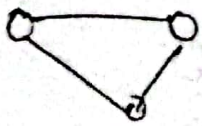
Null Graph: n vertices, no edge

Ex: 

Directed Graph: consist the direction of edges then this is called directed graph.

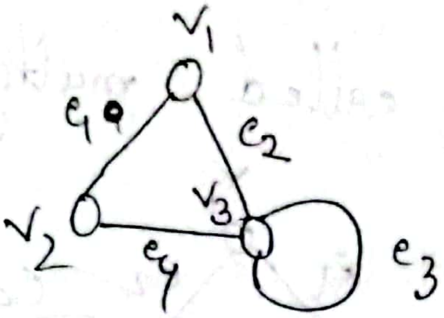
Ex: 

Undirected graph: which is not directed

Ex: 

Self loop in a Graph: edge having the same vertex as both it's end

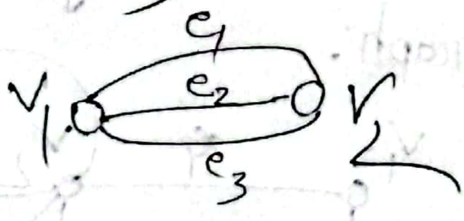
Ex:



Proper edge : An edge which is not self loop.

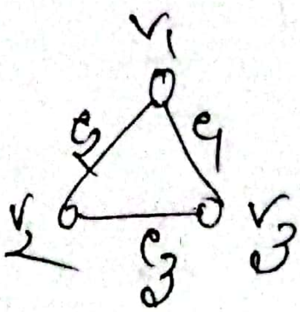
Multiedge : two or more edges having identically end point.

Ex:



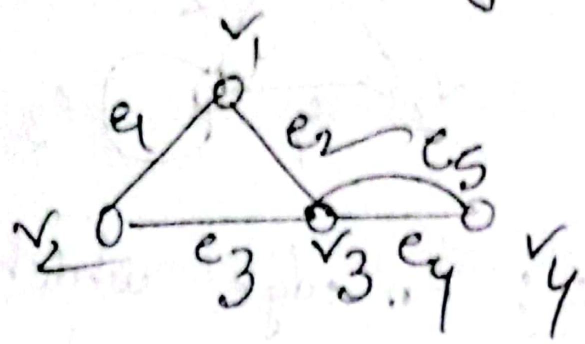
Simple Graph: A graph does not contain any self loop and multiedge.

Ex:



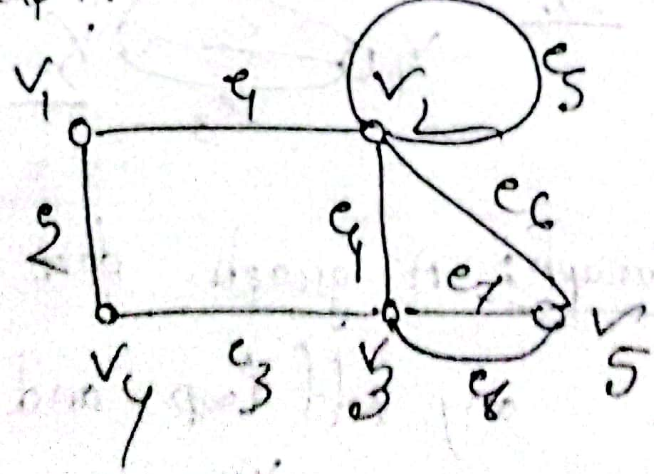
Multigraph: A graph does not contain any self loop but contain multiedge is called multigraph.

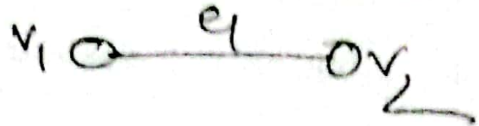
Ex:



Pseudo Graph: A graph contain both self loop and multiedge is called pseudo graph.

Ex:





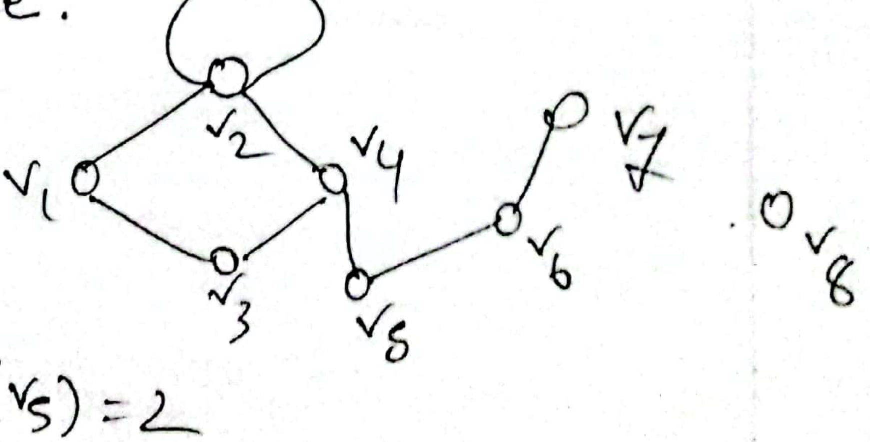
e_1 is said to be incidence of v_1 and v_2.

Two ventices are said to be adjacent if there exist an edge joining this ventices.

Degree of verrtex:

The degree of veretex V in a graph or writlen as $d(v)$ is equal to numberr of edges which are incident on V with self loop counted twice.

Ex:



$d(v_1) = 2$

$d(v_2) = 4$

$d(v_3) = 2$

$d(v_4) = 3$

$d(v_5) = 2$

$d(v_6) = 2$

$d(v_7) = 1$

$d(v_8) = 0$

* Isolated vertex: degree of vertex = 0

* Pendant vertex: degree of vertex = 1

* Finite Graph: Finite numbers of vertices and edges.

* Infinite Graph: Infinite numbers of vertices and edges.

(c) Draw the graph:

H : $V = \{v_1, v_2, v_3, v_4, v_5\}$

$E = \{(v_1, v_4), (v_2, v_3), (v_3, v_4), (v_4, v_5)\}$

$f : V \rightarrow \{\text{cities}\}$

$g : E \rightarrow N$

$f(v_1) = $ Madrid, $g((v_1, v_4)) = 199$

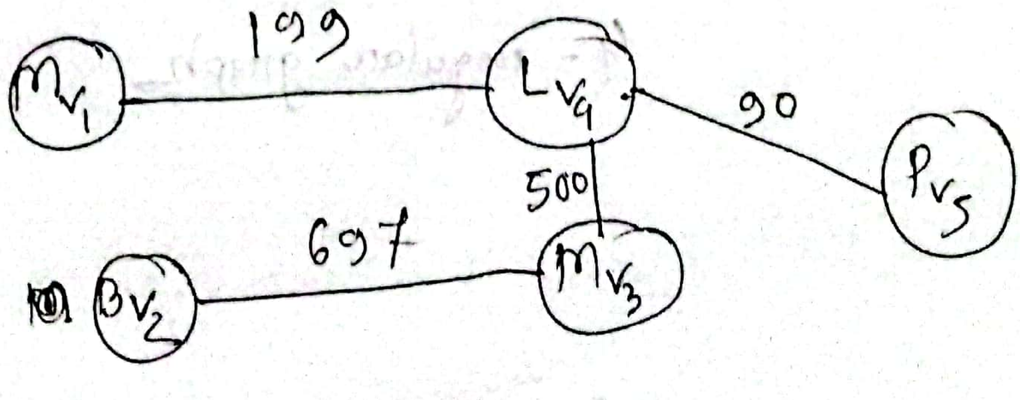
$f(v_2) = $ Barca, $g((v_2, v_3)) = 697$

$f(v_3) = $ Milan, $g((v_3, v_4)) = 500$

$f(v_4) = $ Liverpool, $g((v_4, v_5)) = 90$

$f(v_5) = $ Paris.

Soln:



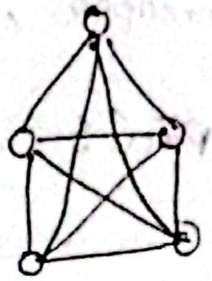
Complete Graph: if each vertex is connected to every vertex

Ex:

~~Regl~~ Regular Graph: if every vertex has the same degree

If the degree is k, then it's called k-regular graph

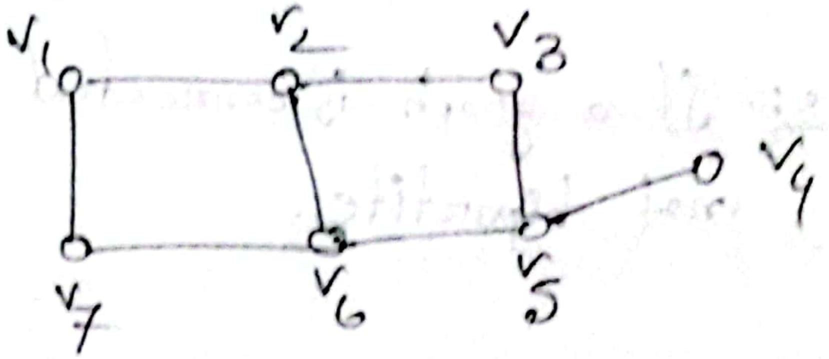
Ex:



4-regular graph

Bigraph (or Bipartite): If the vertex set V of a graph G can be partitioned into two non-empty disjoint subsets X and Y in such a way that edge of G has one end in X and one end in Y. Then G is called bipartite.

Ex:



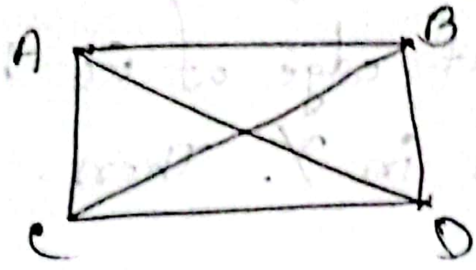
$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$

$X = \{v_1, v_6, v_3, v_4\}$

$Y = \{v_2, v_7, v_5\}$

Connected Graph: An undirected graph is said to be connected if there is a path between every two vertices.

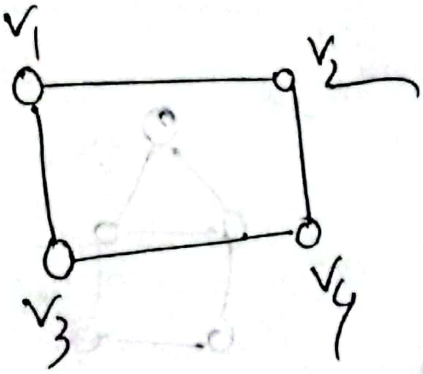
Ex:



Note: If a graph is connected then it will not bipartite.

Complete Bipartite Graph: If every vertex in X is disjoint is every vertex in Y, then it is called a complete bipartite graph. If X and Y condition m and n vertices then this graph is denoted by $K_{m,n}$.

Ex:



$X = \{v_1, v_4\}$

$Y = \{v_2, v_3\}$

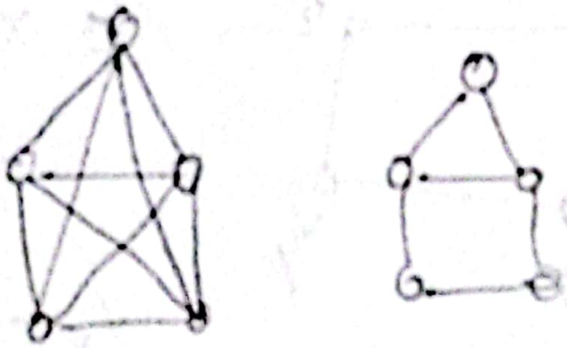
$X \rightarrow$ elements 2

$Y \rightarrow$ elements 2

$\therefore$ It is denoted by $K_{2,2}$

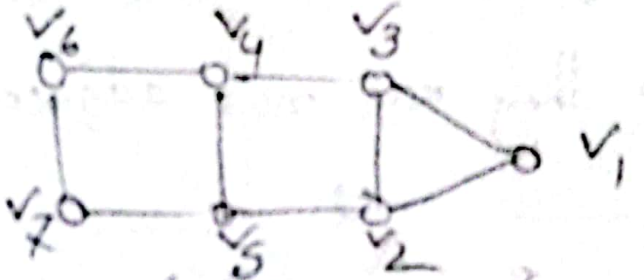
□ **Subgraph:** Let $G(V,E)$ be a graph. Let V' be a subset of V and E' be a subset of E whose end point belong to V'. Then $G(V',E')$ is a graph and called a subgraph of $G(V,E)$.

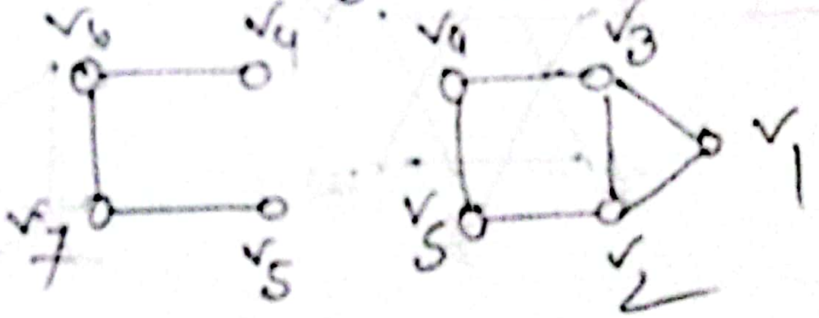
Ex:



Decomposition of Graph: A graph is said to be decomposed into two subgraph G_1 and G_2 if $G_1 \cup G_2 = G$ and $G_1 \cap G_2 =$ null graph

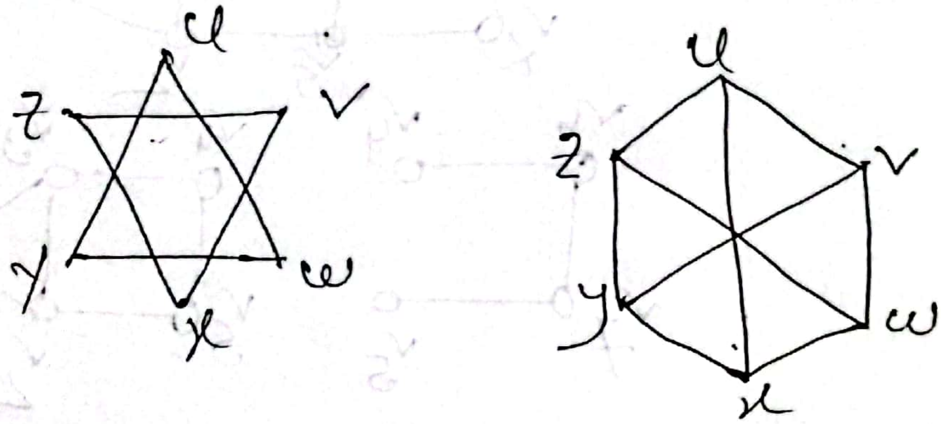
Ex:





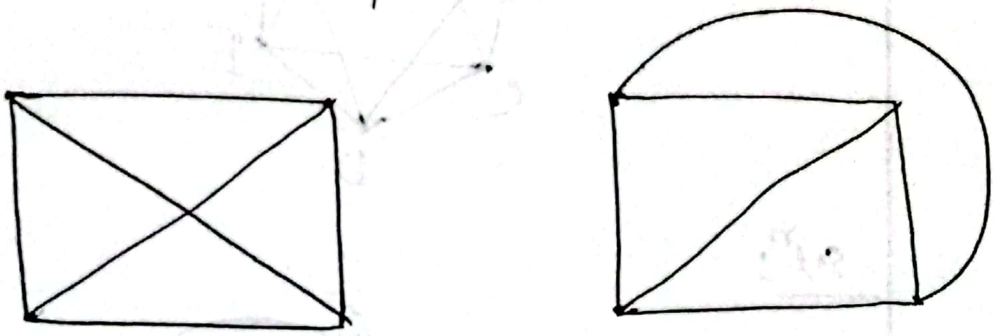
Complement of Graph: The complement of a graph G is defined as a simple graph with same vertex set as G and where two vertices u and v are adjacent only when they are not adjacent in G.

Ex:

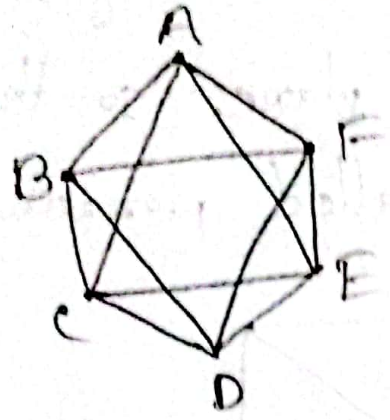


~~REA~~ **Planar Graph:** A graph which can be drawn in the plane so that its edges do not cross is called planar.

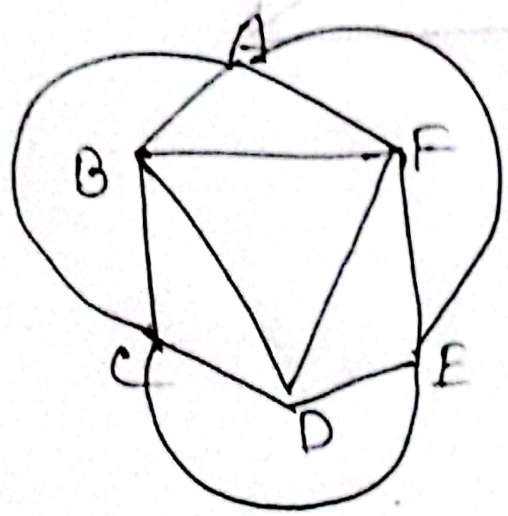
Ex:



Q) Draw a planar representation of a graph.

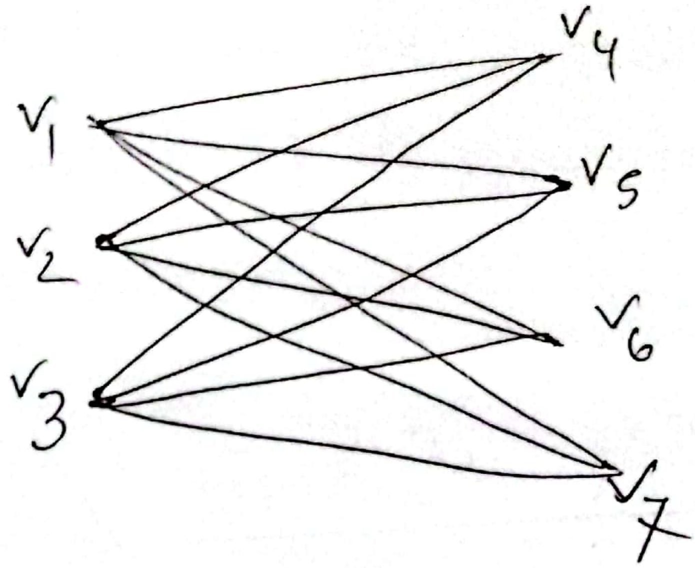


Sol^n:



Q] Draw $K_{3,4}$ bipratite graph.

Sol^n)

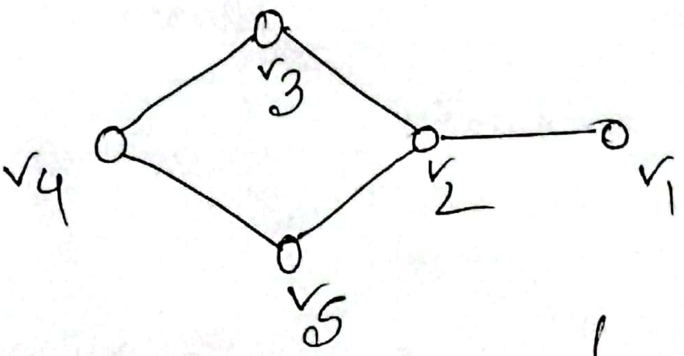


Hand shaking Theorem

* The sum of the degree of the vertices of a graph G is equal to twice the number of edges in G.

$$\boxed{\sum_{i=1}^{n} deg(v_i) = 2 \times \text{Number of edges}}$$

Ex :



$deg(v_1) = 1$
$deg(v_2) = 3$
$deg(v_3) = 2$
$deg(v_4) = 2$
$deg(v_5) = 2$

Sum of degree of all vertices = 1+3+2+2+2 = 10

$\therefore$ Number of edges are = 5

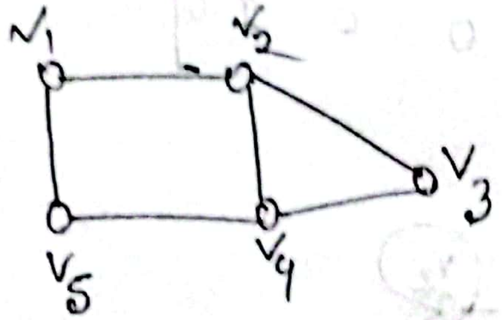
$\therefore$ $\boxed{V = 2 \times E}$

Matrix presentation of graph

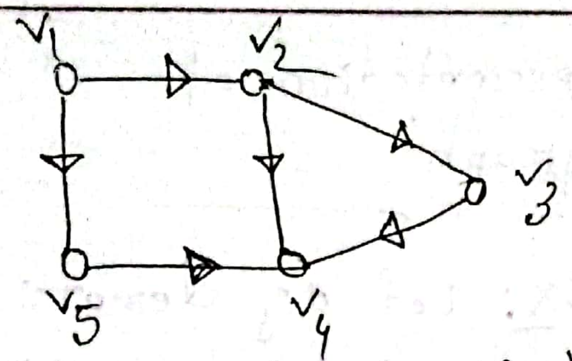
① <u>Adjacency Matrix</u>: Let a_{ij} denote the number of edges (v_i, v_j) then $A = [a_{ij}]_{m \times n}$ is called adjacency matrix of G if

$$a_{ij} = \begin{cases} 1, & \text{if } (v_i, v_j) \text{ is an edges} \\ 0 & \text{otherwise} \end{cases}$$

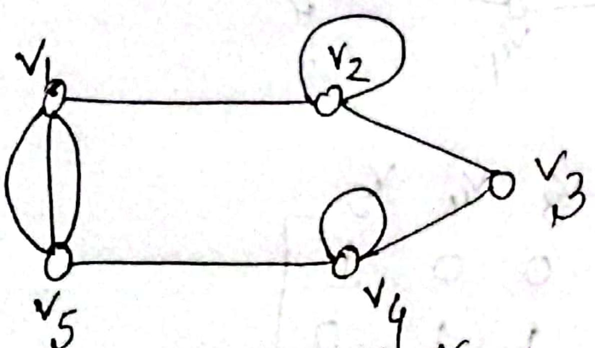
Example:



$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$



$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$



[self loop → 1]

$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 3 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 3 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

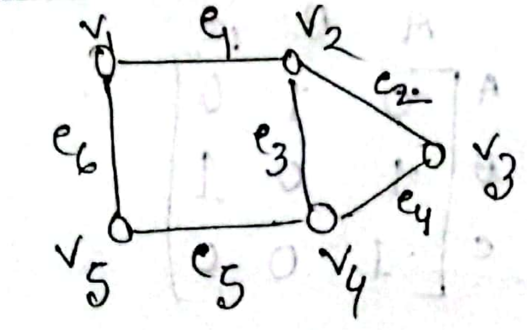
Incidence Matrix: Let G be a graph with m vertices $v_1, v_2, \ldots, v_m$ and n edges $e_1, e_2, \ldots e_n$.

Let a matrix $M = [m_{ij}]_{m \times n}$ defined by

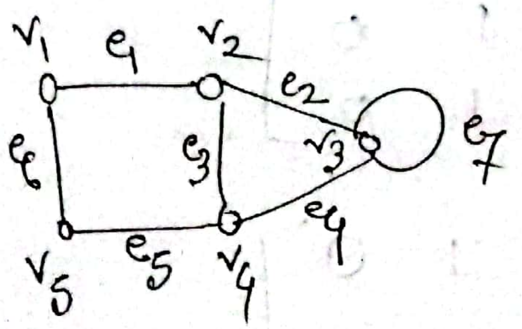
$$m_{ij} = \begin{cases} 1 & \text{if the vertex } v_i \text{ is incident on the } e_j \\ 0 & v_i \text{ is not incident on } e_j \\ 2 & v_i \text{ is an end of the loop } e_j. \end{cases}$$

$\boxed{\text{self loop} \to 2}$

Ex:

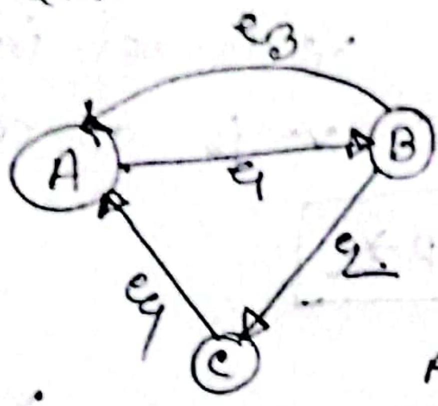


$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array}$$



$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Path matrix: Suppose G is a simple directed graph with m vertices; then path matrix P and B_m have same non-zero entries where $B_m = A + A^2 + \cdots + A^m$, where A is adjacency matrix.



Ex:

Vertices $\boxed{m = 3}$

Adjacency matrix $A = \begin{array}{c c} & \begin{array}{ccc} A & B & C \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \end{array}$

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

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

Now,

$$B_3 = A + A^2 + A^3$$

$$B_3 = \begin{bmatrix} 2 & 2 & 1 \\ 3 & 2 & 2 \\ 2 & 1 & 1 \end{bmatrix}$$

(P) $B \to$ non zero $\to 1$

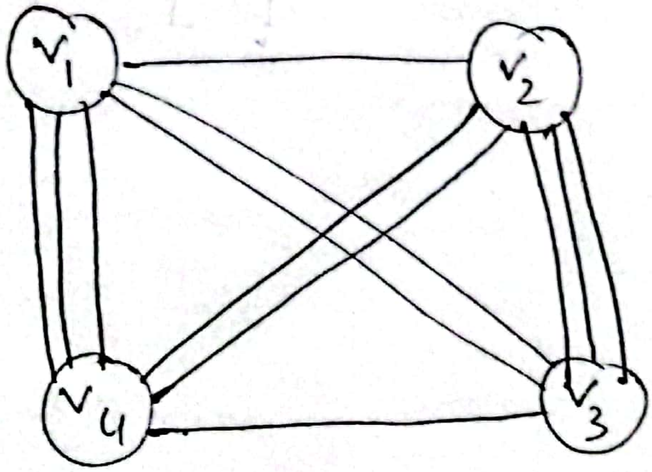
$B \to$ zero $\to 0$

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

A graph is given by the following adjacency matrix

$$A = \begin{array}{c c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix} \end{array}$$

Solⁿ:



Isomorphism of Graph:

Two graphs G_1 and G_2 are isomorphic if

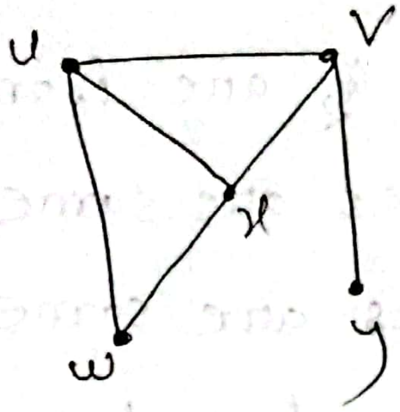
(i) Number of vertices are same.

(ii) Number of edges are same.

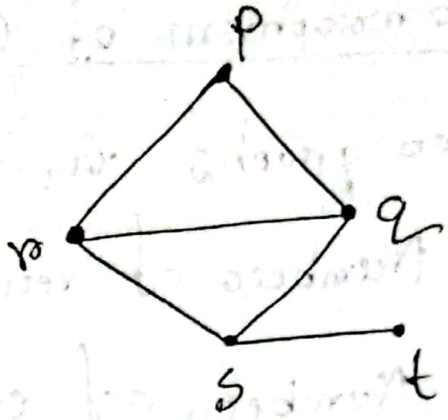
(iii) An equal number of vertices with given degree.

(iv) Vertex correspondence and edge correspondence valid.

Examples



G_1



G_2

(i) Number of vertices are same.

(ii) Numbers of edges are same.

Degree	
$u \to 3$	$p \to 2$
$v \to 3$	$q \to 3$
$w \to 2$	$r \to 3$
$x \to 3$	$s \to 3$
$y \to 1$	$t \to 1$

(iii) An equal number of vertices with given degree.

(iv) Degree wise matching : (correspondence)

verties

y — t

w — p

v — s

u — q

x — r

and for edges .

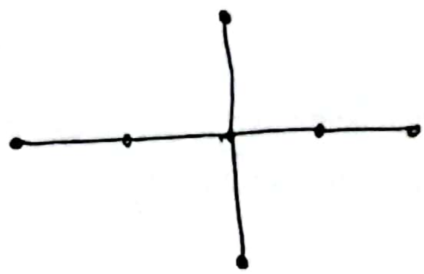
so Both are isomorphic .

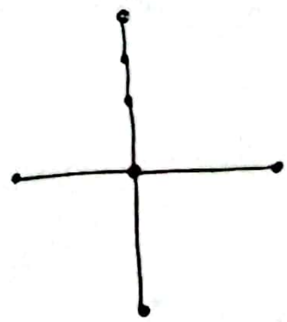
Homeomorphic Graph:

Two graph G and G′ are said to be homeomorphic if they can be obtained from the same graph.

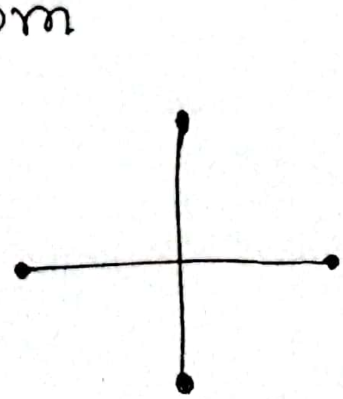
Note: If G and G′ are homeomorphic then they need not be isomorphic.

Example:

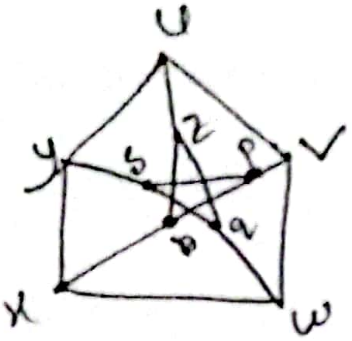


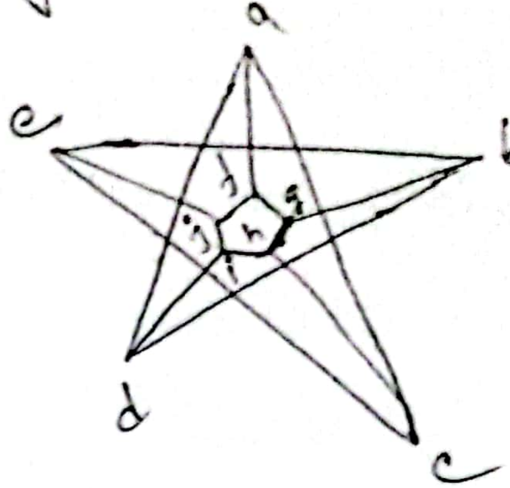


Both are homeomorphic because both are obtain from



□ Show the ~~the~~ following graphs are isomorphic.





soln: ~~all verti~~

* Both graph ~~sam~~ has same vertices and edges.

* all vertices have same degree.

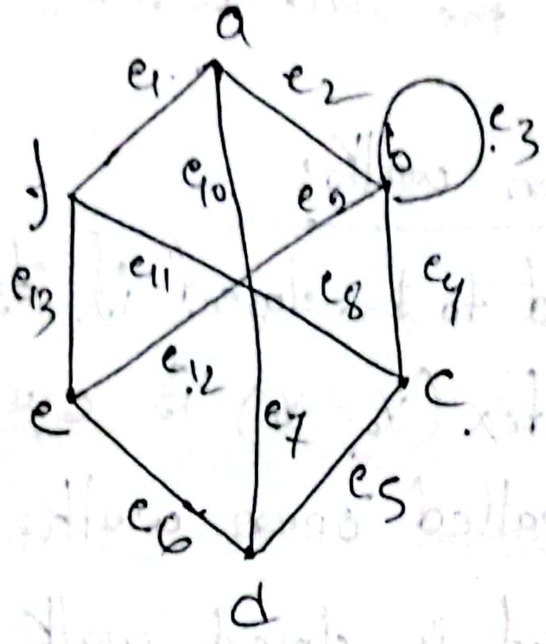
correspondence for both vertices and edges

y — e
u — b
v — d
w — a
x — c

r — h
p — i
s — j
q — f
z — g

Walk: A walk is a finite alternating sequence $v_1 e_1 v_2 e_2 v_3 e_3 \ldots e_n v_n$ of vertices and edges, beginning and ending with same or different vertices.

Ex:



then,

(i) $a e_1 f e_{11} g e_9 b e_3 b e_4 c$

(ii) $a e_2 b e_9 g e_8 c e_4 b e_9 g e_{12} e$ is walk

Length of walk: The number of edges is called length of the walk.

Ex: If walk → $a e_2 b e_4 c e_8 g e_{11} f$

So, Length of the walk will be 4.

Closed and Open walk:

A walk is said to be closed if its origin and terminous vertex ($v_0 = v_n$) is equal otherwise it is called open walk.

Ex: $f e_{11} g e_{10} a e_1 f$ is closed walk.

Trail: Any walk having different edges is called trail.

circuit: A closed trail is called circuit

Ex: $a e_1 f e_{11} g e_9 b e_2 a$

Path: A walk is called path if all vertices are not repeated.

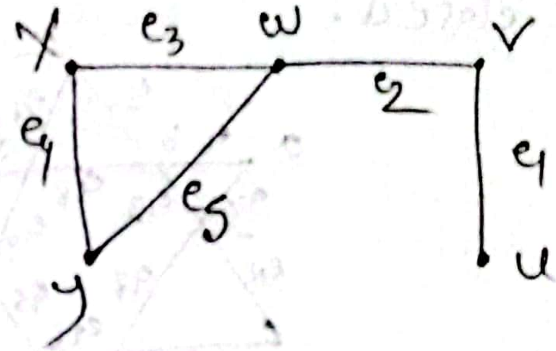
Cycle: A closed path is called cycle.

Ex: $a e_{10} g e_7 d e_6 e e_{13} f e_1 a$

Eulerian Path

A path in a graph is said to be an Eulerian path if it traverses each edge in the graph once and only once.

Ex:



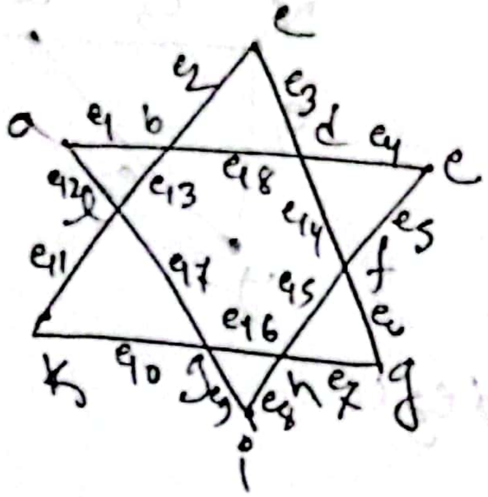
then,

$u e_1 v e_2 w e_3 x e_4 y e_5 w$.

Eulerian Circuit

A circuit in a graph is said to be an Eulerian circuit if it traverses each edge in the graph once and only once.

* closed.



Ans

$l\, e_{11}\, k\, e_{10}\, j\, e_9\, i\, e_8\, h\, e_7\, g\, e_6\, f\, e_5\, e\, e_4\, d\, e_3\, c\, e_2\, b\, e_1\, a$

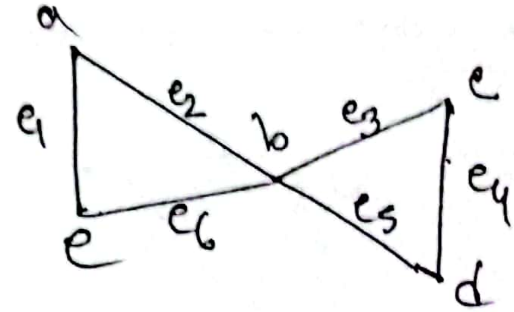
$e_{12}\, l\, e_{17}\, j\, e_{16}\, h\, e_{15}\, f\, e_{14}\, d\, e_{18}\, b\, e_{13}\, l$

* All edges have to cover and no repeat

Eulerian Graph

A connected graph which contain an Eulerian circuit is called Eulerian graph.

Ex)

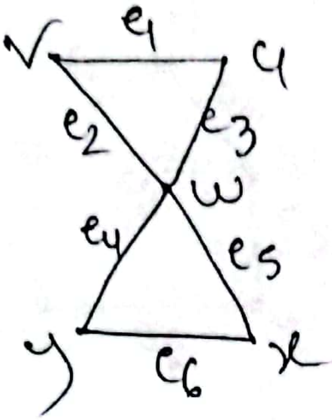


Hamiltonian Path

A path which contain every vertex of a graph G exactly once is called Hamiltonian Graph.

Ex ④ All vertex covers and no repeat

Ex:

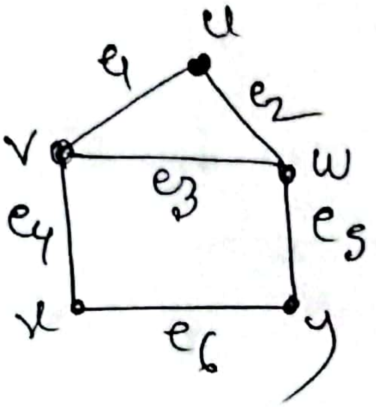


then, $u e_1 v e_2 w e_5 x e_6 y$

Hamiltonian Circuit

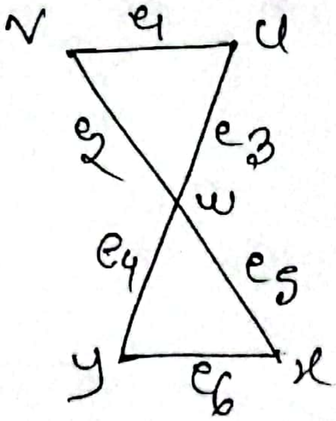
A circuit that passes through each of the vertices in a group G exactly one except the starting vertex and end vertex is called hamiltonian circuit.

Ex:



* All verties cover and no repeat ↓ (without start and end)
* closed.

then, $u e_1 v e_4 x e_6 y e_5 w e_2 u$



then, $u e_1 v e_2 w e_5 x e_6 y e_4 w e_4 w$

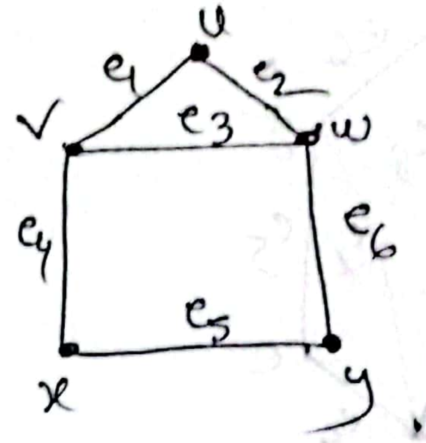
is not Hamiltonian circuit.

because w vertex repeat.

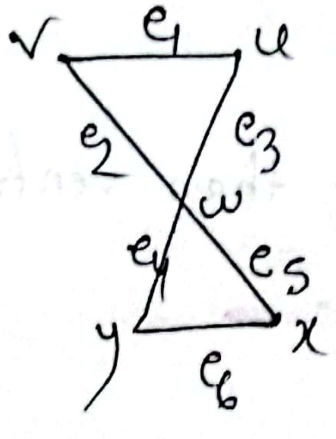
Hamiltonian Graph

Def: A connected graph which contain Hamiltonian circuit is called Hamiltonian Graph.

Ex:



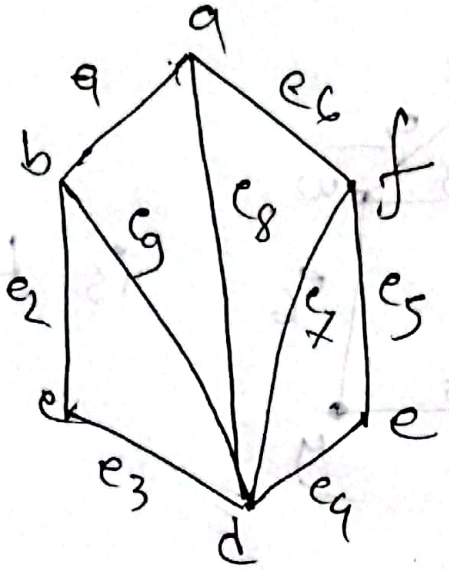
is Hamiltonian Graph.



is not Hamiltonian Graph

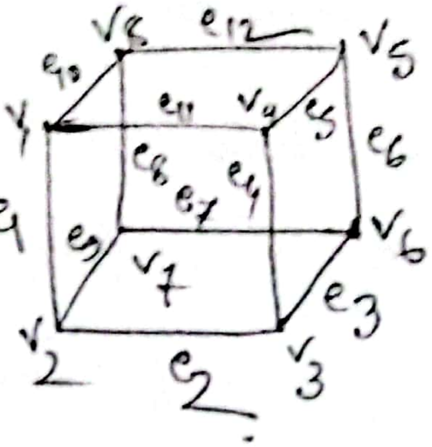
* Draw a graph with six vertices conta a Hamiltonian circuit but not Eulerian circuit.

Ex:



(x) if edges is more than vertices is n Eulerian.

[7]



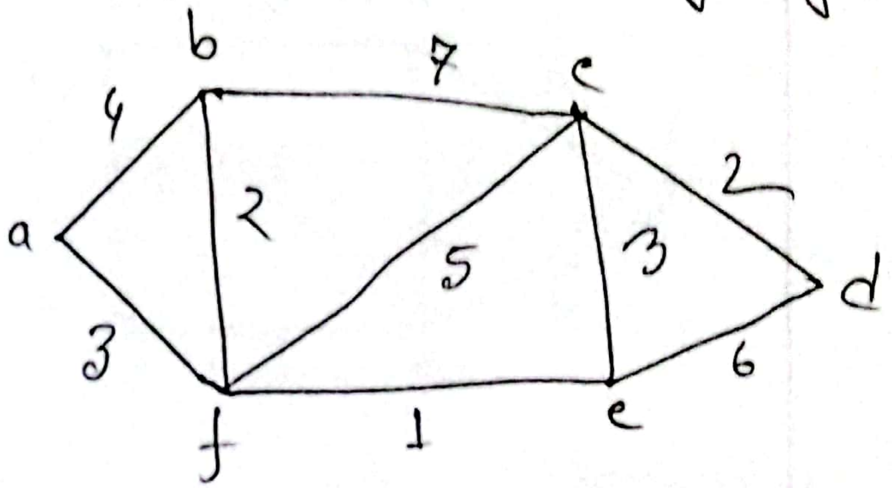
Hamiltonian graph.

$v_2\, e_2\, v_3\, e_3\, v_6\, e_7\, v_7\, e_8\, v_8\, e_{12}\, v_5\, e_5\, v_4\, e_{11}\, v_1\, e_1\, v_2$

Weighted Graph

A graph is called weighted graph if a non-negative integers $w(e)$ associate to each edge and this $w(e)$ is a weight of corresponding edge.

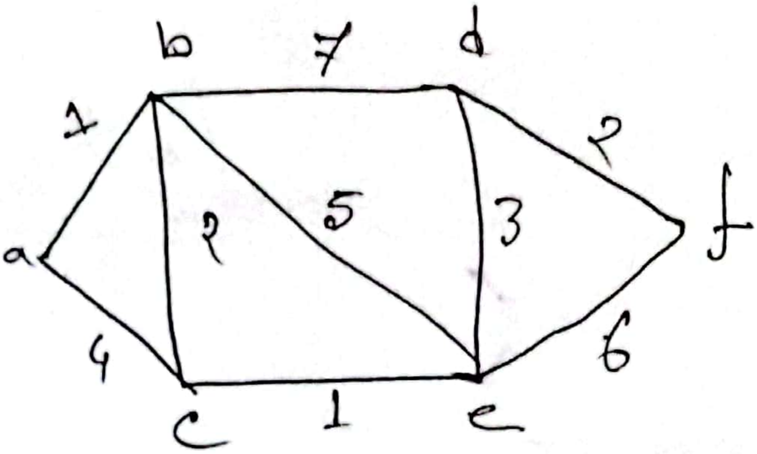
Ex:



Shortest Path in weighted Graph

DIJKSTRA's Algorithm:

Example:



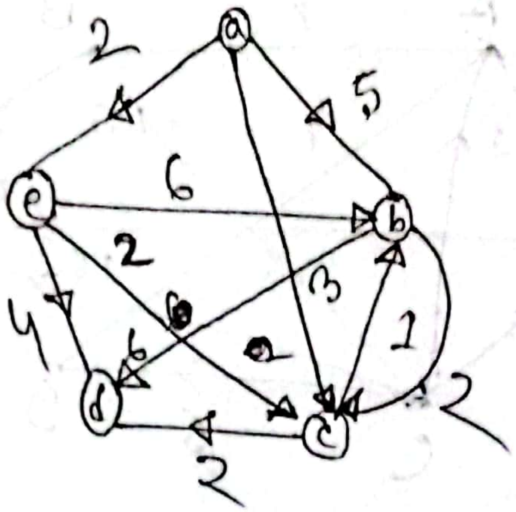
a	b	c	d	e	f
$\boxed{0}$	∞	∞	∞	∞	∞
$\boxed{0}$	$\boxed{1}$	4	∞	∞	∞
$\boxed{0}$	$\boxed{1}$	3	8	6	∞
$\boxed{0}$	$\boxed{1}$	$\boxed{3}$	8	$\boxed{4}$	∞
$\boxed{0}$	$\boxed{1}$	$\boxed{3}$	$\boxed{7}$	$\boxed{4}$	10
$\boxed{0}$	$\boxed{1}$	$\boxed{3}$	$\boxed{7}$	$\boxed{4}$	$\boxed{9}$

shortest path

Shortest path: abcedf

* যার সাথে connection নাই তার ∞ ।
* যেইটা সবচেয়ে ছোট তার বক্স।
* New direction এ গেলে আগের গুলো যোগ করতে হবে।
* কোনো connection না থাকলে previous টা হবে
* ~~[illegible]~~ previous and new এর মধ্যে যেটা ছোট তথা হবে।

☐ compute the shortest path for the following graph between a and d.



a	b	c	d	e
$\boxed{0}$	∞	∞	∞	∞
$\boxed{0}$	5	3	∞	$\boxed{2}$
$\boxed{0}$	5	$\boxed{3}$	6	$\boxed{2}$
$\boxed{0}$	$\boxed{4}$	$\boxed{3}$	5	$\boxed{2}$
$\boxed{0}$	$\boxed{4}$	$\boxed{3}$	$\boxed{5}$	$\boxed{2}$

$2+6 = 8 > 5$

$2+2 = 4 > 3$

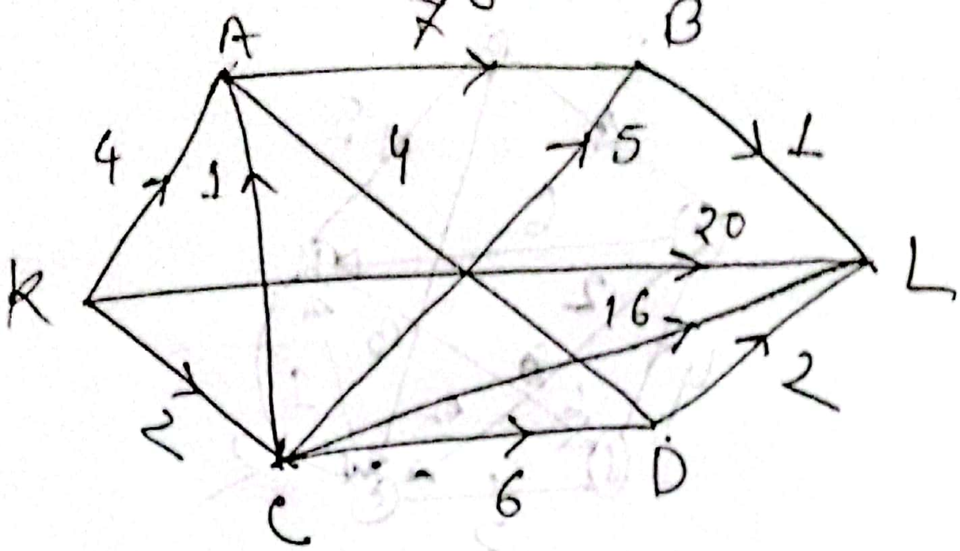
$3+1 = 4 < 5$

$4+6 = 10 > 5$

distance = 5

- shortest path = a c d

(b) Find ~~the~~ minimum distance between two vertices K and L of graph.



Sol^n:

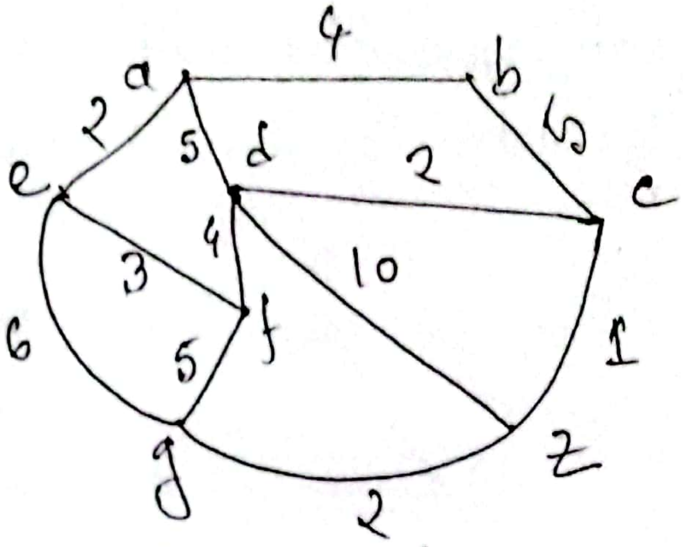
K	A	B	C	D	L
$\boxed{0}$	∞	∞	∞	∞	∞
$\boxed{0}$	4	∞	$\boxed{2}$	∞	20
$\boxed{0}$	$\boxed{3}$	7	$\boxed{2}$	8	18
$\boxed{0}$	$\boxed{3}$	$\boxed{7}$	$\boxed{2}$	7	18
$\boxed{0}$	$\boxed{3}$	$\boxed{7}$	$\boxed{2}$	$\boxed{7}$	8

min দুইটা sam
হলে যেকোনো
একটি নিলেই হ

Minimum distance = 8

Minimum path = KCBL

b) Find the length of shortest path from vertex a to z is



a	b	c	d	e	f	g	z
$\boxed{0}$	∞	∞	∞	∞	∞	∞	∞
$\boxed{0}$	4	∞	5	$\boxed{2}$	∞	∞	∞
$\boxed{0}$	$\boxed{4}$	∞	5	$\boxed{2}$	5	8	∞
$\boxed{0}$	$\boxed{4}$	9	$\boxed{5}$	$\boxed{2}$	5	8	∞
$\boxed{0}$	$\boxed{4}$	7	$\boxed{5}$	$\boxed{2}$	$\boxed{5}$	8	15
$\boxed{0}$	$\boxed{4}$	$\boxed{7}$	$\boxed{5}$	$\boxed{2}$	$\boxed{5}$	8	15
$\boxed{0}$	$\boxed{4}$	$\boxed{7}$	$\boxed{5}$	$\boxed{2}$	$\boxed{5}$	$\boxed{8}$	8
$\boxed{0}$	$\boxed{4}$	$\boxed{7}$	$\boxed{5}$	$\boxed{2}$	$\boxed{5}$	$\boxed{8}$	8

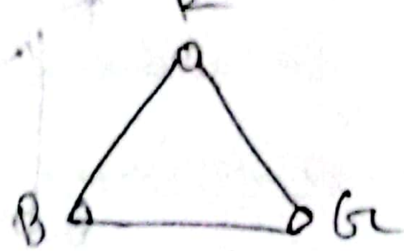
shortest distance = 8

shortest path = adcz

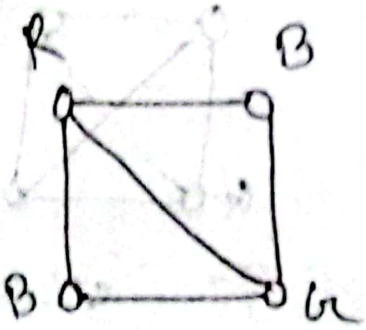
Graph Coloring

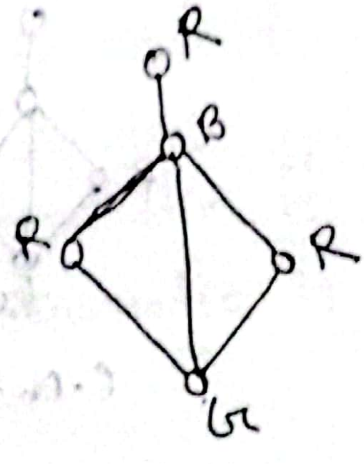
Painting all the vertices of a graph with colors such that no two adjacent vertices have the same color is called coloring of graph.

Ex:



R- red
B → black
G → Green
P → purple

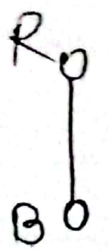




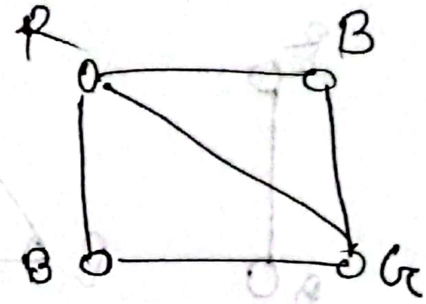
Chromatic Number

The least number of colors required for coloring of a graph G is called its chromatic number.

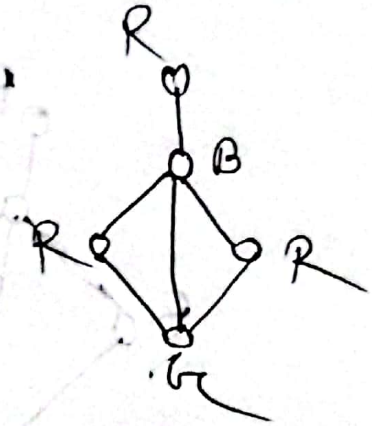
Ex:



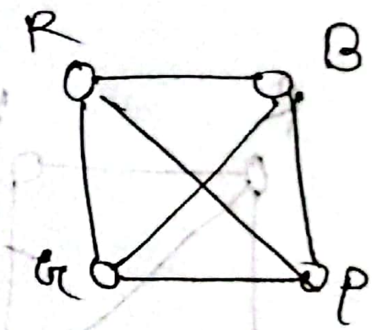
C.N = 2



C.N. = 3



C.N. = 3



C.N. = 4

Note :

1. The chromatic number of graph G is denoted by $\chi(G)$

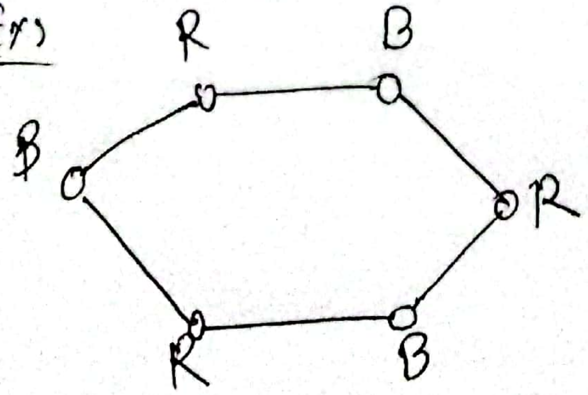
2. If $\chi(G) = k$, then the graph is called k-chromatic.

3. Chromatic number of null graph is 1.

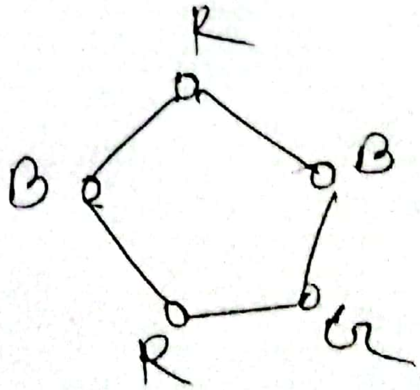
4. Chromatic number of complete graph K_n of n vertices is n.

5. If a graph is circuit with n vertices then
   (i) It is 2-chromatic if n is even.
   (ii) It is 3-chromatic if n is odd.

Ex)



∴ 2-chromatic



3-chromatic

Welsh Powell Algorithm

→ This algorithm gives the ~~minimum~~ number of minimum colours we need.

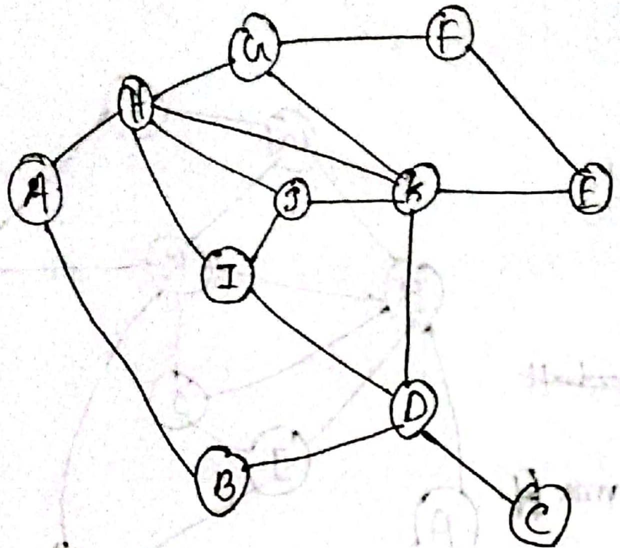
→ This algorithm is also used to find the chromatic number of a graph. This is an iterative greedy approach.

welsh powell algorithm consists of following steps.:

1. Find the degree of ~~vertex~~ each vertex.
2. List the vertices in order of descending degrees.
3. Colour the first vertex with color 1.
4. Move down the list and color all the vertices not connected to the coloured vertex, with same color.
5. Repeat step 4 on all uncolored vertices with a new color, in descending order of degrees until all the vertices are coloured.

By starting with the highest degree, we make sure that the vertex with the highest number of conflicts can be taken care of as early as possible.

Example



$d(A) = 2$
$d(B) = 2$
$d(C) = 1$
$d(D) = 4$
$d(E) = 2$
$d(F) = 2$
$d(G) = 3$
$d(H) = 5$
$d(I) = 3$
$d(J) = 3$
$d(K) = 5$

- First, order the list in descending order of degrees.
- In case of tie, we can randomly choose any ways to break it.

So, the new order will be: H, K, D, G, I, J, A, B, E, F, C

So, order,

H, K, D, G, I, J, A, B, E, F, C

H → color red

K → ~~k~~ can't red, as it connects to H

D → color red, ~~connect H~~

G → can't red connec H

I → can't red, connect H

J → can't red, connects H

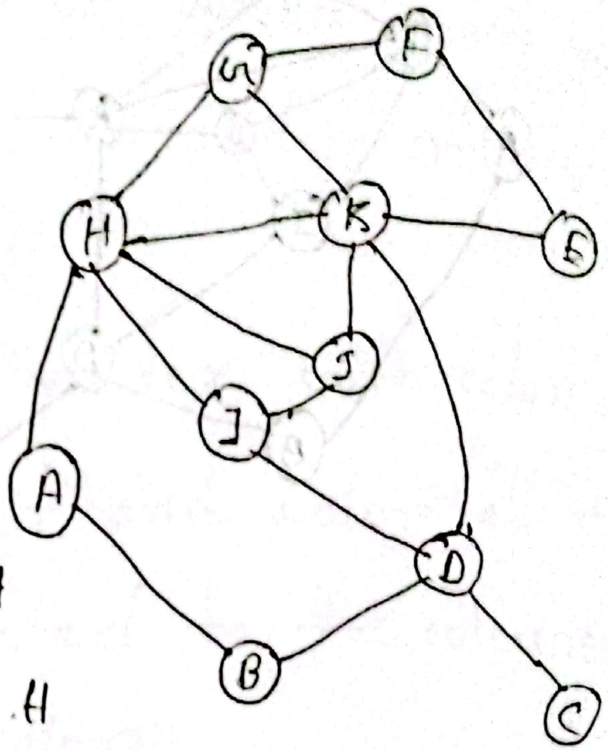
A → can't red, connects H

B → can't reed, connects to D

E → color red

F → can't red, connects to E

C → can't red, connects D



Then, Ignore the ventices whic alread coloured,

∴ K, G, I, J, A, B, F, C

K → color Green

G → can't Green, connects to K

I → color Green

J → can't Green, connects to K

A → color green

B → can't green, connects to A

F → color green

C → color green

Then left ventices are → G, J, B

G → color Blue

J → color Blue

B → color Blue

so, we need minimum 3 types of color

so; the chromatic number of this graph is 3.