5. Graph theory :-

Graph :- A graph is a collection of vertices connected to each other through a set of edges

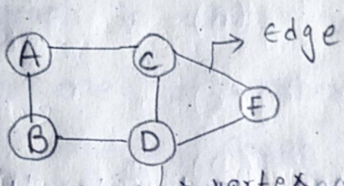
→ A study of graphs is known as "Graph Theory"

Formal definition :-

formally a Graph is defined as Ordered pair of a set of vertices and a set of edges denoted as

$G = (V, E)$ where $V \rightarrow$ set of vertices
$E \rightarrow$ set of Edges

Ex :-



Here the graph contains '5' vertices and '6' edges denoted as

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

$E = \{ AB, AC, BD, CD, CE, DE \}$

** NOTE :- The maximum number of Edges in a graph with 'n' vertices is given by

$$\boxed{{}^{n}C_{2} = \frac{n(n-1)}{2}}$$

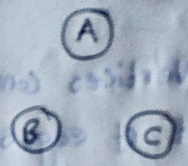
Types of Graphs :-

Various important types of graphs in graph theory are

1. Null graph :-

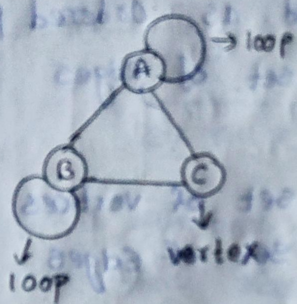
- A graph where edge set is empty is called as a null graph
- A null graph do not have any edge in it.

Ex :-



Here the graph Contains only '3' vertices (A, B, C) but there are no edges (Edge set in empty)

2. Loop :-



A Loop is an edge that Connect a vertex to itself Here there are '3' vertices (A, B, C) vertex A, B, are having an edge Connected to themselves.

3: Trivial Graph :-

A Graph having only one vertex in it is Called as Trivial graph

→ It is a smallest possible graph

Ex :-

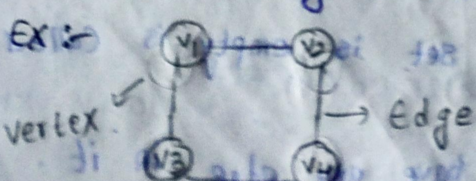
Here the graph has only one vertex (A) and no edges so it is a trivial graph.

4. un-directed graph :-

A graph in which all the edges are undirected is Called as un-directed graph

→ Edges of an un-directed graph does not Contain any direction.

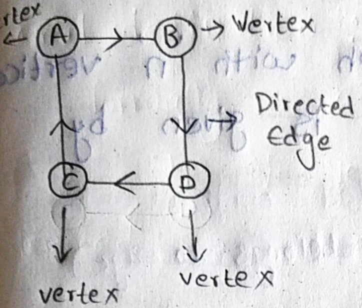
Ex :-



5. Directed graph :-

A graph in which all the edges are assigned some direction to perticular vertex such graph is called as a directed graph.

Ex :-



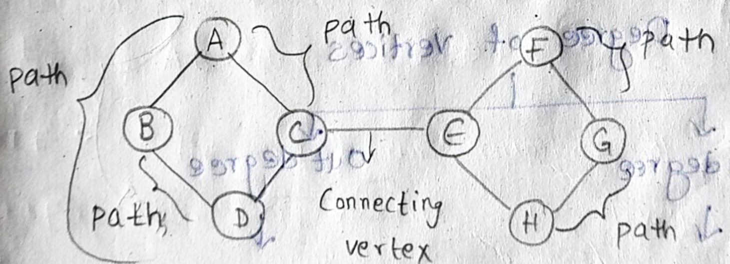
Note :- Directed graphs are also known as "Diagraph"

6. Connected Graph :-

A graph in which we vist any of the vertex from one to another vertex such graph is called as Connected graph

Note :- Here at least one path exists between every pair of vertices

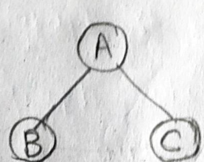
Ex :-



7. DisConnected graph :-

A graph in which there does not exist any path between atleast one pair of vertices then that graph is called as DisConnected graph

Ex :-



Here one pair of vertices (B, C) are not connected Hence it is a disconnected graph

8. Degree of vertex in a graph :- pair of angles

Imp

The number of vertices <u>adjacent</u> to the vertex 'v' in a graph is called as degree of the vertex

→ It is denoted as deg(v)

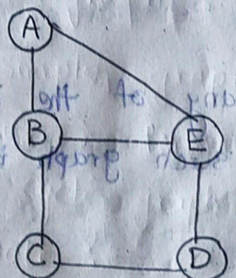
Self loop Counted twice

Note :- In a simple graph with 'n' vertices the degree of any vertex is given by

$$\boxed{deg(v) = n-1 \quad \forall\, v \in G}$$

$\in$ → belongs to

Ex :-



Here degree (A) = 2 [As there are '2' edges meeting at vertex A]

Similarly deg (B) = 3

deg (C) = 2

deg (D) = 2

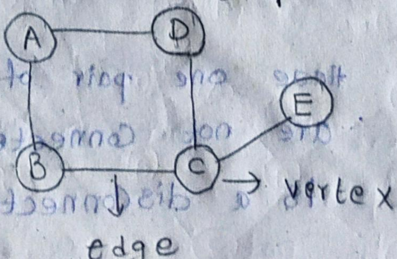
deg (E) = 3

Types of Degree of vertex :-

Degree of vertices

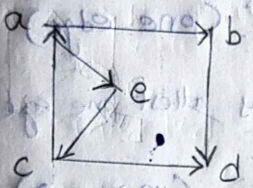
- In degree → Number of vertices comming into vertex 'v' is called as In-degree denoted as $deg^{-}(v)$
- Out degree → Number of vertices going out from vertex 'v' is called as Out-degree denoted as $deg^{+}(v)$

Ex :-1 undirected graph



vertex	Indegree	out degree
A	2	2
B	2	2
C	3	3
D	2	2
E	1	1

Ex :- 2 Directed graph



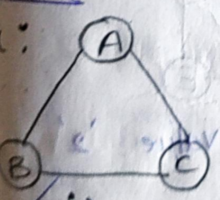
vertex	Indegree	outdegree
a	1	2
b	1	1
c	~~0~~ 1	2
d	2	0
e	~~2~~ 1	1

9. **Regular graph :-**

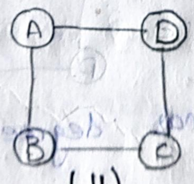
A graph in which degree of all the vertices is same is called as Regular graph.

→ If all the vertices of a graph are of degree 'k' then that graph is called as "k-degree Regular graph".

Ex :



(i)



(II)

Here in both the graphs every vertex has the same degree value. So they are Regular of degree - 2

deg (A) = 2
deg (B) = 2
deg (C) = 2

deg(A) = 2
deg(B) = 2
deg(C) = 2
deg(D) = 2

10) **Complete graph :-**

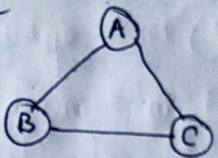
A graph in which exactly one edge is present between every pair of vertices such graph is called as Complete graph.

Note :-

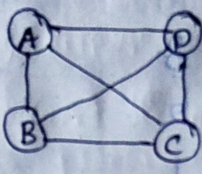
A Complete graph of 'n' vertices Contains $\boxed{{}^{n}C_2 \text{ edges}}$

A Complete graph with 'n' vertices is represented as K_n

Ex :-



(i) K_3



K_4

Here every vertex is connected with all the remaining vertices 'exactly once' (one edge)

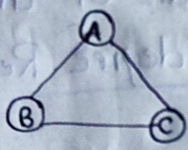
11. Cycle graph :- ~~A graph contains atleast one cycle~~

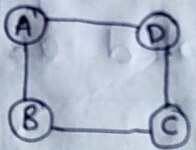
A Simple graph of 'n' vertices ($n >= 3$) and 'n' edges forming a Cycle of length 'n' is Called as Cycle graph (closed walk)

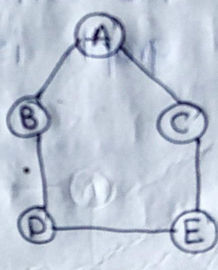
** Note :-

In a Cycle graph all the vertices has degree = 2

Ex :-





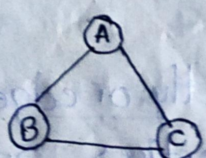


Here all the 3 graphs has degree value '2' so they are cycle graphs

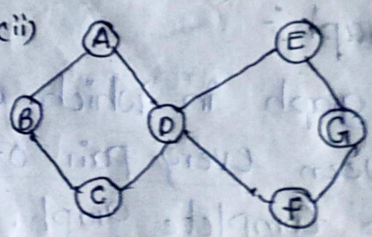
12. Cyclic graph :-

A graph Containing at least one cycle is Called as Cyclic graph [all vertices must have degree value = 2]

(i)



(ii)



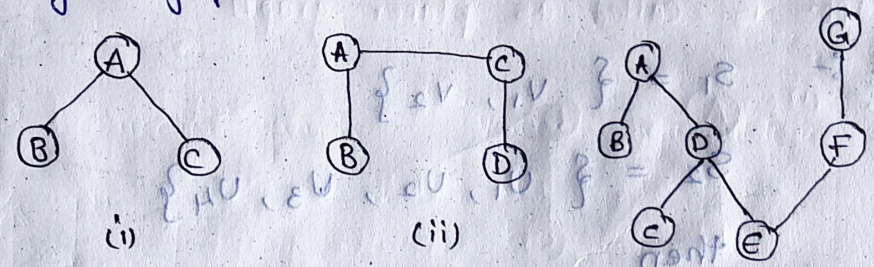
Here both graphs there exist atleast one cycle (i.e, vertices having degree value as '2')

So they are Cyclic graphs

13. Acyclic graph :-

A graph that do not Contain any Cycle is Called as Acyclic graph

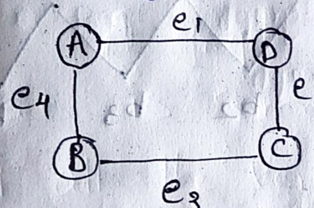
Ex :-



(i) (ii)

14. Finite graph :-

A graph Consisting of finite number of Vertices and edges that graph is Called as finite graph



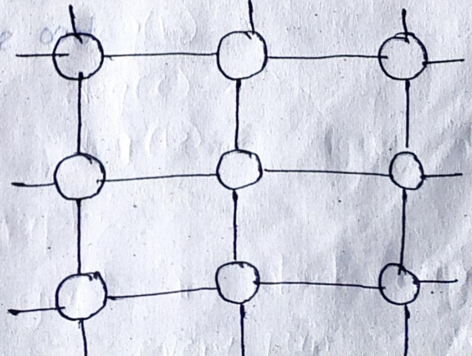
Here vertex set $V = \{A, B, C, D\}$

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

15. Infinite graph :-

A graph Containing infinite number of vertices and Edges that graph is Called as Infinite graph.

Ex :-



16 **Biparatite graph :-** (two parts)

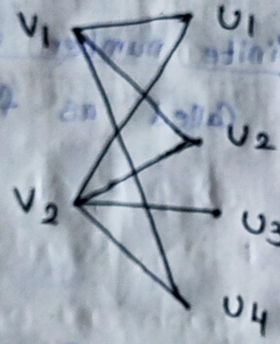
A graph whose vertices can be divided into two Independent set U, V such that every edge m (U, V) either connects a vertex from U to V (or) from V to U

Ex :- $S_1 = \{v_1, v_2\}$

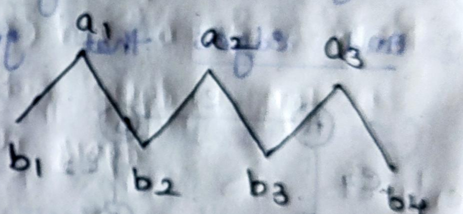
$S_2 = \{U_1, U_2, U_3, U_4\}$

then

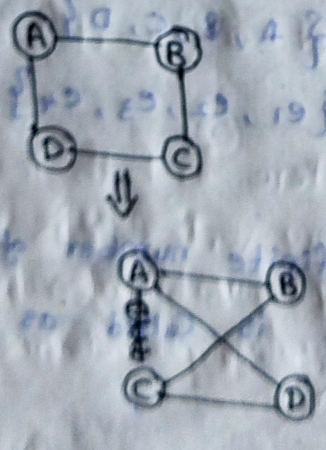
(i)



(ii)



(iii)



$X = \{A, C\}$

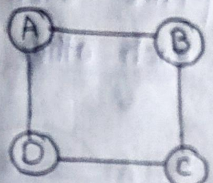
$Y = \{B, D\}$

vertices divided into two sets X & Y

Complete Biparatite graph :-

A biparatite graph in which every vertex of one set is joined with every vertex of another set such graph is called as Complete biparatite graph

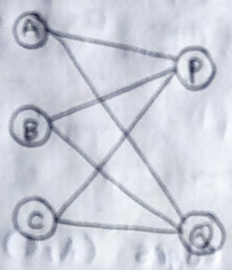
Ex :- 1



$X = \{A, C\}$

$Y = \{B, D\}$

Ex :- 2

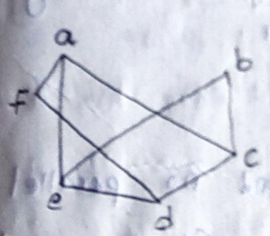


Note :- $\boxed{\text{maximum number of edges for a biparatite graph of 'n' vertices} = \frac{1}{4} * n^2 \text{ edges}}$

where 'n' is the no of vertices

Verify whether the following graph is a biparatite graph or a complete biparatite graph.

(i)



from the given graph total vertex V_1 & V_2

$V_1 = \{a, b, d\}$

$V_2 = \{c, e, f\}$

from the given graph if the two set of graph vertices is assumed V_1 & V_2 then

$V_1 = \{a, b, d\}$

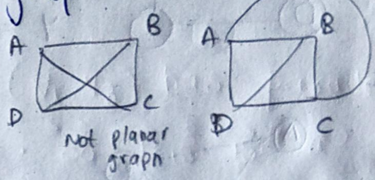
$V_2 = \{c, e, f\}$

Here we observe that each vertex in V_1 has an edge (path) with the vertices available in set V_2

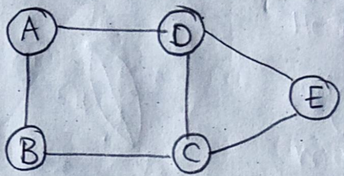
∴ we can say that the given graph is Biparatite graph.

18. planar graph :-

A graph that can be drawn in a plane such that no two edges of the graph cross each other such graph is called as planer graph

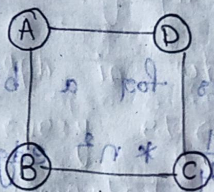


Ex :- (i)

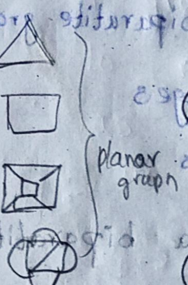


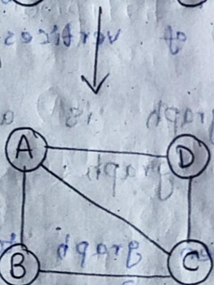
It is a planer graph

(ii)



Here the edges (A, C) and (B, D) has a cross over so this graph is not a planar graph



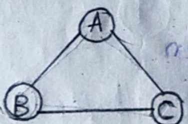


This graph is a planer graph since no edges are having a cross over.

19. Simple graph :-

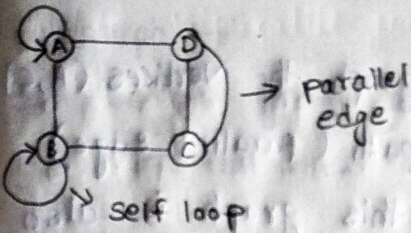
A graph having no self loop and no parallel edges. such a graph is called as simple graph

Ex :- 1



It is a simple graph since it do not have
(i) self loop
(ii) parallel edges

(ii)

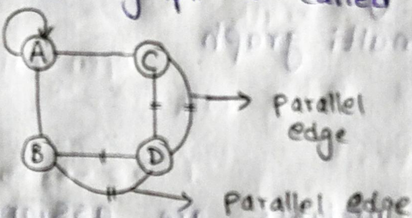


Since the above graph has both self loop and parallel edges this graph is not a simple graph

** 20. pseudo graph :-

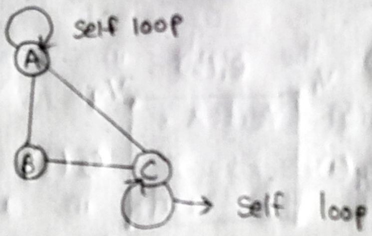
A graph having no parallel edges but has a self loop then such graph is called as pseudo graph.

Ex :-



Since this graph has '2' vertices having parallel edges so it is not a pseudo graph.

(ii)

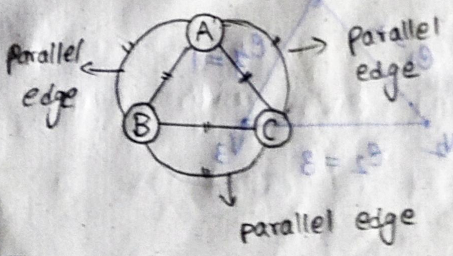


This graph has only self loop's so it is a psudo graph.

** 21 multi graph :-

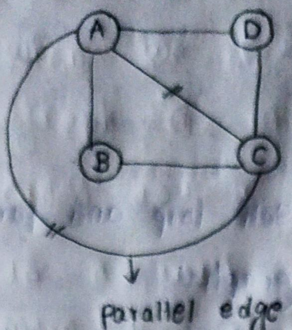
A graph having no self loop but has parallel edges then such a group is called as multi graph

Ex :-



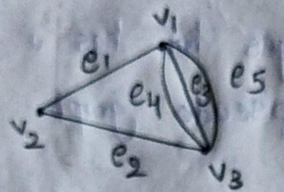
In the above graph each vertex is having a mapping with other vertex in '2' ways. so it is a multi graph

2)



In this graph the vertices (A, C) has two path (parallel edge) So that this graph is also a multi graph.

3)

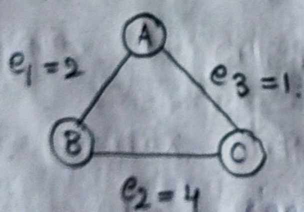


Here (v_1, v_3) has parallel edges. So this is a multi graph

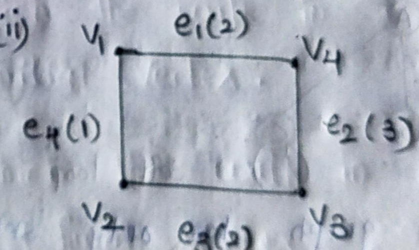
22. weighted graph :-

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

eg :- (i)

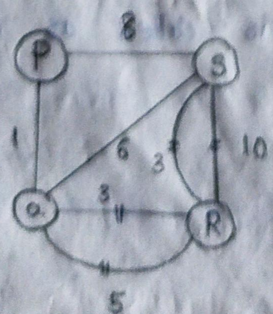


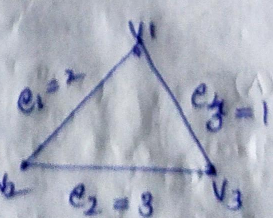
(ii)



Here every edge has a weight assigned so this graph is a weighted graph

(iii)





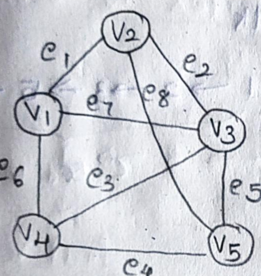
28' Sub graph :-

If we are given 2 graphs G, G_1, we say that G_1 is a subgroup of G' if the following conditions holds :

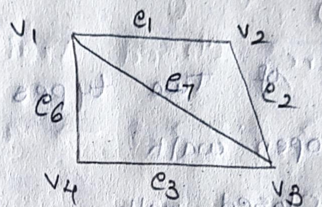
(i) All the vertices and edges of G_1 are in G
i.e $V_1 \in V$ and $E_1 \in E$
↳ belongs to ↳ belong to

(ii) Every Edge in graph G_1 has the same end vertices in graph G as in G_1

Ex :-



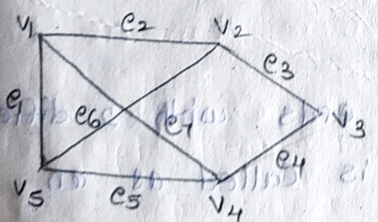
Graph (V, E)



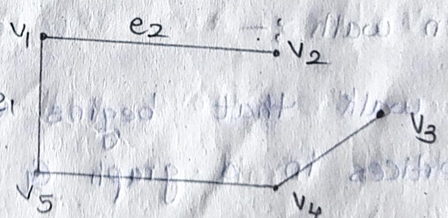
Graph $G_1 (V_1, E_1)$

Here the graph $G_1 (V_1, E_1)$ is a subgraph of graph $G (V, E)$

Ex 2 :-



Graph $G(V, E)$



Graph $G_1 (V_1, E_1)$

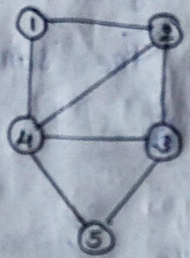
Here G_1 is a subgraph of G

24. Walk :-

when we Cover the vertices and edges of a graph 'G' in a sequence then that path is Called as a walk.

In a walk both vertices and edges appears more than once.

ex :-



Here $\underbrace{\underbrace{1 \to 2 \to 3 \to 4}_{\text{walk}} \to 2 \to 3}_{\text{open walk}} \to 5$ is a walk

$\underbrace{1 \to 2 \to 3 \to 5 \to 4 \to 1}_{\text{closed walk}}$

→ walk ~~is~~ are ~~of~~ Two types :-

1. open walk
2. Closed walk

→ The vertex with which the walk in a graph begins is Called Initial vertex.

→ The vertex with which the walk in a graph ends is Called as final vertex

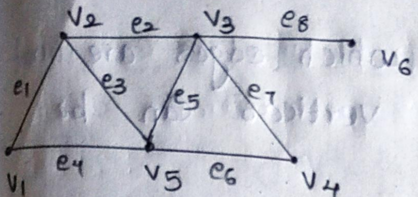
open walk :-

A walk that begins and ends with 2 different vertices in a graph G it is Called as an open walk.

Closed walk :-

A walk that begins and ends with same Vertex in a graph G. It is Called as a Closed walk.

Ex :-



begining vertex, ending vertex

$$\underbrace{\underline{v_1}\ e_1\ v_2\ e_2\ v_3\ e_8\ \underline{v_6}}_{\text{open walk}}$$

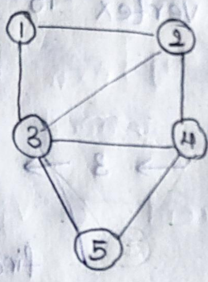
begining vertex, ending vertex

$$\underbrace{\underline{v_1}\ e_1\ v_2\ e_3\ v_5\ e_5\ v_3\ e_2\ v_2\ e_1\ \underline{v_1}}_{\text{Closed walk}}$$

**
25. Trail :-

It is an open walk in which <u>no edges appears more than once</u>

Ex :-



beging vertix (Initial vertex), ending vertex

$\underline{1} \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow \underline{3}$

⇒ Here begining and ending vertices are different, so it is an open walk

⇒ No edges are repeated more than once

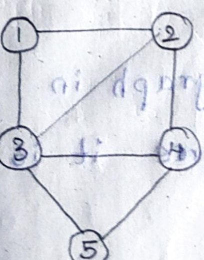
∴ The above path is a walk

**
26. path :-

A Trail in which both vertices and edges are not repeated more than once is called as a path

(Open)

Ex :-

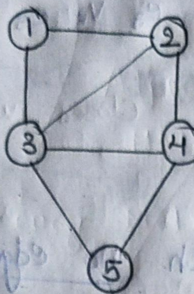


$$\underbrace{1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5}_{\text{Path}}$$

**
27. Closed Trail :-

It is a closed walk in which edges are not repeats more than once but vertices can be repeates more than once

Ex :-



Imp
Note :- The begining and ending vertex is same in a closed Trail

$$\underbrace{\underbrace{\underset{\text{Initial vertex}}{1} \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow \underset{\text{final vertex}}{1}}_{\text{same}}}_{\text{Closed trail}}$$

⇒ No edge is repeated more than once

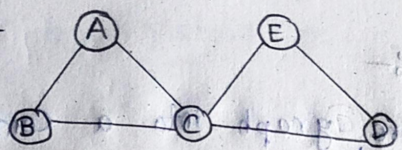
⇒ Vertices are repeated more than one time

Hence the above path is an Example for Closed Trail.

28. Eulers graph :-

Any Connected graph in which all the vertices are of Even degree it is called as Euler's graph.

Ex :-



deg(A) = 2
deg(B) = 2
deg(C) = 4
deg(D) = 2
deg(E) = 2

→ the above graph is a Connected graph

→ Each vertex has degree value = Even

→ so this graph is a Euler's graph

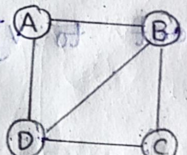
***
29. Euler's Path :-

starting & ending diff for euler path

In a connected graph 'G' if there exists a walk that visits every vertex, Edge exactly once (without repeteation of edge) then that path is called as Euler's path.

Edge ← Each path is covered exactly once

Ex :-

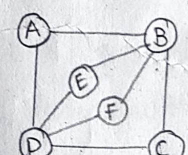


Euler's Path for the graph is

B → C → D → B → A → D

***
Note :- In a path starting and ending vertices are same then that path is called as Euler's Circuit

Ex :-



$\underbrace{B \rightarrow C \rightarrow D \rightarrow F \rightarrow B \rightarrow E \rightarrow D \rightarrow A \rightarrow B}_{\text{Euler Circuit}}$

B ↓ Starting vertex

B ↓ Ending vertex

Starting vertex, Ending vertex : Same

*** Representation of Graphs :-

→ The technique used to store graph into a Computer memory is called as Graph representation

Graph Representation
- Adjacency Matrix
- Incident Matrix

1. Adjacency Matrix :-

Let $G(V, E)$ be a simple graph with 'n' vertices ordered from v_1 to v_n, then the Adjacency matrix $A = [a_{ij}]_{n \times n}$ of G is an $n \times n$ Symmetric matrix defined by

$$a_{ij} = \begin{cases} 1 \longrightarrow \text{if } v_i \text{ is adjacent to } v_j \\ 0 \longrightarrow \text{otherwise} \end{cases}$$

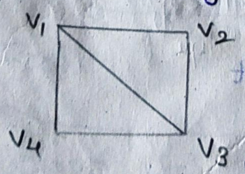
(or)

$$a_{ij} = \begin{cases} 1 \longrightarrow \text{if } v_i, v_j \text{ are Connected by an edge} \\ 0 \longrightarrow \text{other wise} \end{cases}$$

It is denoted as $A(G)$

row, columns both are vertices called adjacent

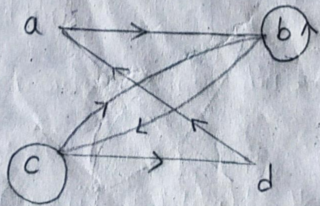
Ex :- (i) Undirected graph



	v_1	v_2	v_3	v_4
v_1	0	1	1	1
v_2	1	0	1	0
v_3	1	1	0	1
v_4	1	0	1	0

(ii) Directed graph

It follows Arrow > of Direction



	a	b	c	d
a	0	1	0	0
b	0	1	1	0
c	0	1	1	1
d	1	0	0	0

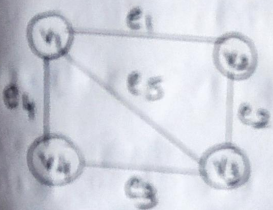
Incident Matrix :-

let $G = (V, E)$ is an unidrected graph with 'n' vertices ordered from $v_1, v_2, v_3 \dots v_n$ and the edges ordered from $e_1, e_2, e_3 \dots e_n$ then the Incident matrix $I_G = [M_{ij}]_{n \times m}$ is given by

Incident graph

$$M_{ij} = \begin{cases} 1 & \text{when the edge is incident to vertex } V_i \\ 0 & \text{otherwise} \\ -1 & \text{when the edge is not incident from } V_i \end{cases}$$

EX :- (i) undirected graph

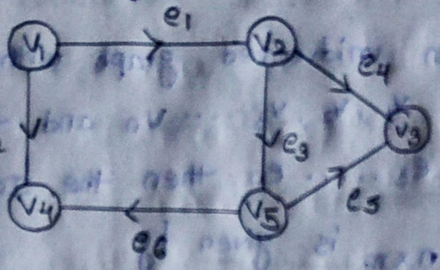


$$\Rightarrow I_G \Rightarrow \begin{array}{c c} & \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Note :-

1. Each Column of Incident matrix Contains exactly two 1's
2. A row with all zero's is represent as Isolated vertex.
3. A row with single 1 is represented as pendent vertex.
4. By Counting number of 1's in a matrix for each row we get the degree value for the vertex.

(ii) Directed graph :-



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

***

Euler's theorem :-

let 'G' be a Connected planar graph with 'e' as edges and 'v' as vertices, where if we have 'r' number of regions then

$$\boxed{r = E - V + 2}$$

$E = r + V - 2$

$V = E - r + 2$

r → number of regions

E → Number of edges in a graph

V → Number of vertices

Proof :-

we shall prove this by using mathematical Induction.

(i) Consider the Basic step and prove it is true

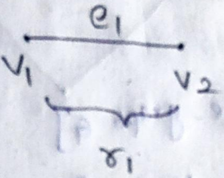
(ii) Assume that the above result is true for 'K' value [Induction step]

(iii) verify the result by proving Basic step is true for $K+1$ values

(i) Basic step :-

Consider we have one region $(n=1)$

Then



$\Rightarrow r_1 = e_1 - V_1 + 2$

$r_1 = 1 - 2 + 2$

$r_1 = 1$

Hence result is true for $n=1$

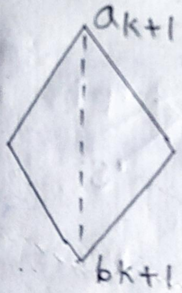
(ii) Induction step :-

Assume the Basic step is true for $n=k$

$\Rightarrow r_K = e_K - V_K + 2$ is true $\rightarrow$ (1)

(iii) Now we have to verify that (1) is true for $n=k+1$

Let (a_{k+1}, b_{k+1}) be the edges that are added to graph G_K



Graph G_K

Case 1 :- Here both the vertices are in G_K

$\Rightarrow V_{K+1} = V_K$ [$\because$ no new vertices are created]

$r_K = r_K + 1$ [new region is generated by adding new edges]

$e_{K+1} = e_{K+1}$ [$\because$ 2 new edges are created / added to the graph G]

Substituting the values in equation (1)

$r_K + \cancel{1} = e_K + \cancel{1} - V_K + 2$

$\Rightarrow \boxed{r_K = e_K - V_K + 2}$

which is assumed to be true as per Induction step.

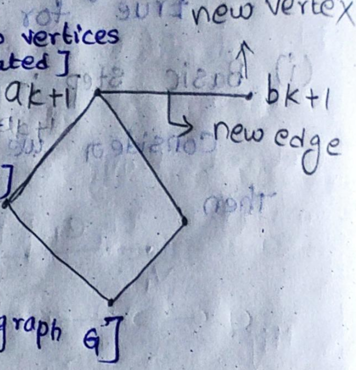
Case 2 :- Here both the vertices are in G_K

$\Rightarrow V_{K+1} = V_K + 1$ [$\because$ no new vertices are created]

$r_K = r_K$ [region is not created even vertex is added]

$e_{K+1} = e_K + 1$

[$\because$ 2 new edges are created / added to the graph G]



Substituting the values in equation ①

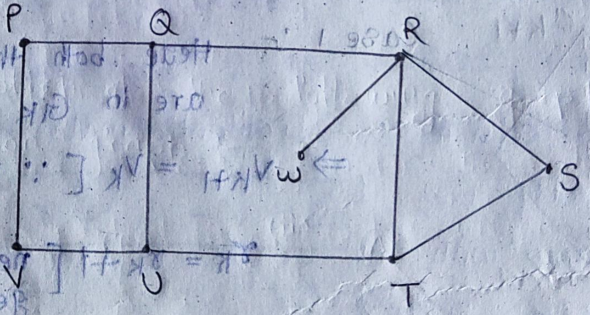
$$r_K = e_{K+\cancel{1}} - V_K - \cancel{1} + 2$$

$$\Rightarrow \boxed{r_K = e_K - V_K + 2}$$

① which is assumed to be true as per Induction step.

By the principle of mathematical Induction Euler's theorem is proved.

<u>Verify the Euler's formula for the following graph</u>



from the given graph number of vertices = 8

Number of Edges = 10

Number of regions that can be obtain from the given graph

$$\boxed{r = E - V + 2}$$

$$r = 10 - 8 + 2$$

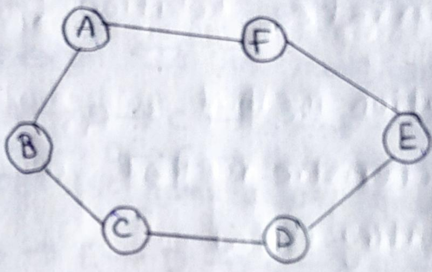
$$r = 2 + 2$$

$$\boxed{r = 4}$$

Hamiltanion graph :- Closed ~~graph~~ walk

In a Connected graph if there exist a Closed walk that visits every vertex of a graph exactly once (except starting vertex) without repetation of edges then that graph is Called as Hamiltanion graph.

Ex :-



(i) Here this graph Contains a closed walk

$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow A$

(ii) Every Vertex is visites exactly once (exactly Starting vertex)

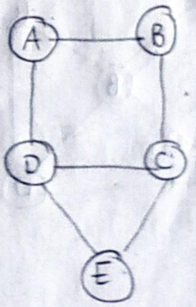
(iii) Edges are not repeated more than once

∴ This graph is a Hamiltanion graph

Hamiltanian path :-

In a Connected graph if there exists a walk that visits the vertices and edges exactly once then that path is Called as Hamiltanion path.

Ex :-



Hamiltanion path

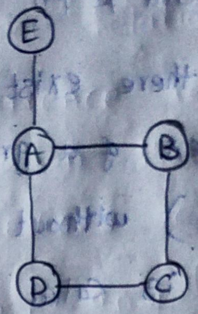
$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$

It is open walk

Hence it is Hamiltanian path

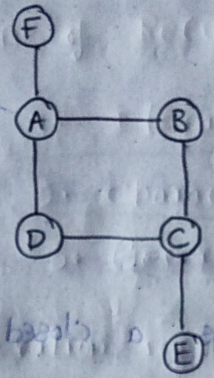
It is not a Hamiltanian graph

Ex :- 2



Hamiltanian path $E \rightarrow A \rightarrow B \rightarrow C \rightarrow D$

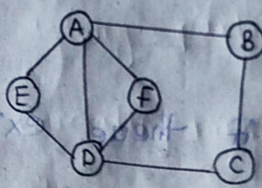
Ex :- 3



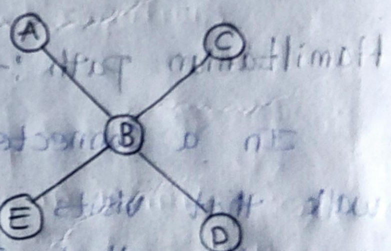
This graph does not contain hamiltanion path because it does not visit all vertices of the graph.

verify which of the following graphs are Hamiltanion graph :-

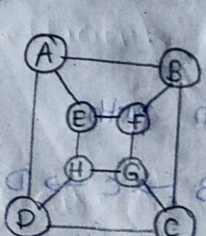
(i)



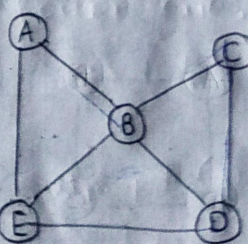
(ii)



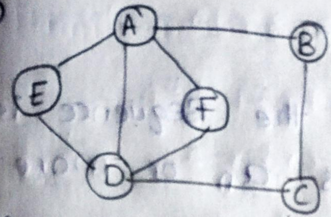
(iii)



(iv)



(i)



No it is not a hamiltanian graph.

This graph doe's not contain hamiltanian path because it does not visit all vertices of the garph.

(ii) No it is not a hamiltanian graph. This graph does'not contain hamiltanian path. because it doe's not visit all vertices of the graph

(iii) ~~AEXB~~. It is Hamiltanian graph.

Path:- $A \to D \to C \to B \to F \to G \to H \to E \to A$

$A \to D \to C \to B \to F \to G \to H \to E$

(iv) It is hamiltanian graph.

$A \to E \to D \to C \to B \to A$

It contain hamiltaian path

$A \to E \to D \to C \to B$

Theorem ① Hand shaking Theorem

Let $G=(V,E)$ be a undirected graph with 'e' edges. Then
$\sum_{v\in V} deg(v) = 2e$.

Proof:- We have to prove sum of degree of all the vertices of an undirected graph is twice number of edges of the graph, and hence even.

We know that since every edge is incident with exactly two vertices, every edge contributes '2' to the sum of degree of the vertices.

$\therefore$ All the 'e' edges contributes ($2e$) to the sum of degree of the vertices

$$\therefore \sum_{v\in V} deg(v) = 2e.$$

Theorem ②:- In a undirected graph the number of odd degree vertices are even.

Proof:- Let V_1 be the set of all vertices of even degree and V_2 be the set of all vertices of odd degree in a graph $G = (V, E)$.

$$\Rightarrow \sum d(v) = \sum_{v_i \in V_i} d(v_i) + \sum_{v_j \in V_j} d(v_j)$$

By Hand shaking theorem we have

$$2e = \sum_{v_i \in V_i} d(v_i) + \sum_{v_j \in V_j} d(v_j) \quad \text{—— ①}$$

Since each degree of (v_i) is even then

$\sum_{v_j \in V_j} d(v_i) =$ even

As left hand side of equation (1) is even and the first expression of equation (1) is even, we have second expression of equation (1) must be even

$\therefore \sum_{v_j \in V_j} d(v_j)$ is even

$\therefore$ Number of vertices of odd degree is even.

Note:- (1) The maximum degree of each vertex in a graph G is $(n-1)$

(2) In an undirected graph the number of odd degree vertices is alway even.

Theorem (3):- The maximum number of edges in a simple graph with 'n' vertices is $\frac{n(n-1)}{2}$.

Proof:- By Handshaking Theorem

$$\sum_{i=1}^{n} deg(v_i) = 2e$$

$$\Rightarrow deg(v_1) + deg(v_2) + deg(v_3) + \cdots\cdots deg(v_n) = 2e \quad \text{--- (1)}$$

We know that maximum degree of each vertex is $(n-1)$

$$\Rightarrow deg(v_1) = n-1, \; deg(v_2) = (n-1), \; deg(v_3) = (n-1)$$

$$|||^{ly} \; deg(v_n) = n-1$$

Substituting these value in equation (1) we have

$$\underbrace{(n-1) + (n-1) + (n-1) + \cdots\cdots + (n-1)}_{n \text{ times}} = 2e$$

$$\Rightarrow n(n-1) = 2e$$

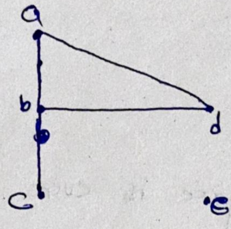
$$\Rightarrow e = \frac{n(n-1)}{2}$$

Hence Proved.

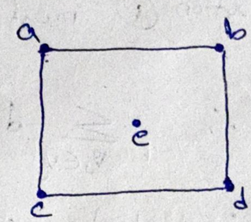
$\therefore$ The maximum number of edges in a Simple graph with 'n' vertices is $\frac{n(n-1)}{2}$.

Degree of a vertex: The number of vertices adjacent to a vertex 'v' in a graph 'G' is called degree of the vertex. denoted as deg (v).

ex:-



} Undirected graph



deg (a) = 2 [as there are 2 edges meeting at vertex 'a']

deg (b) = 3 [as there are 3 edges meeting at vertex 'b']

deg (c) = 1 [as there is 1 edge formed at vertex 'c']

deg (d) = 2 [as there are 2 edges meeting at vertex 'd']

and deg (e) = 0

2 Types in

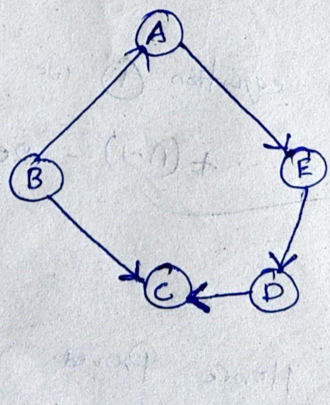
Degree of vertex

Indegree | out degree

Indegree:- The number of edges which are coming into vertex 'v'. denoted as $deg^-(v)$.

Outdegree:- The number of edges which are going out from the vertex v. denoted as $deg^+(v)$

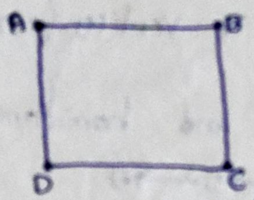
ex:-



Vertex	Indegree	outdegree
A	1	1
B	0	2
C	2	0
D	1	1
E	1	1

1. **Dirac's theorem:-** If G is a simple graph with 'n' vertices ($n \geq 3$) such that every vertex degree in 'G' is atleast $n/2$ then that Graph is a Hamiltonian cycle. where 'n' is the number of vertices in a graph G.

Sol:- Consider a simple Graph with '4' Vertices ($n \geq 3$) as



The Hamiltonian Path for the graph is

$A \to B \to C \to D \to A$

$\to$ no of vertices

Here $n = 4$ ($n \geq 3$) $n/2 = \frac{4}{2} = 2$

$\therefore$ we have to verify that degree of each vertex is atleast $n/2 = 2$ (as per our example graph).

$deg(A) = 2$ $deg(B) = 2$ $deg(C) = 2$ and $deg(D) = 2$

Since the degree of all the vertices exactly equals to $n/2$

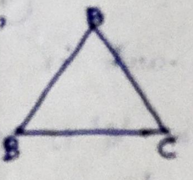
$\therefore$ we can say that the given graph 'G' Contains a Hamiltonian Circuit

$\therefore$ The theorem is Proved

2. **Ore's Theorem:-** If 'G' is a simple graph with 'n' Vertices ($n \geq 3$) such that for every pair of non-adjacent vertices u and v $deg(u) + deg(v) \geq n$; where 'n' is the no of vertices in graph G.

Sol:- Consider a simple Graph with 9 vertices ($n \geq 3$).

(ii)

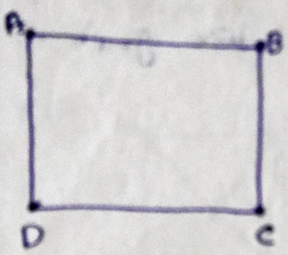


$n=3$ (number of vertices)

But there are no non adjacent vertices in the graph.

So this simple Graph with $(n \geq 3)$ vertices does not hold

Now Consider a simple Graph with $n=4$ vertices $(n \geq 3)$.



This is a Simple Graph and hamiltonian path is $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$

Clearly $n \geq 3 \Rightarrow 4 \geq 3 \Rightarrow$ True

Now Consider a pair of non-adjacent vertices in the above graph. (i) (A, C) (ii) B, D

(i) $\deg(u) + \deg(v) \geq n$ here $n=4$ (from given Graph)

$\Rightarrow \deg(A) + \deg(C)$ must be greater than or equal to 'n'

$\Rightarrow 2 + 2 = 4 \geq 4$ (True).

(ii) $\deg(B) + \deg(D)$ must be greater than or equal to 'n'

$\Rightarrow 2 + 2 \geq n \Rightarrow 4 \geq 4$ ($\because n=4$ vertices)
↳ no of vertices)

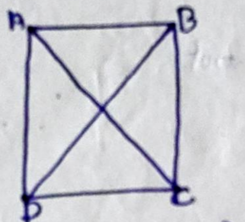
This is also True

$\therefore$ we Can say that if 'G' is a simple graph with $n \geq 3$ vertices such that for every pair of non-adjacency vertices U and V $\deg(u) + \deg(v) \geq n$. where 'n' is the number of vertices in a graph G.

-be Isomorphic if there is one-to-one Correspondence between the vertices and edges such that adjacency of vertices is Preserved.

* Representing a graph in more than one form.

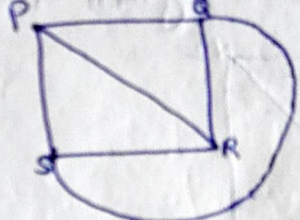
ex:-



(G = (V, E)

is similar to

⟺



(G') = (V', E')

$A \rightarrow B$	⟺	$P \rightarrow Q$
$B \rightarrow C$	⟺	$Q \rightarrow R$
$C \rightarrow D$	⟺	$R \rightarrow S$
$A \rightarrow D$	⟺	$P \rightarrow S$
$A \rightarrow C$	⟺	$P \rightarrow R$
$B \rightarrow D$	⟺	$Q \rightarrow S$

From the above we notice that

$A \leftrightarrow P$, $B \leftrightarrow Q$, $C \leftrightarrow R$, $D \leftrightarrow S$

$\therefore$ we Can Say that $G \cong G'$
↳ (Isomorphic to)

For any two graphs to be Isomorphic the following '4' Conditions must be satisfied.

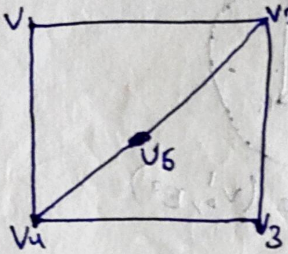
(i) Number of vertices in both the graphs must be same.

(ii) Number of edges in both the graphs must be same.

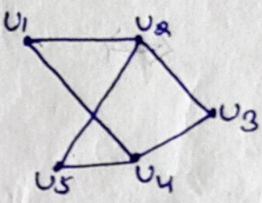
(iii) Degree sequence (In ascending order) must be same.

(iv) If cycle of length K is framed by vertices $\{v_1, v_2, v_3 \cdots v_k\}$ in one graph, then a cycle of same length K must be formed by vertices $\{f(v_1), f(v_2) \cdots f(v_k)\}$ in other Graph as well.

Ex:- Verify given graph is Isomorphic or not



G



G'

(i) No of vertices in $|G| = 5$

(ii) No of edges in $|G| = 6$

(i) No of vertices in $|G'| = 5$

(ii) No of edges in $|G'| = 6$

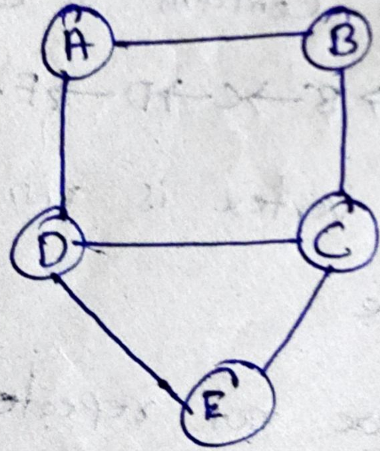
(iii) Degree of vertices has same sequence.

Vertices of G	degree	Vertices of G'	degree
v_1	2	u_1	2
v_2	3	u_2	3
v_3	2	u_3	2
v_4	3	u_4	3
v_5	2	u_5	2

∴ the above 2 Graphs are Isomorphic

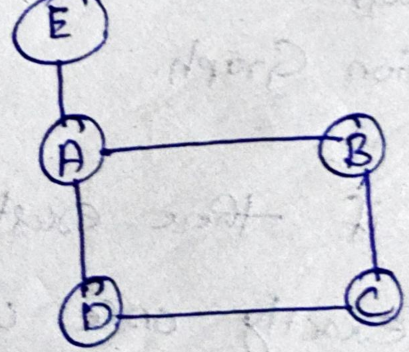
Hamiltonian Circuit:- A hamiltonian Path that starts and ends at same vertex, it is called as Hamiltonian Circuit

ex1:



Hamiltonian Circuit:- A → B → C → E → D → A

ex2:

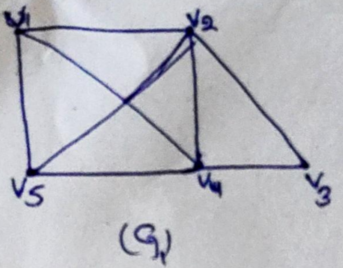


Not a Hamiltonian Circuit

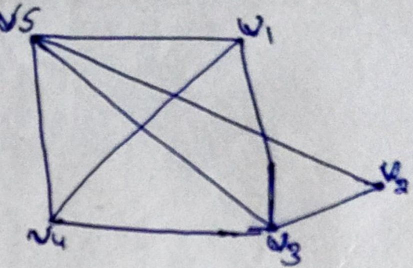
(or)

Hamiltonian Circuit does not Exist

ex: 2 show the following 2 graphs are Isomorphic or not.



(G_1)



G_2

sol:- (i) No of vertices in G_1 = 5

(i) No of vertices in G_2 = 5

(ii) No of edges in G_1 = 8

(ii) No of edges in G_2 = 8

(iii) G_1: $deg(v_1) = 3$
$deg(v_2) = 4$
$deg(v_3) = 2$
$deg(v_4) = 4$
$deg(v_5) = 3$

G_2: $deg(w_1) = 3$
$deg(w_2) = 2$
$deg(w_3) = 4$
$deg(w_4) = 3$
$deg(w_5) = 4$

mapping
$u_1 \rightarrow v_1$ $\quad$ $u_5 \rightarrow v_4$
$u_3 \rightarrow v_2$ $\quad$ $v_4 \rightarrow v_5$
$u_2 \rightarrow v_3$

(iv) Adjacency matrix

G_1

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

G_2

$$\begin{array}{c} \\ u_1 \\ u_5 \\ u_2 \\ u_3 \\ u_4 \end{array} \begin{array}{c} \begin{array}{ccccc} u_1 & u_5 & u_2 & u_3 & u_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

They are Same. So 2 graphs are Isomorphic.

Shortest Path :- In graph theory the shortest path problem is the problem of finding a path between two vertices (or) nodes in a graph such that the sum of weights of constituent edges is minimised.

→ This problem is also called as Single-Pair shortest path problem.

Three variations are there

1. Single-Source shortest path problem
2. Single-destination shortest path problem
3. All pair shortest path problem

(1) Single source shortest path problem we have to find shortest path from a source vertex to all other vertices in a graph.

(2) In single-destination shortest path problem, we have to find shortest path from all vertices in a directed graph to a single destination vertex.

(3) In All pair shortest path we have to find shortest path between every pair of vertices

Flyoid warshal Algorithm :- This is used to solve "All pair shortest path problem".

→ The main requirements are :-

→ graph must be a weighted graph

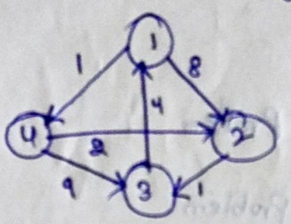
→ Edges weight can be positive / negative.

→ There should be no negative cycles

→ [Negative cycle is a cycle whose edges sum to a negative value]

→ The main advantage of this problem is its simplicity

Ex:- Find the shortest path between every pair of vertices for the following graph using Floyd-warshal algorithm



"Remove all self loops and Parallel edges"

Sol:- Initial distance matrix is → represent distance between every pair of vertices in given weights.

$$D_0 \Rightarrow \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 8 & \infty & 1 \\ 2 & \infty & 0 & 1 & \infty \\ 3 & 4 & \infty & 0 & \infty \\ 4 & \infty & 2 & 9 & 0 \end{array}$$

$$D_1 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 8 & \infty & 1 \\ 2 & \infty & 0 & 1 & \infty \\ 3 & 4 & 12 & 0 & 5 \\ 4 & \infty & 2 & 9 & 0 \end{array}$$

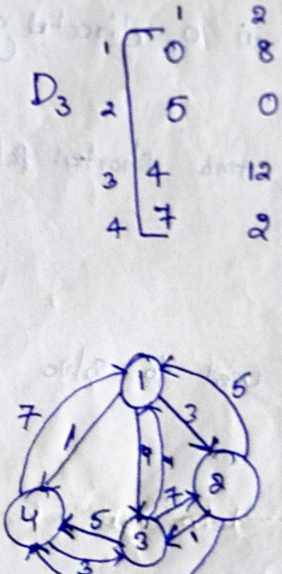
$$D_2 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 8 & 9 & 1 \\ 2 & \infty & 0 & 1 & \infty \\ 3 & 4 & 12 & 0 & 5 \\ 4 & \infty & 2 & 3 & 0 \end{array}$$

$$D_3 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 8 & 9 & 1 \\ 2 & 5 & 0 & 1 & 6 \\ 3 & 4 & 12 & 0 & 5 \\ 4 & 7 & 2 & 3 & 0 \end{array}$$

$$D_4 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 3 & 4 & 1 \\ 2 & 5 & 0 & 1 & 6 \\ 3 & 4 & 7 & 0 & 5 \\ 4 & 7 & 2 & 3 & 0 \end{array}$$

↓

Final shortest distance matrix



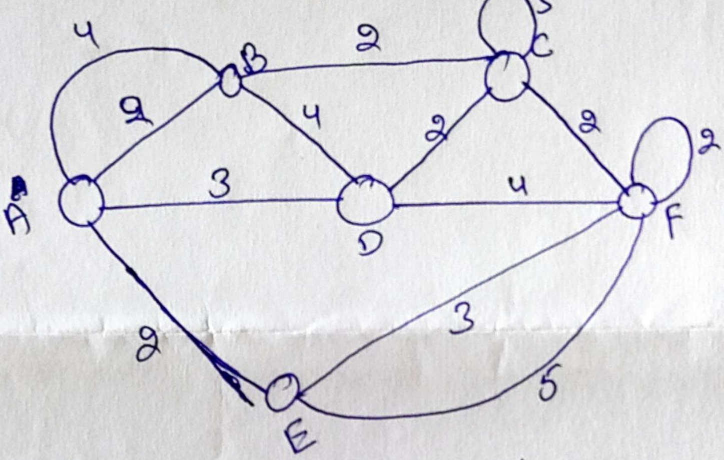
In All Pair shortest Path we Construct the final matrix and stop the Process by Completing weights of all the Pairs

Dijkstar's shortest Path Algorithm

Given a weighted graph and a ~~sou~~ source vertex 's' the algorithm returns the shortest path from 's' to any of other vertices.

→ This algorithm is used to find shortest path between a (source vertex) and b (destination vertex)

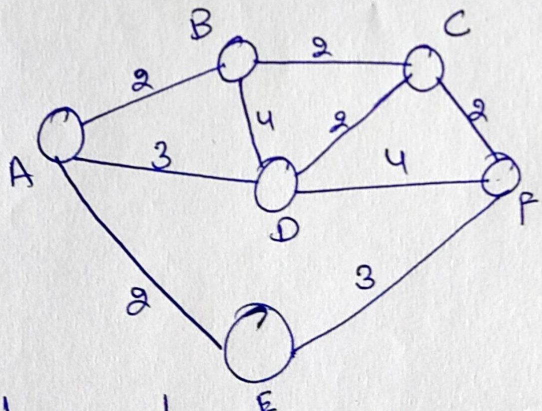
ex:-



(i) Remove all self loops

(ii) Remove Parallel edges (having highest weights)

→



marked	A	$\bar{B}$	$\bar{C}$	$\bar{D}$	$\bar{E}$	$\bar{F}$
A	$\boxed{0}$	∞	∞	∞	∞	∞
B		$\boxed{2}$	∞	3	2	∞
E			4	3	$\boxed{2}$	∞
D			4	$\boxed{3}$		5
C			$\boxed{4}$			5
F						$\boxed{5}$

New distance value = min(old destination value, marked value + Edge weight)

shortest Path :- F E A