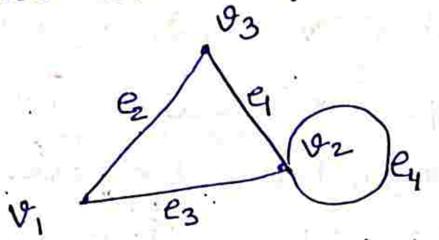
GRAPH

Definition: A graph $G = (V, E)$ consists of V, a nonempty set of vertices (or nodes) and E, set of edges.
Each edge has either one or two vertices associated with it, called its endpoints.

Eg:



$V = \{v_1, v_2, v_3\}$
$E = \{e_1, e_2, e_3, e_4\}$

Remark:- ① The set of vertices V of a graph G may be infinite. A graph with an infinite vertex set or an infinite number of edges is called an infinite graph

② A graph with a finite vertex set and a finite edge set is called a finite graph.

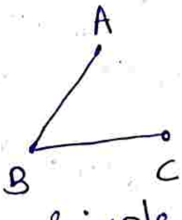
Example: ① Telecommunication of the whole world is example of Infinite graph.

② computer network can be modelled using a graph in which the vertices of the graph represent the data centers and the edges represent communication links.

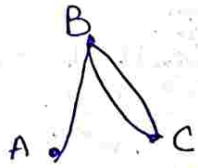
Self loop:- An edge having same vertices at its end points.

Simple graph:- A simple graph is a graph that does not have more than one edge between any two vertices and no edge starts and ends at the same vertex. In other words, a graph does not have any loop or multiple edges.

Eg.



Simple graph



Not a simple graph

Multigraphs: Graphs that may have multiple edges connecting the same vertices are called multigraphs.

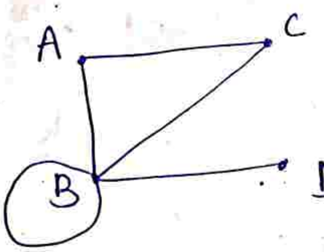
Pseudograph:- A pseudograph is a non-simple graph in which both loops and multiple edges are allowed. *

Null graph: A graph whose edge set is empty. In otherwords, a graph with vertices without edges.

Directed graph:- A directed graph (or diagraph) (V, E) consists of a non-empty set of vertices V and a set of directed edges E. Each directed edge associated with the ordered pair~~(u,v)~~ of vertices. The directed edge associated with the ordered pair (u,v) is said to start at u & end at v.

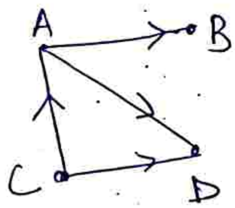
* A directed graph that may have multiple directed edges from a vertex to a second vertex. Such graphs are called directed multigraphs

Eg:

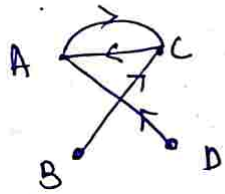


Pseudograph (Undirected graph)

Null Graph



Directed Graph



Directed Multigraph

* Mixed graph: A graph with both directed and undirected edges.

Graph Terminology:-

Type	Edges	Multiple edges	Loops
Simple	Undirected	×	×
Multi	"	✓	×
Pseudo	"	✓	✓
Simple directed	Directed	×	×
Directed graph	"	✓	✓
Mixed graph	Both	✓	✓

* For directed graph.

Def: When (u, v) is an edge of the graph G with directed edges, "u is said to be adjacent to v" and "v is said to be adjacent from u". The vertex u is called initial vertex of (u, v) and v is called the terminal or end vertex at (u, v). The initial vertex and terminal vertex of a loop are the same.

Indegree and out-degree of vertex v.

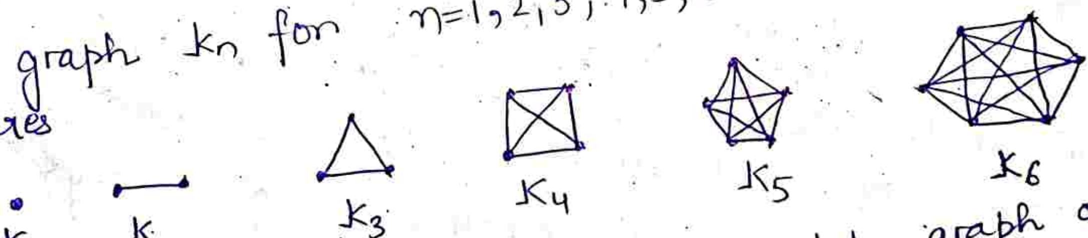
Def^n:- In a graph with directed edges, the indegree of a vertex v, denoted by $deg^-(v)$, is the number of edges with v as their terminal vertex. The out-degree of v, denoted by $deg^+(v)$, is the number of edges with v as their initial vertex.

* For self loop, 1 is indegree and 1 is outdegree

$\boxed{* \sum deg^-(v) = \sum deg^+(v) = |E|}$

Some special simple graphs

1. Complete graphs: A complete graph on n vertices, denoted by K_n, is a simple graph that contains exactly one edge between each pair of distinct vertices. The graph K_n for $n = 1, 2, 3, 4, 5, 6$ are displayed in following figures



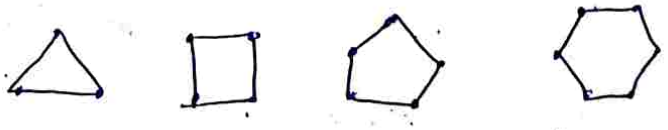
K_1 K_2 K_3 K_4 K_5 K_6

Total number of edges in a complete graph of N vertices

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

2. Cycles: A cycle C_n, $n \geq 3$ consists of n vertices $v_1, v_2, \ldots v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \ldots \{v_{n-1}, v_n\}$ and $\{v_n, v_1\}$.

The cycles C_3, C_4, C_5 and C_6 are:



Basic Terminology of vertices/edges of Undirected graphs.

Def^n 1. Two vertices u and v in an undirected graph G are called adjacent (or neighbours) in G if u and v are endpoints of an edge e of G. Such an edge e is called incident with the vertices u and v and e is said to connect u and v.

Def^n 2. The degree of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that verte[x]

The degree of the vertex v is denoted by deg(v)

$$\boxed{\text{Total degree of a vertex} = \text{incident edges} + 2 * (\text{self loops})}$$

Def^n 3: A vertex of degree zero is called isolated

Def^n 4: A vertex is pendent if and only if it has degree one

Theorem:- The Handshaking Theorem:- Let $G=(V,E)$ be an undirected graph with m edges. Then,

$$2m = \sum_{v \in V} \deg(v) = \text{Total degree of a graph}$$

Theorem:- An undirected graph has an even number of vertices of odd degree.

(Number of odd degree vertices are always even in undirected graph)

Proof:

$$2m = \sum_{v \in V_1} \deg(v) + \sum_{v \in V_2} \deg(v)$$

where V_1 and V_2 be set of vertices of even and odd degree, respectively.

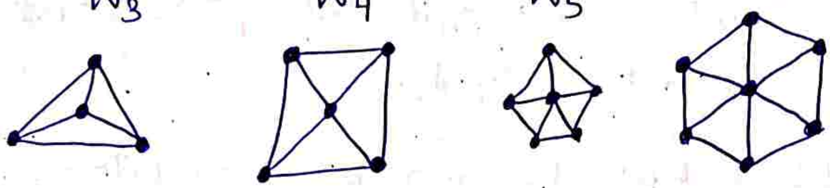
$\Rightarrow$ i.e. $2m = \sum \text{even degree Vertices} + \sum \text{odd deg. vertices}$

$\Rightarrow \sum \text{odd deg. vertices} = 2m - \sum \text{even deg. vertices}$

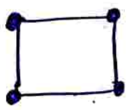
Wheel: We obtain a wheel W_n when we add an additional vertex to a cycle C_n, for $n \geq 3$, and connect this new vertex to each of the n vertices in C_n, by new edges.

A wheel graph is a graph formed by connecting a single universal vertex to all vertex of a cycle

W_3 W_4 W_5 W_6



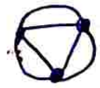
Regular graph:- A graph is called regular graph if degree of each vertex is equal. A graph is called k-regular if degree of each vertex in the graph is k.



2 - regular

3 regular

* It is not a simple graph

Eg.  is 4 Regular graph

Properties: (Cycle)

- A cycle in which only the first and last vertices are equal. All edges in cycle are distinct.
- The number of vertices in C_n equals to the number of edges and every vertex has degree 2 i.e., every vertex has exactly two edges incident with it
- A cycle is a closed walk in which all vertices are distinct but the first and last vertices are same.

(Regular)

- A complete graph with N vertices is (N-1) regular.
- For a k-regular graph, if K is odd, then the number of vertices of the graph must be even.
- Cycle C_n is always 2-Regular
- Number of edge of a k-regular graph with N vertices $= \frac{N*K}{2}$.

Proof :- Let the number of edges of a k-regular graph with N vertices be E.

From Handshaking theorem, we know,

sum of degree of all the vertices = 2 * E

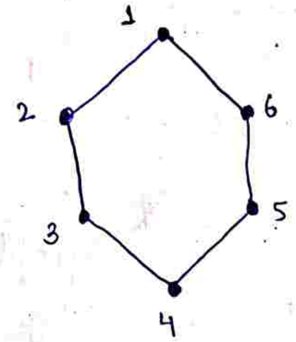
$$N * K = 2 * E$$

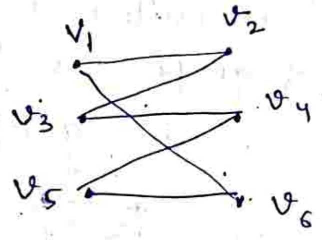
$$\Rightarrow E = (N*K)/2.$$

Bipartite graphs :- A simple graph G is called bipartite if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in a graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2).
When this condition holds, we call the pair (V_1, V_2) a bipartition of a vertex set V of G.

Eg: C_6 is bipartite but K_3 is not bipartite.

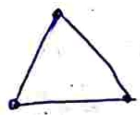
C_6:

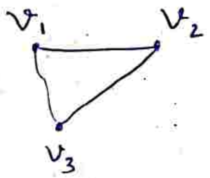




break into two vertices
[1 is connected with 2, so it will be in opposite set of vertices]

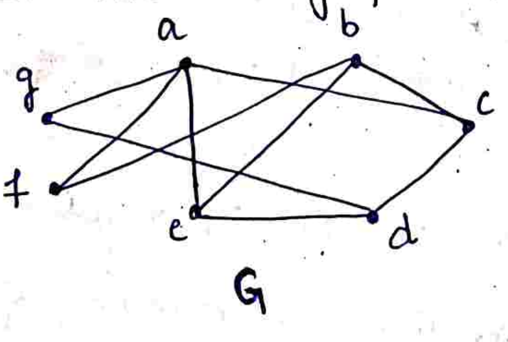
K_3:



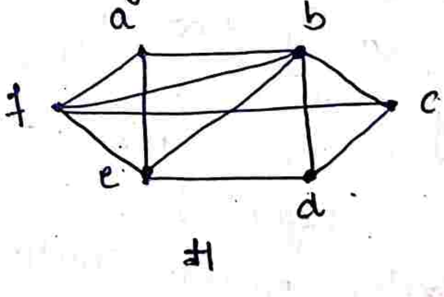


1 is connected with 2, 2 is connected with 3, but 3 is also connected with 1, so can't place in same set of vertices. $\therefore K_3$ is not bipartite.

Question: Are the graphs G and H displayed as follows bipartite?

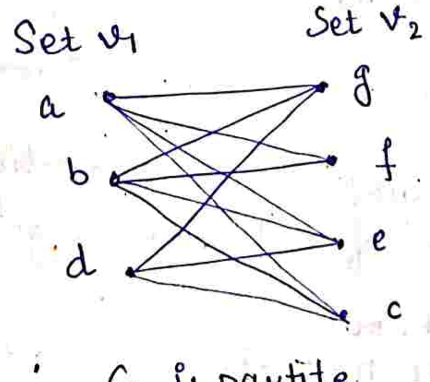


G

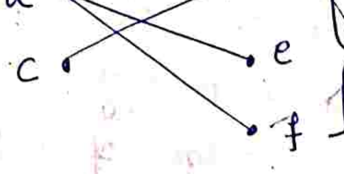


H

Solution :-



$\therefore$ G is partite



but b and f are in same set

$\therefore$ H is not bipartite.

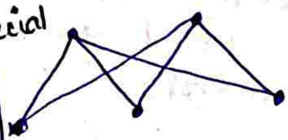
Complete Bipartite Graphs :-

A complete bipartite graph $K_{m,n}$ is a graph that has its vertex set partitioned into two subsets of m and n vertices, respectively with an edge between two vertices if and only if one vertex is in the first subset and the other vertex is in the second subset.

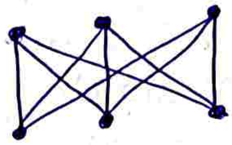
Draw $K_{3,5}$ & $K_{2,6}$

Examples :- and special

It is a simple type of bipartite graph where every vertex of one set is connected to ~~connect~~ every vertex of other set.



$K_{2,3}$



$K_{3,3}$

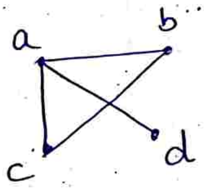
10.3: Representing Graphs and Graph Isomorphism.

Suppose that $G=(V,E)$ is a simple graph where $|V|=n$. Suppose that the vertices of G are listed arbitrarily as $v_1, v_2 \ldots v_n$. The adjacent matrix A (A_G) of G with respect to this listing of the vertices, is the $n \times n$ zero-one matrix with 1 as its (i,j)th entry when v_i and v_j are adjacent, and 0 as its (i,j)th entry when v_i and v_j are not adjacent.

entry when v_i and v_j is an edge of G.

$$A = [a_{ij}] = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G \\ 0 & \text{otherwise.} \end{cases}$$

Eg:

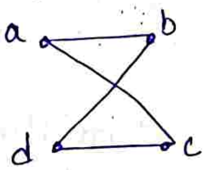


$$A = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

is the adjacent matrix with respect to the ordering of vertices a, b, c, d.

Eg:- Use an adjacency matrix to represent the pseudograph shown in following figures.

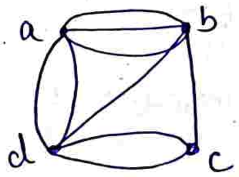
(i)



Adjacency matrix with ~~repres~~, respect to the ordering a,b,c,d.

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

(ii)



Adjacent matrix with respect to the ordering a, b, c, d

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

Important points

1. Note that an adjacency matrix of a graph is based on the ordering chosen for the vertices. Hence, there may be as many as $n!$ different adjacency matrices for a graph with n vertices.

   So, adjacency matrix is not unique.

2. The adjacency matrix of a simple graph is symmetric. Moreover, the principle diagonal entries are zero because, a simple graph has no loops.

3. For simple graph, Adjacent matrix is zero-one matrix
   For non-simple graph, " " is no longer zero-one matrix.

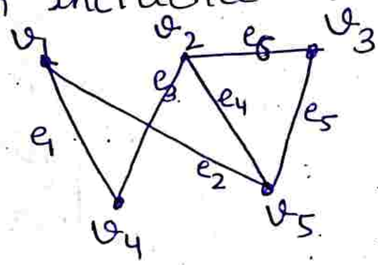
<u>Incidence Matrix</u> :- Another way to represent the graph.

Let $G = (V, E)$ be an undirected graph. Suppose that $v_1, v_2, \dots, v_n$ are the vertices and $e_1, e_2, \dots, e_m$ are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix.

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

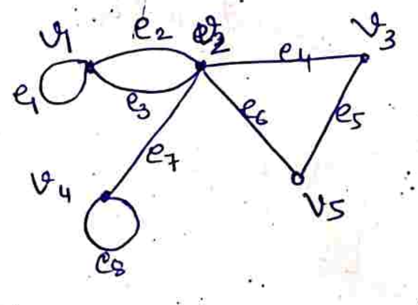
Example:- Represent the graph shown in following figures with an incidence matrix

a)



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

b)

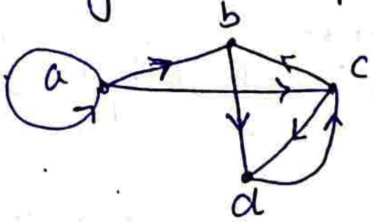


$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \end{matrix} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

* Incidence matrices can also be used to represent multiple edges and loops ⓐ Same columns indicate multiple edges.

* ⓑ Only (Exactly) one entry equal to 1 indicates loops corresponding to the vertex.

Adjacency matrix for directed graph:

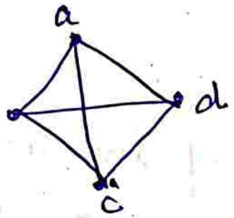


$$A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$ w.r.t. to ordering a, b, c, d

(In this case, Number of 1's = No's of edges.)

Subgraph:- A subgraph of a graph $G=(V,E)$ is a graph $H=(W,F)$, where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a proper subgraph of G if $H \neq G$.

Eg:- K_4

$G =$ 

its subgraph

$H_1 =$ 

$H_1 \neq G$

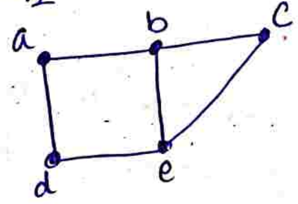
$\therefore$ proper subgraph

proper subgraph.

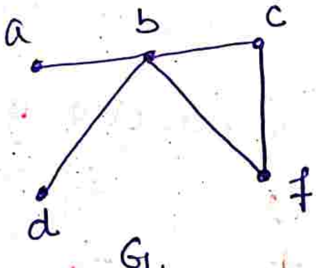
Union and intersection of Graph

The union of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$. The union of G_1 and G_2 is denoted by $G_1 \cup G_2$.

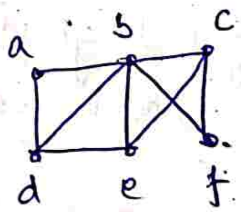
Example:-



G_1



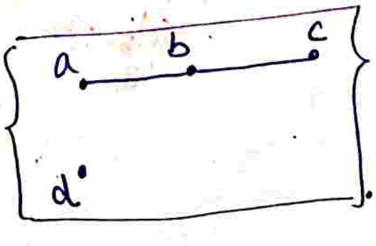
G_1



$G_1 \cup G_2$

The intersection of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cap V_2$ and edge set $E_1 \cap E_2$. The intersection of G_1 and G_2 is denoted by $G_1 \cap G_2$

Example:-

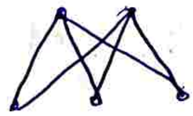
$G_1 \cap G_2 =$ 

The complementary graph: $\overline{G}$ of a simple graph G has the same vertices as G. Two vertices are adjacent in $\overline{G}$ if and only if they are not adjacent in G.

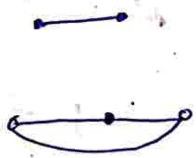
Eg: $\overline{K_n}$: The graph with n vertices and no edges

$\overline{K_{m,n}}$: The disjoint union of K_m and K_n

$K_{2,3}$

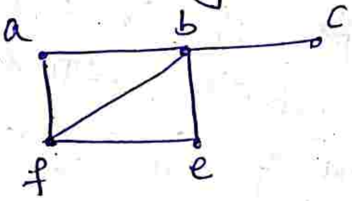


$\overline{K_{2,3}}$



* **Degree sequence :-** A degree sequence of the graph is the sequence of the degrees of the vertices of the graph in non increasing order.

Eg:



degree sequence of the graph G is $[4, 4, 4, 3, 2, 1, 0]$

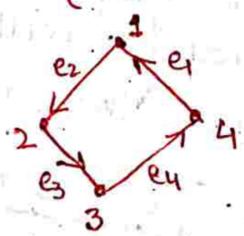
* **Weighted graph:** Graphs that have a number assigned to each edge is called weighted graph.

optional

* **Incidence Matrix of a directed graph :-**

$$M = [m_{ij}] = \begin{cases} -1 & \text{if the ith vertex is an initial vertex} \\ +1 & \text{if the ith vertex is a terminal vertex} \\ 0 & \text{otherwise} \end{cases}$$

Eg: Graph:



$$M = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} e_1 & e_2 & e_3 & e_4 \end{array} \\ \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Connectivity :-

Path: A path is a sequence of edges that begins at a vertex of a graph and travels from vertex to vertex along edges of the graph.

- A path of length n from u to v in G is a sequence of n edges $e_1, e_2, \ldots e_n$ of G for which there exists a sequence $x_0 = u, x_1, x_2, \ldots, x_{n-1}, x_n = v$ of vertices such that e_i has end points x_{i-1} and x_i, for $i = 1, 2, \ldots n$.

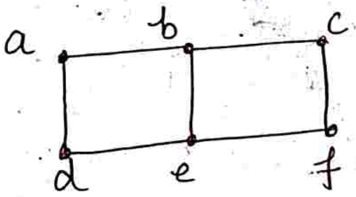
$\boxed{\text{Note}}$ In a path, edges can repeat (Vertices can also repeat).

Circuit The path is circuit if it begins and ends at the same vertex, i.e., if $u = v$, and has length greater than zero.

Simple Path:- A path or circuit is simple if it does not contain the same edge more than once.

Length of path :- Total number of edges in a path.

Example:-



- Simple path of length 4: a b c f e
- a d e c a is not path
- b, c, f, e, b is circuit of length 4
- a, b, e, d, a, b is not simple path.

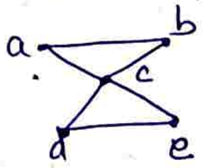
* Walk = Path
* Trail = simple path
* Circuit = closed path
* A circuit in a graph is also called as cycle in a graph.

if ~~only~~ ~~first~~ an en no vertex is repeated except endpoints

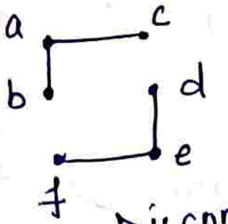
Connected graph: An undirected graph is called connected if there is a path between every pair of distinct vertices of the graph.

* An undirected graph that is not connected is called disconnected.

* Any two computers in the network can communicate if only if the graph of this network is connected.

Eg: 

Connected graph



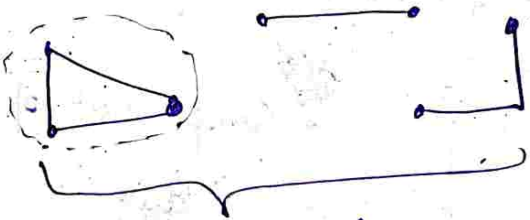
Disconnected graph (because there is no path from 'a' to 'd')

~~Co~~ Theorem:-

* There is a simple path between every pair of distinct vertices of a connected undirected graph.

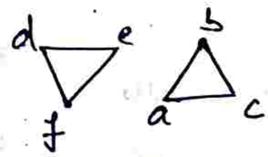
Connected components:- A connected component of a graph G is a connected subgraph of G that is not a proper subgraph of another connected subgraph of G.

* A graph G that is not connected has two or more connected components that are disjoint and have G as their union.

Eg: 

H graph

→ This is disconnected graph and connected components are



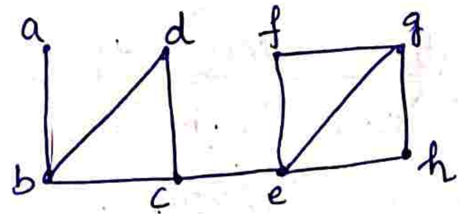
Cut vertex:-

A single vertex whose removal disconnects a graph is called a cut vertex. It is also called articulation points

Cut edge:- An edge 'e' ∈ G is called a cut edge if its removal disconnects a graph. It is also called as bridge.

Example: Find the cut vertices and cut edges in the graph G.

G:



Cut vertices: b, c & e

Cut edges: {a,b} and {c,e}

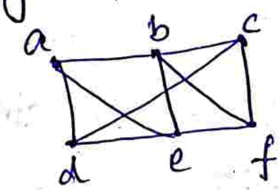
Remark:- Not all graphs have cut vertices; for eg: Complete graph K_n, where $n \geq 3$, has no cut vertices.

optional

* Non-separable graphs:- Connected graph without cut vertices

Vertex cut or separating set :- A vertex subset V' of $G=(V,E)$ is a vertex cut or separating set, if $G - V'$ is disconnected.

Eg: In following ~~above~~ figure, the set {b,c,e} is a vertex cut.



Vertex connectivity:- The vertex connectivity of a noncomplete graph G, denoted by $K(G)$ [Kappa of G], as the minimum number of vertices in a vertex ~~set~~ cut.

Edge cut:- A set of edge E' is called an edge cut of G if the subgraph $G - E'$ is disconnected.

Edge connectivity:- The edge connectivity of a graph G, denoted by $\lambda(G)$ ~~if the subgraph $G-E'$ is disconnected.~~ is the minimum number of edges in an edge cut of G.

Inequality for vertex connectivity and edge connectivity.

$$K(G) \leq \lambda(G) \leq \min_{v \in V} \deg(v)$$

(Connectivity):-

Connectedness in Directed graphs

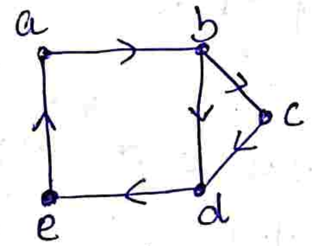
Def^n:- A directed graph is strongly connected if there is a path from 'a' to 'b' and from 'b' to 'a' whenever 'a' and 'b' are vertices in the graph ('every pair')

Def^n:- A directed graph is weakly connected if there is a path between every two vertices in the underlying undirected graph.

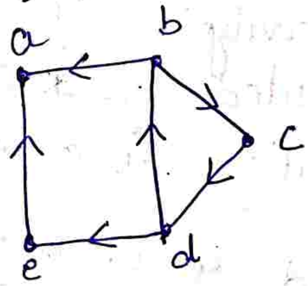
i.e., a directed graph is weakly connected if and only if there is always a path between two vertices when the directions of the edges are disregarded.

* clearly, any strongly connected directed graph is also weakly connected.

Q. Are the directed graphs G and H strongly connected? Are they weakly connected?



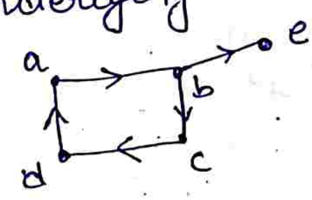
G



H

G is strongly connected. Hence, G is also weakly connected. H is not strongly connected because no directed path from a to b in this graph. However, H is weakly connected, because there is a path between any two vertices in the underlying undirected graph of H.

Eg 2.



Not strongly connected because no path from e to b.

Isomorphism:-

Two simple graphs are isomorphic, if there is a one to one correspondence between vertices of the two graphs that preserves the adjacency relations

F. For isomorphism first check
- no. of vertices
- no. of edges
- degree sequence

b/w two vertices are same?

If yes then build mapping and corresponding adjacency matrices.

Examples:

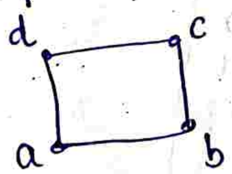
Non isomorphic ∵ vertices count not same

Non isomorphic ∵ edges count not same.

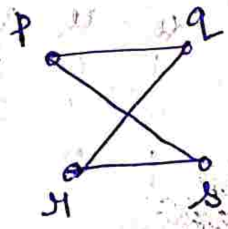
Non isomorphic ∵ degree sequence not equal.

Eg. Show that the graphs $G=(V, E)$ and $H=(W, F)$ are isomorphic

G:

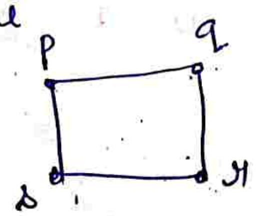


H:



Sol^n:- Reconstruct the complicated figure in form of simple figure

H:



Now mapping

$f(a) = s$

$f(b) = r$

$f(c) = q$

$f(d) = p$

02.03.19

Now adjacency matrix

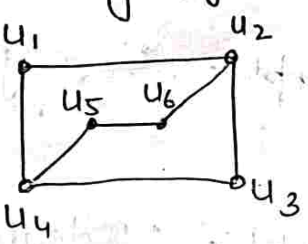
$$A_G: \begin{array}{c|cccc} & a & b & c & d \\ \hline a & 0 & 1 & 0 & 1 \\ b & 1 & 0 & 1 & 0 \\ c & 0 & 1 & 0 & 1 \\ d & 1 & 0 & 1 & 0 \end{array}$$

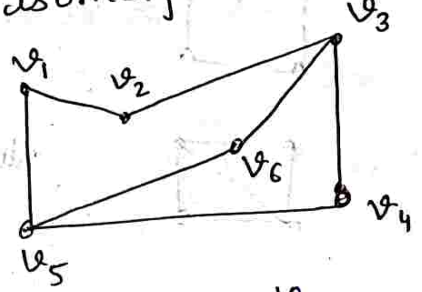
w.r.t. to ordering a, b, c, d

$$A_H: \begin{array}{c|cccc} & s & r & q & p \\ \hline s & 0 & 1 & 0 & 1 \\ r & 1 & 0 & 1 & 0 \\ q & 0 & 1 & 0 & 1 \\ p & 1 & 0 & 1 & 0 \end{array}$$

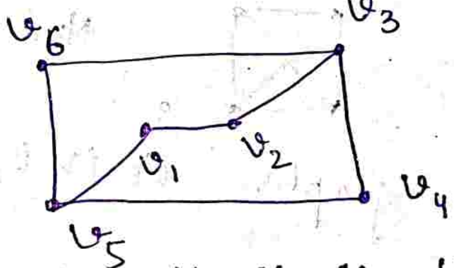
Both the adjacency matrices are same. $\therefore$ the obtained mapping proves that G and H are isomorphic

Example: Determine whether the graphs G and H displayed in following figures are isomorphic.





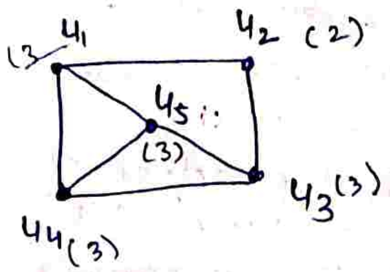
Solution

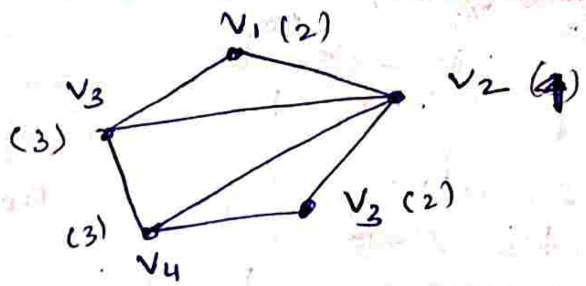


$f(u_1) = v_6$
$f(u_2) = v_3$
$f(u_3) = v_4$
$f(u_4) = v_5$
$f(u_5) = v_1$
$f(u_6) = v_2$

$$\begin{array}{c|cccccc} & u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \\ \hline u_1 & 0 & 1 & 0 & 1 & 0 & 0 \\ u_2 & 1 & 0 & 1 & 0 & 0 & 1 \\ u_3 & 0 & 1 & 0 & 1 & 0 & 0 \\ u_4 & 1 & 0 & 1 & 0 & 1 & 0 \\ u_5 & 0 & 0 & 0 & 1 & 0 & 1 \\ u_6 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}$$

Eg 36

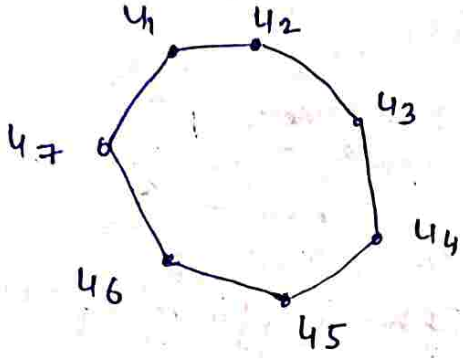


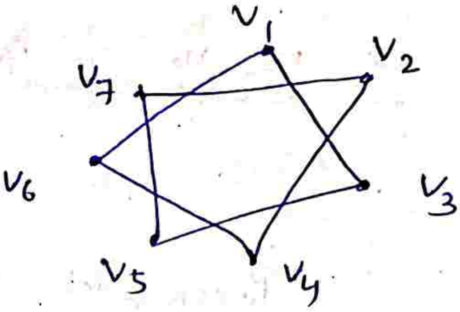


~~Reconstruct U in form of V~~

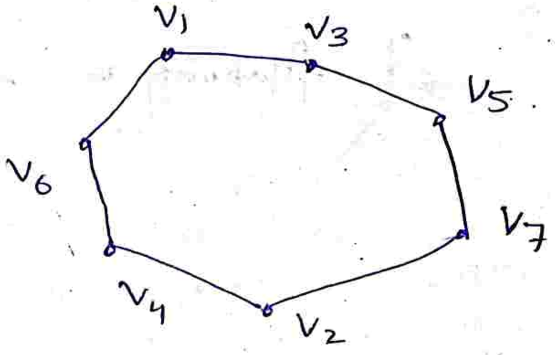
degree sequence is not same
∴ Not an isomorphic graphs

Eg 37





Reconstruct V in form of U



So mapping is

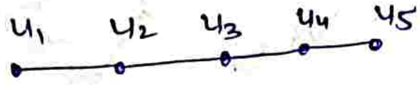
$f(u_1) = v_1$
$f(u_2) = v_3$
$f(u_3) = v_5$
$f(u_4) = v_7$
$f(u_5) = v_2$
$f(u_6) = v_4$
$f(u_7) = v_6$
~~$f(u_8) =$~~

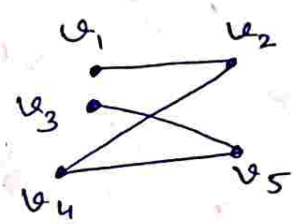
Construct their adjacency matrices and show that they are equal according to ordering of mapping. Which further proofs that graphs are isomorphic.

$$\begin{array}{c|cccccc} & v_6 & v_3 & v_4 & v_5 & v_1 & v_2 \\ \hline v_6 & & 1 & & 1 & & \\ v_3 & 1 & & 1 & & & 1 \\ v_4 & & 1 & & 1 & & \\ v_5 & 1 & & 1 & & 1 & \\ v_1 & & & & 1 & & 1 \\ v_2 & & 1 & & & 1 & \end{array}$$

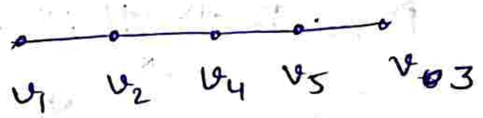
Both the adjacent matrix are same. Thus, defined mapping shows that graphs are isomorphic.

Example: 34





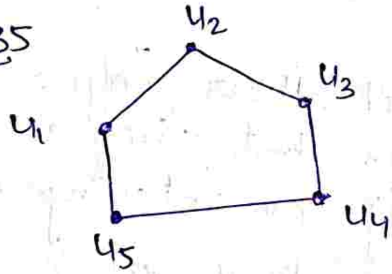
Reconstruct V graph in form of U

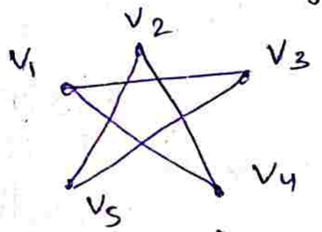


∴ Mapping is $f(u_1)=v_1$
$f(u_2)=v_2$
$f(u_3)=v_4$
$f(u_4)=v_5$
$f(u_5)=v_3$

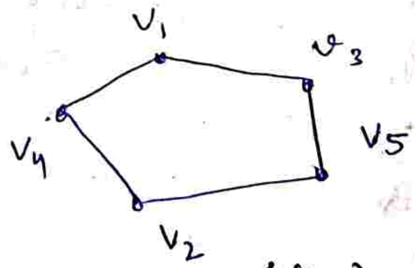
Now check its adjacency matrices are equal?

Example 35





→ reconstruct like U



∴ Mapping is

$f(u_1)=v_4$
$f(u_2)=v_1$
$f(u_3)=v_3$
$f(u_4)=v_5$
$f(v_5)=v_2$

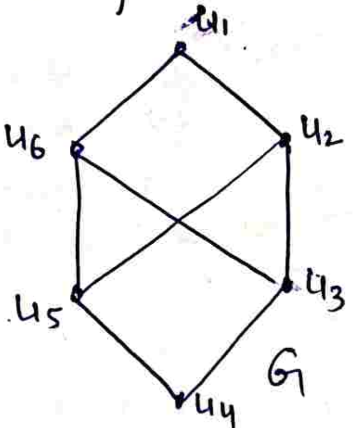
Now Draw their adjacency matrices.

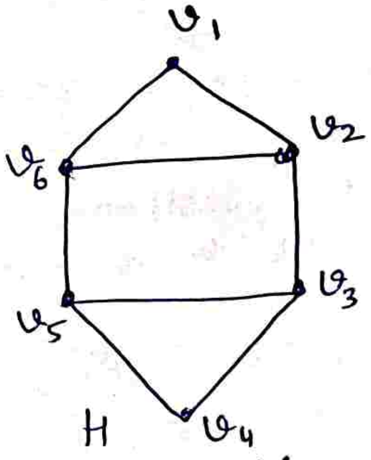
Paths and Isomorphism

Paths and circuits can help determine whether two graphs are isomorphic.
For eg, the existence of a simple circuit of a particular length is a useful invariant that can be used to show that two graphs are not isomorphic.

⊙ A useful isomorphic invariant for simple graphs is the existence of a simple circuit of length k, where k is a positive integer greater than 2.

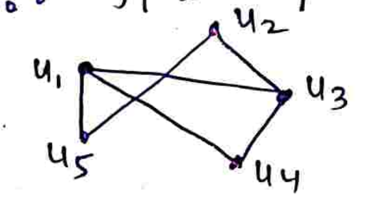
Example :- Determine whether the graph G and H are isomorphic ?

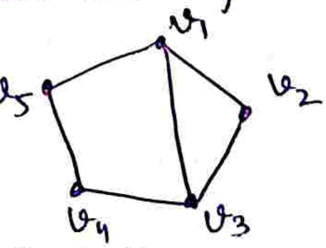




Solⁿ: Both G and H have six vertices and eight edges.
Degree sequence $(3,3,3,3,2,2)$ is same.
So three invariants - no. of vertices, edges & degree sequence agrees for the two graphs.
However, H has a simple circuit of length three, $v_1 v_2 v_6 v_1$ whereas G has no simple circuit of length three.
∴ H and G are not isomorphic.

Ex2:





Solⁿ: No. of vertices - 5 in G & H
No. of edges - 6 in G as well as H
Degree seqⁿ of G : $[3,3,2,2,2]$ same as H
Both have a simple circuit of length 3, 4 and 5.
So G and H MAY be isomorphic.

Dijkstra's Algorithm :-

It is used to find the shortest path between two vertices in a connected weighted simple graph.

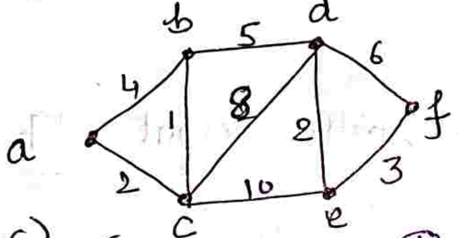
Remark:- ① All edges weights must be non-negative
② Remove self loop and parallel edge
③ Applicable on weighted graph

Note:- If source vertex is given then start from that vertex, otherwise, start from any vertex.

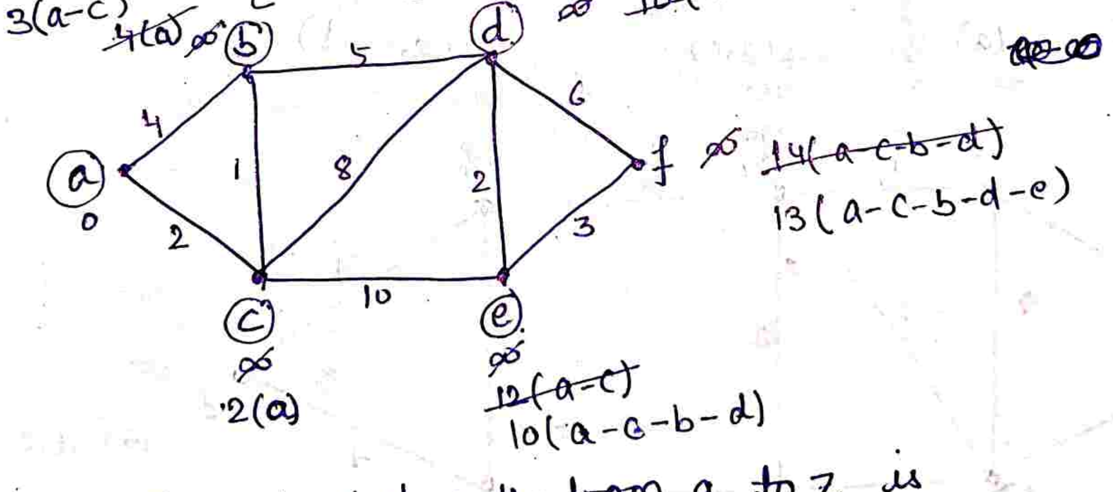
Procedure:-

1. Begins by labelling '0' to source vertex and '∞' to other vertices.
2. Calculate the distance (d) of adjacent vertices and update it '∞' to 'd'.
3. Visit to vertex with smallest distance and mark it as current vertex and repeat step 2.

Example:- find shortest path b/w vertices a and f.

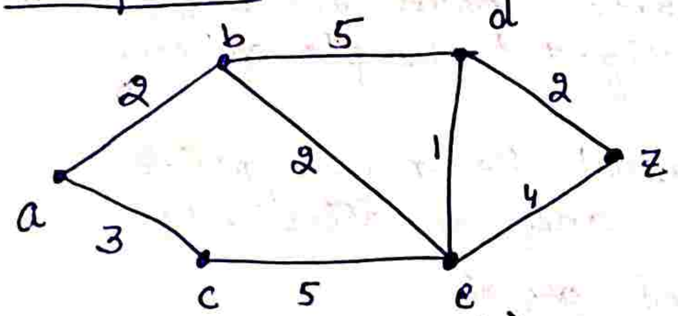


Sol^n:

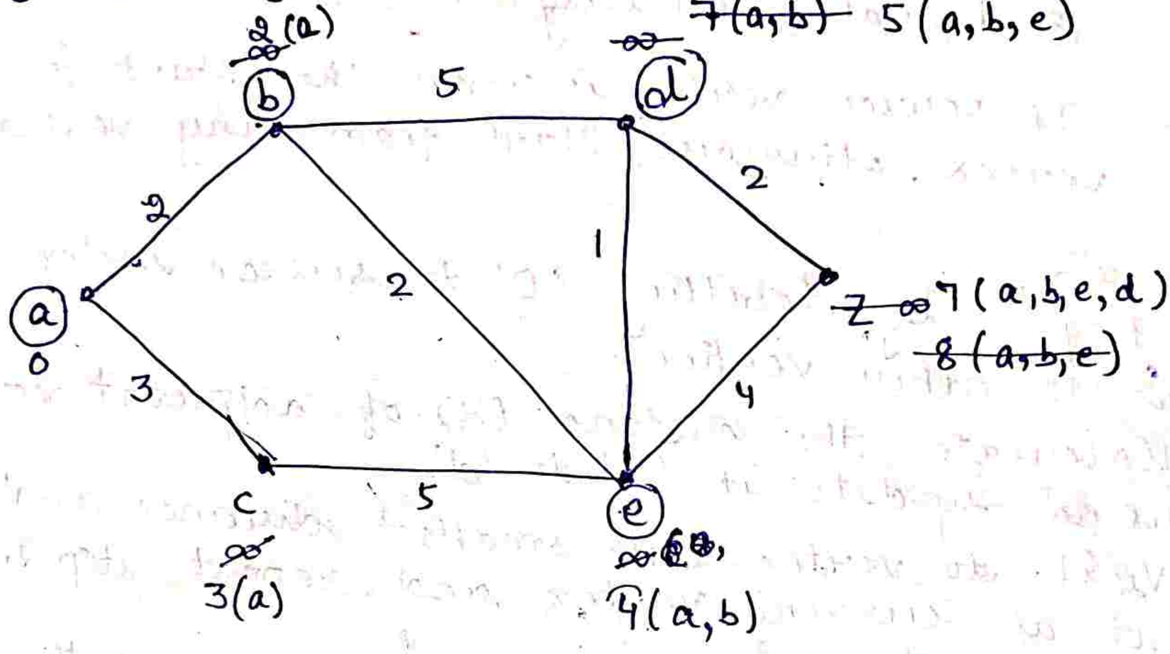


The shortest path from a to z is a-c-b-d-e-z with length 13.

Example 2 Find shortest path between a and z

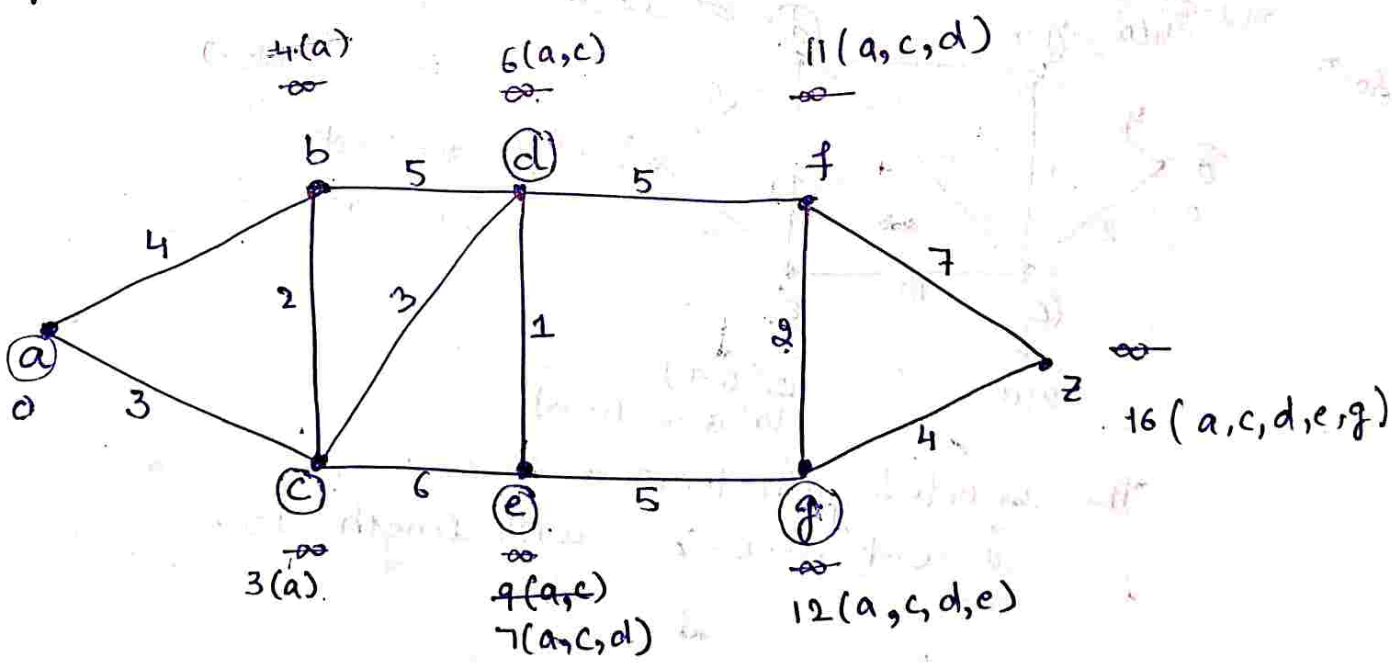


Sol^n



Shortest path from a to z is
a-b-e-d-z with weight 7

Example 3.

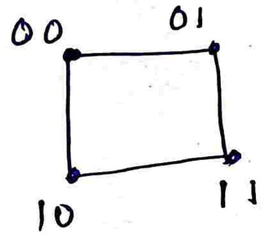


n- Cubes :- An n-dimensional hypercube or n-cube, denoted by Q_n, is a graph that has vertices representing the 2^n bit strings of length n.
Two vertices are adjacent if and only if the bit strings that they represent differ in exactly one bit position.

Eg:- Q_n, $n = 1, 2, 3$

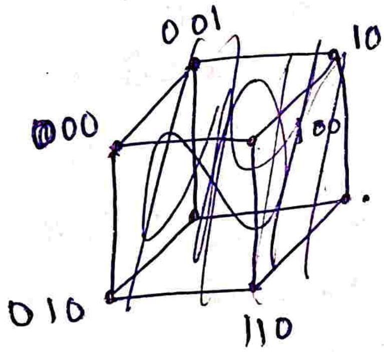
$n=1$ $\quad Q_1$:

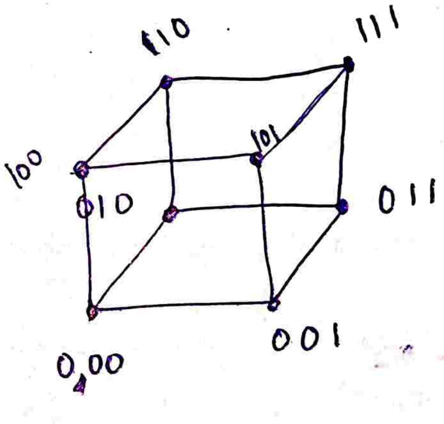
$n=2$ $\quad Q_2$



so ⟸ Two bits possibilities
00] → diff is 1
01] diff is not 1
10
11
↓ diff. is 1

$n=3$ $\quad Q_3$ possibilities
000, 001, 010, 100, 011, 110, 101, 111





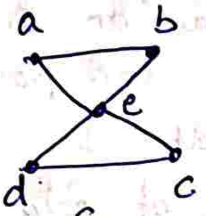
ZYN

Euler and Hamilton Paths

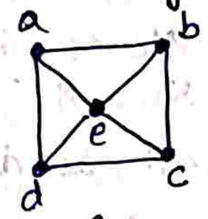
Euler path: An Euler path in G is a simple path containing every edge of G (No repeated edge).

Euler circuit:- An Euler circuit in a graph G is a simple circuit containing every edge of G.

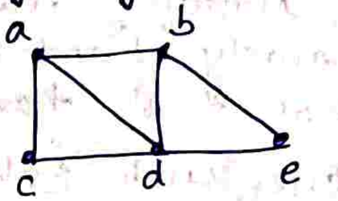
Example:-



G_1



G_2



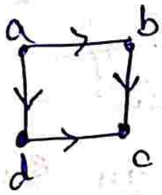
G_3

Solution:

Euler path: — — acdebdab

Euler circuit: aecdeba — —

Example:



No Euler path
& No Euler circuit

Necessary and sufficient conditions for Euler circuits and paths.

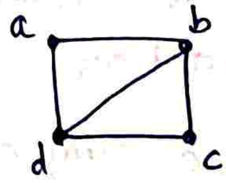
Theorem:- A connected multigraph with atleast two vertices has an Euler circuit if and only if each of its vertices has even degree.

Theorem:- A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree.

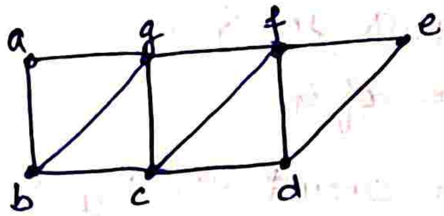
~~Remark~~:
Note:- A graph can't have both Euler paths and Euler circuits

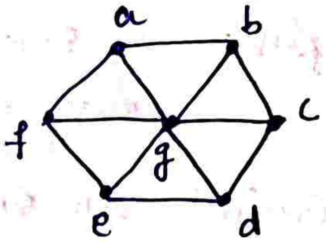
Eulerian graph: A graph which contain Euler circuit is called Eulerian graph.

Q. Which of the following graphs have an Euler path and Euler circuit ?



G_1





Solution :- G_1 contains exactly two vertices of odd degree, namely 'b' and 'd'. Hence it has an Euler path that must have 'b' and 'd' as end points.

One such Euler path is d a b c d b.

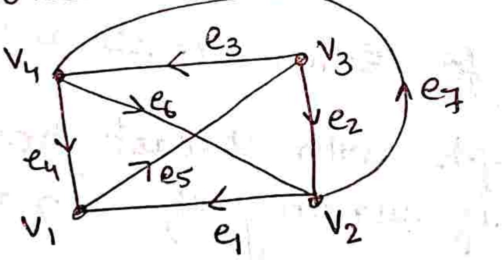
Similarily, G_2 has exactly two vertices of odd degree, namely b and d. So it has an Euler path that must have b and d as endpoints. One such Euler path is

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

G_3 has no Euler path because it has six vertices of odd degree.

G_1, G_2 and G_3 doesn't have Euler circuit.

Q. Does the graph given below possesses an Euler circuit, if it posses then write the Euler circuit.



Solⁿ: Total degree of vertices

$deg(v_1) = 3$

$deg(v_2) = 4$

$deg(v_3) = 3$

$deg(v_4) = 4$

It cannot have Euler circuit because it has exactly two vertices of odd degree. But it has Euler path is given by

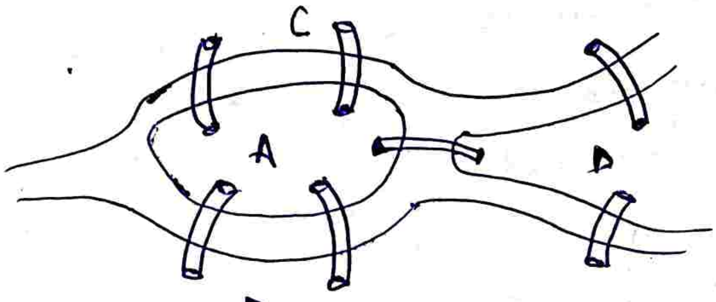
$v_3\, e_2\, v_2\, e_7\, v_4\, e_6\, v_2\, e_1\, v_1\, e_4\, v_4\, e_3\, v_3\, e_5\, v_1$.

Application

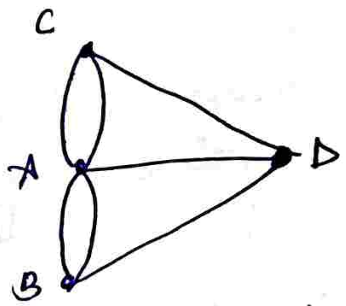
Seven bridges of Konigsberg

There was 7 bridges connecting 4 lands around the city of Konigsberg in Prussia. Was there any way to start from any of the land and go through each of the bridges once and only once?

Soln: Euler first introduced graph theory to solve this problem. He considered each of the lands as a node of a graph and each bridge in between as an edge in between. Now he calculated if there is any Eulerian path in that graph then there is a solution otherwise not.



Seven bridges of Konigsberg



Multigraph Model of problem

Hamilton Paths and Circuit

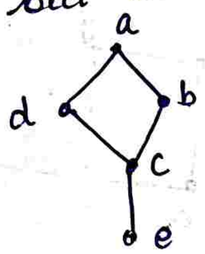
Hamilton Path:- A simple path in a graph G that passes through every vertex exactly once is called Hamilton path.

Hamilton circuit:- A simple circuit in a graph G that passes through every vertex exactly once is called Hamilton circuit.

Hamiltonian graph:- A graph contains Hamiltonian circuit.

Important Points:-

1. A graph can have both Hamilton path as well as Hamilton circuit.
2. If a graph has Hamilton circuit then it also has Hamilton path but converse is not true.



Hamilton path is e, c, d, a, b
But not having Hamilton circuit.

3. Only a connected graph can have Hamilton circuit / Path.
4. A graph with a vertex of degree one cannot have a H.C.

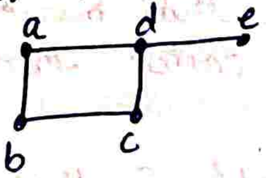
Conditions for the existence of Hamilton circuit

1. DIRAC's Theorem:- If G is a simple graph with n vertices with $n \geq 3$ such that the degree of every vertex in G is at least $\frac{n}{2}$, then G has a Hamilton circuit.

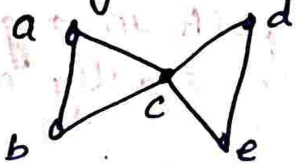
2. ORE'S Theorem:- If G is a simple graph with n vertices with $n \geq 3$ such that $deg(u) + deg(v) \geq n$ for every pair of nonadjacent vertices u and v in G, then G has a Hamilton circuit.

Example:- Show that neither graph displayed below has a H.C.

G:



H:

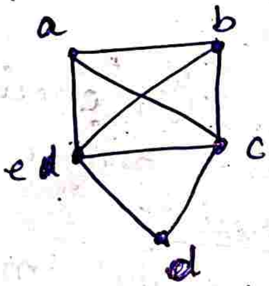


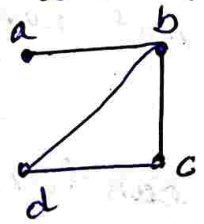
Solution:- There is no Hamilton circuit in G because G has a vertex of degree one.

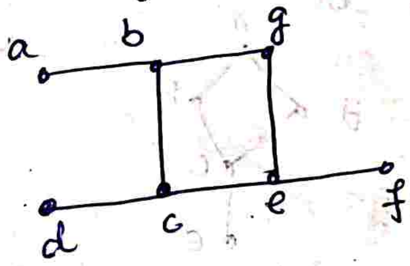
~~In H Graphs a smallest circuit within it exist. Therefore,~~
~~H has no H.C.~~

In H, degrees of the vertices a, b, d, and e are all two, every edge incident with these vertices must be part of any Hamilton circuit. It is now easy to see that no H.C. can exist in H, for any H.C would have to contain four edges incident with c, which is impossible.

Q. Which of the simple graph in following figure have a H.C, or if not, a H.P?







Sol^n: G_1 - has a H.C. a b c d e a

G_2: No H.C. (this can be seen by noting that any circuit containing every vertex must contain the edge {a,b} twice), but it does have a H.P, namely a b c d.

G_3: G_3 has neither H.C. nor H.P. because any path containing all vertices must contain one of the edge {a,b}, {e,f} and {c,d} more than once.