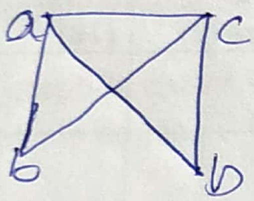
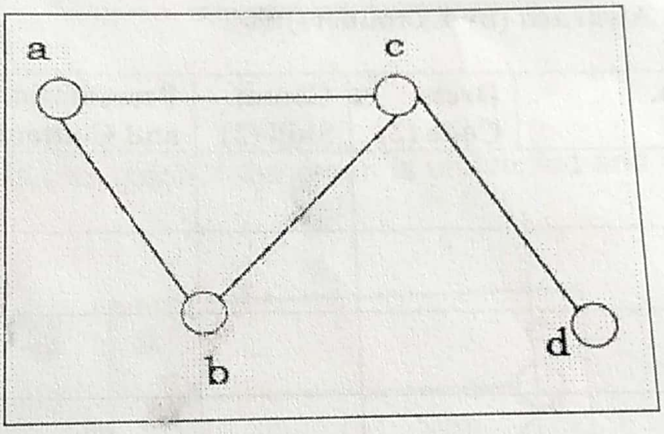
What is a Graph?

Definition – A graph (denoted as $G = (V, E)$) consists of a non-empty set of vertices or nodes V and a set of edges E.

Example – Let us consider, a Graph is $G = (V, E)$ where $V = \{a, b, c, d\}$ and $E = \{\{a, b\}, \{a, c\}, \{b, c\}, \{c, d\}\}$





Degree of a Vertex – The degree of a vertex V of a graph G (denoted by deg (V)) is the number of ed incident with the vertex V.

Vertex	**Degree**	**Even / Odd**
a	2	even
b	2	even
c	3	odd
d	1	odd

n and Odd Vertex – If the degree of a vertex is even, the vertex is called an even vertex and if the egree of a vertex is odd, the vertex is called an odd vertex.

Degree of a Graph – The degree of a graph is the largest vertex degree of that graph. For the above graph the degree of the graph is 3.

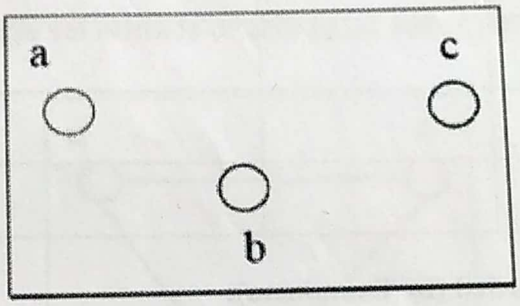
The Handshaking Lemma – In a graph, the sum of all the degrees of all the vertices is equal to twice the number of edges.

Types of Graphs

There are different types of graphs, which we will learn in the following section.

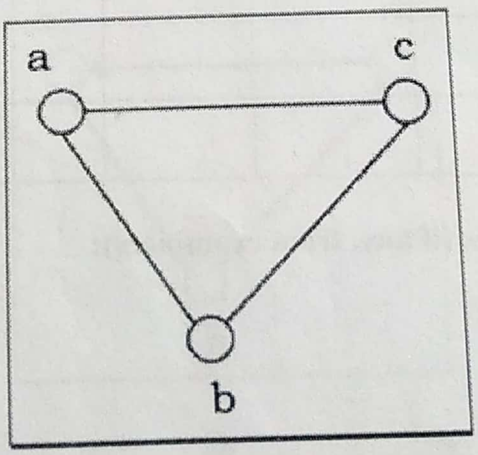
Null Graph

A null graph has no edges. The null graph of n vertices is denoted by N_n



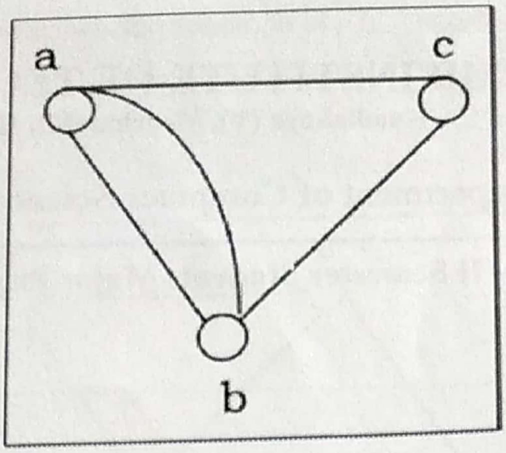
Simple Graph

A graph is called simple graph/strict graph if the graph is undirected and does not contain any loops or multiple edges.



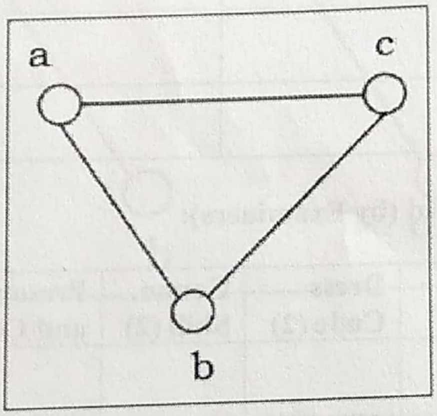
Multi-Graph

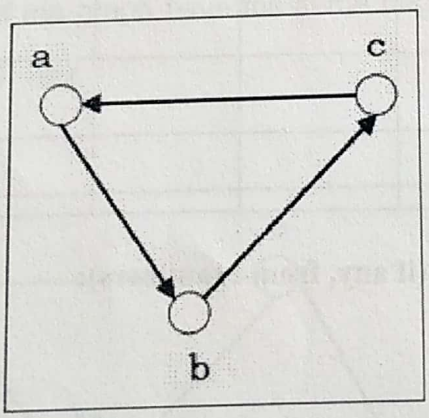
If in a graph multiple edges between the same set of vertices are allowed, it is called Multigraph. In othe words, it is a graph having at least one loop or multiple edges.



Directed and Undirected Graph

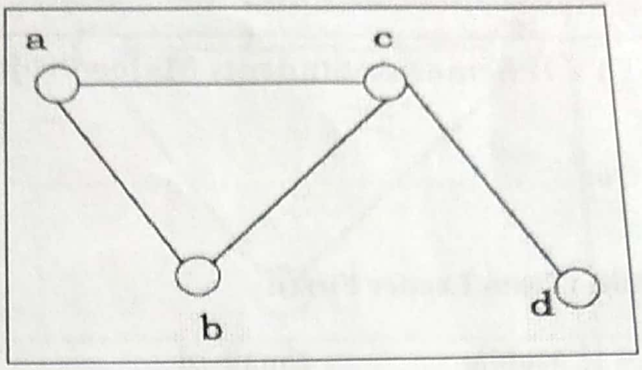
A graph $\boxed{G = (V, E)}$ is called a directed graph if the edge set is made of ordered vertex pair and a graph is called undirected if the edge set is made of unordered vertex pair.

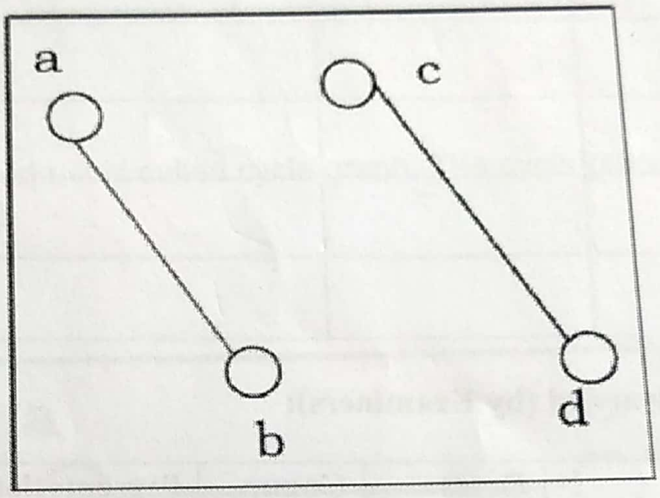




Connected and Disconnected Graph

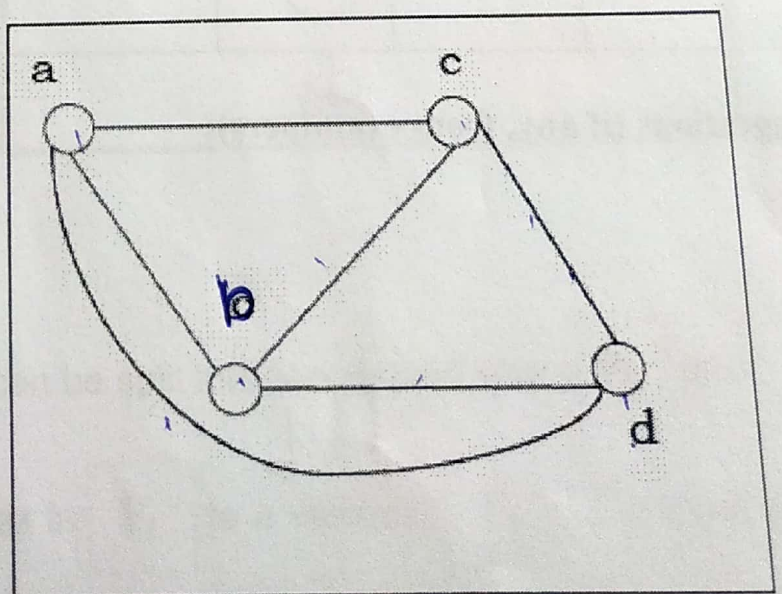
A graph is connected if any two vertices of the graph are connected by a path; while a graph is disconnected if at least two vertices of the graph are not connected by a path. If a graph G is connected, then every maximal connected subgraph of G is called a connected component of the graph G.



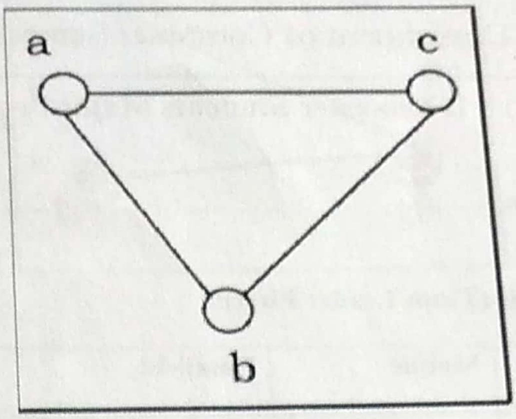


Regular Graph

A graph is regular if all the vertices of the graph have the same degree. In a regular graph G of r, the degree of each vertex of G is r.

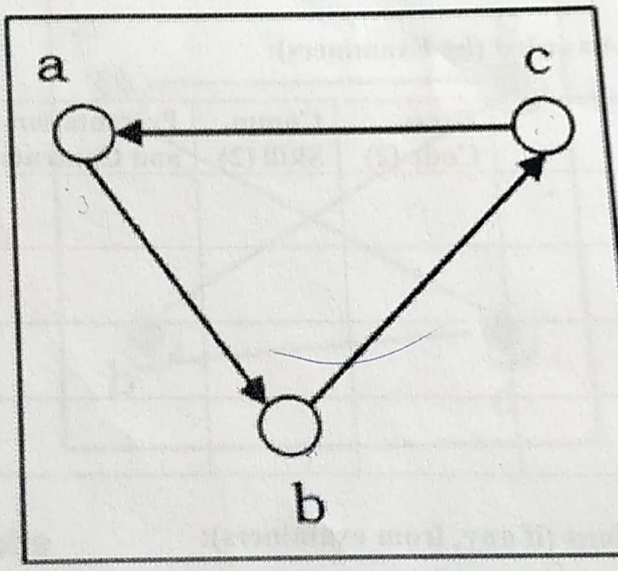


aph is called complete graph if every two vertices pair are joined by exactly one edge. The complete raph with n vertices is denoted by K_n



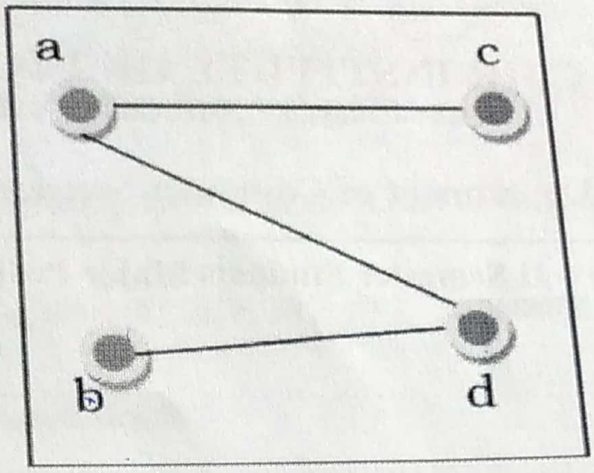
Cycle Graph

If a graph consists of a single cycle, it is called cycle graph. The cycle graph with n vertices is denoted by C_n



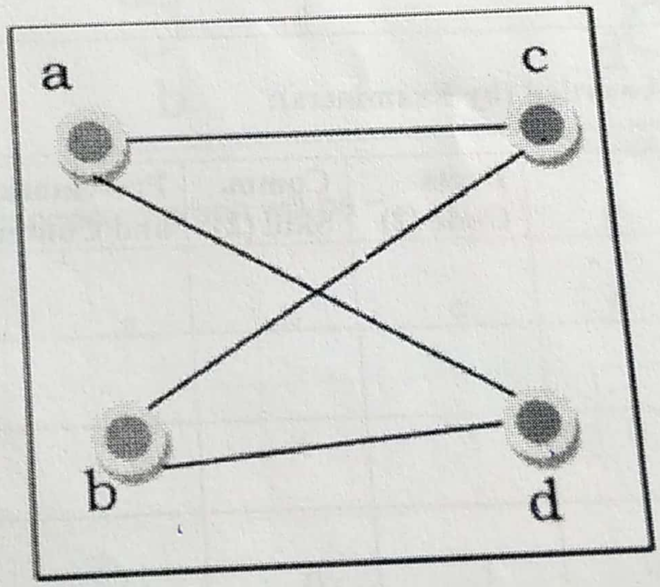
Bipartite Graph

If the vertex-set of a graph G can be split into two disjoint sets, V_1 and V_2, in such a way that ea edge in the graph joins a vertex in V_1 to a vertex in V_2, and there are no edges in G that conn two vertices in V_1 or two vertices in V_2, then the graph G is called a bipartite graph.



Complete Bipartite Graph

A complete bipartite graph is a bipartite graph in which each vertex in the first set is joined to every single vertex in the second set. The complete bipartite graph is denoted by $K_{x,y}$ where the graph G contains x vertices in the first set and y vertices in the second set.



Scanned by CamScanner

SUBGRAPHS

A graph $H = (V_1, E_1)$ is called a <u>subgraph</u> of $G = (V, E)$ if $V_1 \subseteq V$ and $E_1 \subseteq E$

A graph $H = (V_1, E_1)$ is called a <u>proper subgraph</u> of $G = (V, E)$ if $V_1 \subset V$ and $E_1 \subset E$.

$H = (V_1, E_1)$ is called a <u>spanning ~~tree~~ subgraph</u> of G if $V_1 = V$

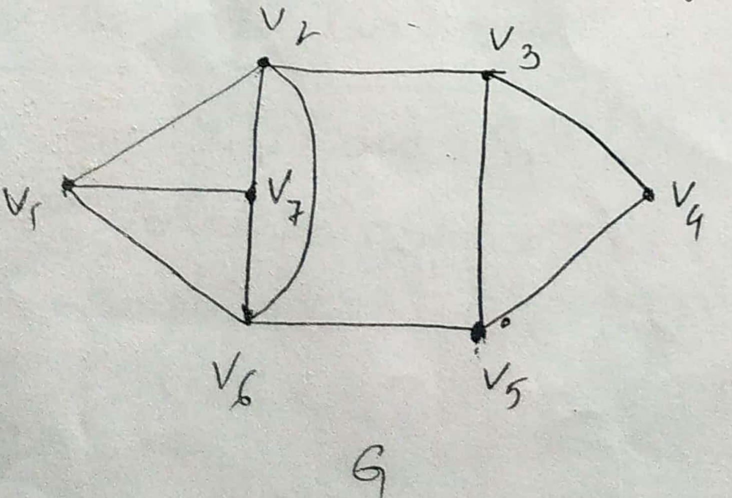
Note :

① If H is a subgraph of G, then

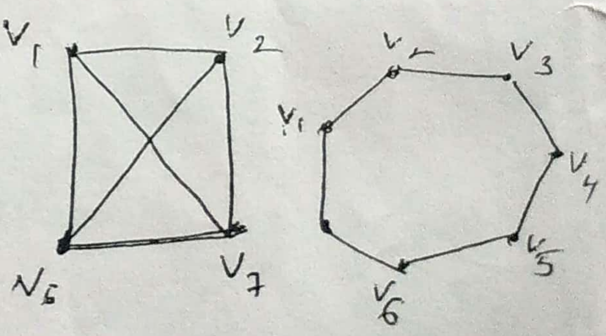
* All the vertices of H are in G
* All the edges of H are in G
* Each edge of H has the same end points in H as in G.

② A spanning subgraph of G need not contain all the edges in G.

Ex ① Consider the graph G given. Show the subgraphs of G

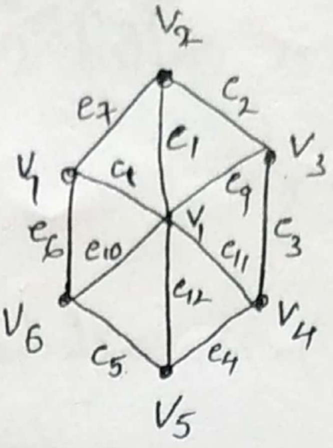


G

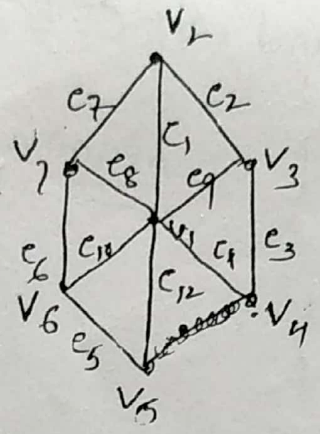


Subgraphs of G.

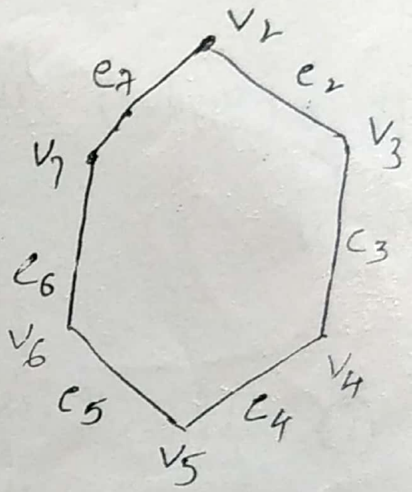
(2) For the graph G draw the subgraphs i, $G - e_4$ ii, $G - V_1$ iii, $G - V_7$



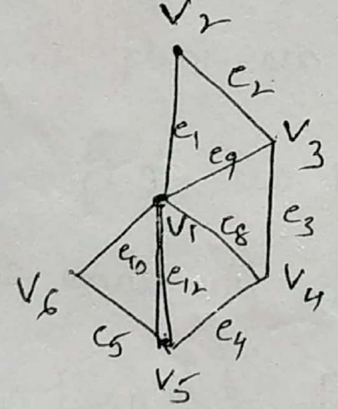
Sol^n: i, $G - e_4$



ii, $G - V_1$



iii, $G - V_7$



OPERATIONS OF GRAPHS

(1) UNION of two graphs

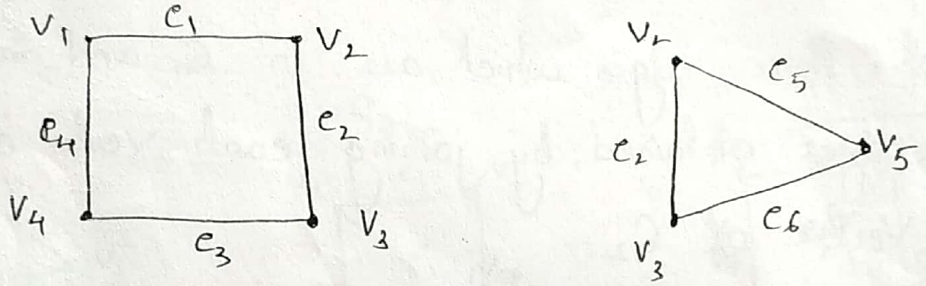
Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs. The union of G_1 and G_2 will be a graph (V, E) such that

$$V = V_1 \cup V_2 \text{ and } E = E_1 \cup E_2$$

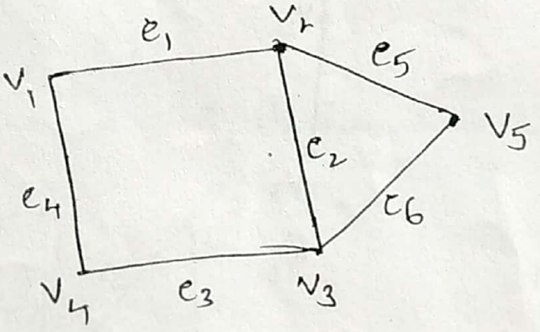
i.e; $G_1 \cup G_2 = (V, E)$ where $V = V_1 \cup V_2$ $E = E_1 \cup E_2$

Ex:

G_1 G_2



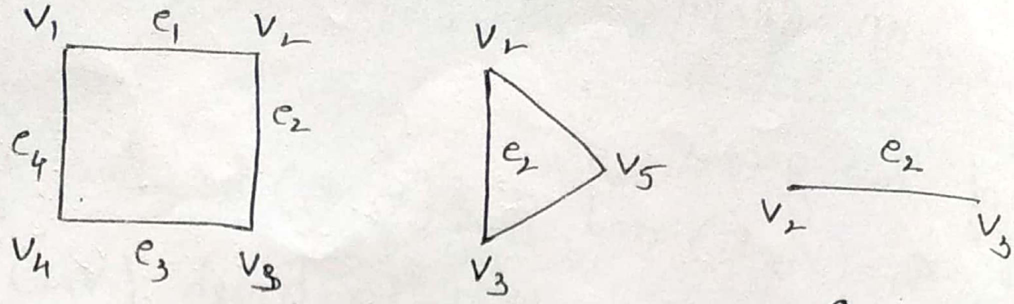
$G_1 \cup G_2$



(2) Intersection of two graphs

Let $G_1 = (V, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs with atleast one vertex in common. The intersection of G_1 and G_2 will be a graph (V, E) such that

$$V = (V_1 \cap V_2) \text{ and } E = (E_1 \cap E_2)$$

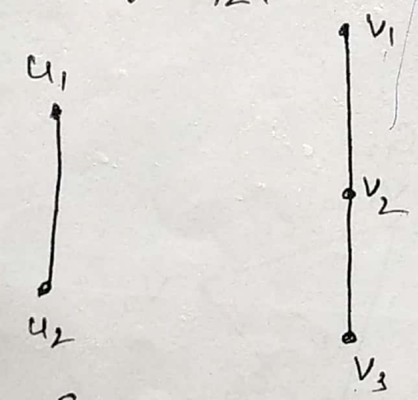


$G_1 \cap G_2$

(3) Sum of two graphs

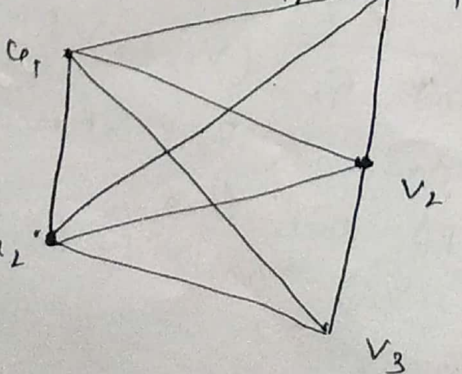
Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $V_1 \cap V_2 = \phi$. The sum $G_1 + G_2$ is defined as the graph whose vertex set is $V_1 + V_2$ and the edge set consists of those edges which are in G_1 and in G_2 & the edges obtained by joining each vertex of G_1 to each vertex of G_2.

Ex:



G_1 G_2

$G_1 + G_2$



i) Ring Sum of two graphs

Let $G = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs. The ring sum of G_1 and G_2 denoted by $G_1 \oplus G_2$ is defined as the graph $G = (V, E)$ such that

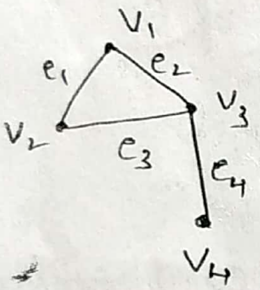
i) $V = V_1 \cup V_2$

ii) $E = E_1 \cup E_2 - (E_1 \cap E_2)$

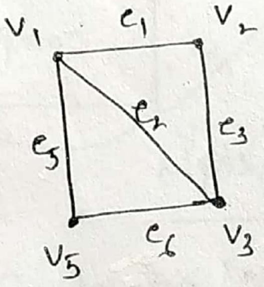
i.e; the edges are either in G_1 or in G_2 but not in both.

Ex:

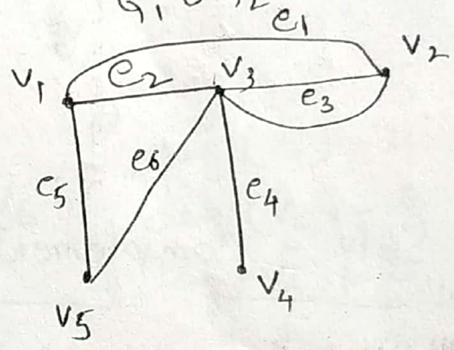
G_1



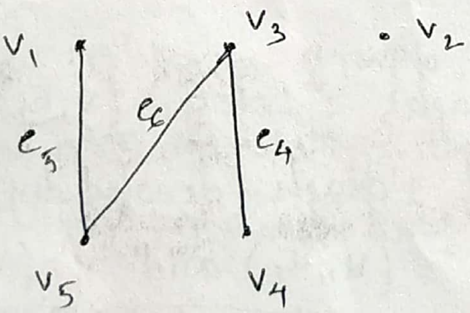
G_2



$G_1 \cup G_2$



$G_1 \oplus G_2$



Incidence and Degree.

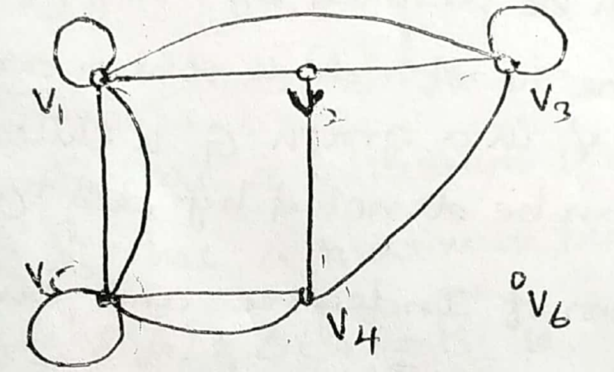
Incidence: Def: Let 'G' be a graph and $e \in E$ then $e = \{u,v\}$ 'e' is said to be incident with vertices u and v.

Ex: $u \xrightarrow[e]{} v$ $e \rightarrow$ incident to u and v

Degree: Let 'G' be a graph and 'v' is vertices of G The degree is obtained from no. of edges which are incident to a vertex 'v'. The Degree of vertex 'v' can be denoted by $deg(v)$, $d_G(v)$ (or) $d(v)$.

* Every loop edge is counted Twice.

Ex:



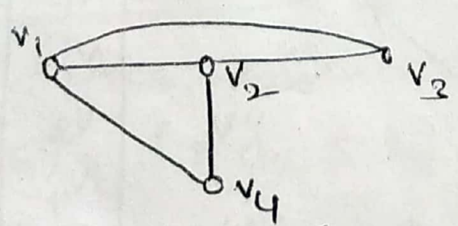
Here
$deg(V_1) = 6$
$deg(V_2) = 3$
$deg(V_3) = 5$
$deg(V_4) = 4$
$deg(V_5) = 6$
$deg(V_6) = 0$

* **Theorm:** Let 'G' be an undirected graph with $|E|$ edges and $|V| = n$ vertices then $\sum_{i=1}^{n} deg(v_i) = 2 \cdot |E|$

Proof: Let 'G' be a graph, with n-vertices (ie, $v_1, v_2 \dots$

When we are finding the degree of all vertices, we are counting each edge Twice.

Ex:



$deg(v_1) = 3$
$deg(v_2) = 3$

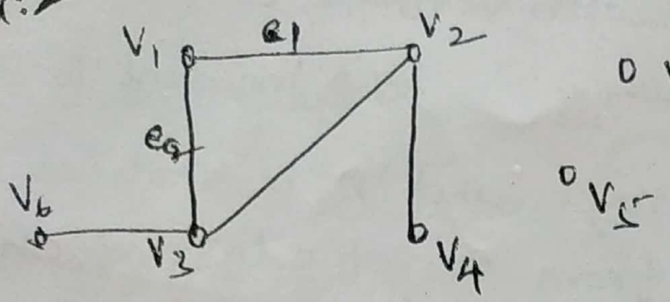
When we are taking the sum of degree of all vertic we count v_1 to v_2 is a edge. Ily v_2 to v_1 is also a Ed From that

$$\sum_{i=1}^{n} deg(v_i) = 2 \cdot |E|$$

Def:

- A vertex of degree one is called Pendent vertex
- A vertex of degree zero is called Isolated vertex

Ex:



Here

v_4 & v_6 are pendent vertices

v_5 & v_7 are Isolated vertices

Def:

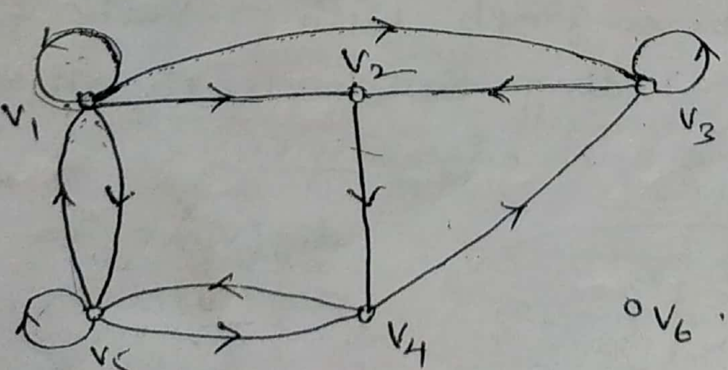
Outdegree: The no. of edges which are incident out from vertex 'v' in a graph 'G' is called outdegree of vertex 'v'. It can be denoted by $\deg^-(v)$.

Indegree: The no. of edges which are incident into of a vertex 'v' in a graph 'G' is called Indegree of vertex 'v'. It can be denoted by $\deg^+(v)$.

→ The sum of Indegree and outdegree is called Total degree. It can be denoted by $\deg(v)$.

$$\boxed{\therefore \deg(v) = \sum_{i=1}^{n} \deg^+(v) + \sum_{i=1}^{n} \deg^-(v)}$$

Ex:



Here

deg	Indegree $\deg^+(v)$	outdegree $\deg^-(v)$
v_1	2	4
v_2	2	1
v_3	3	2
v_4	2	2
v_5	3	3
v_6	0	0

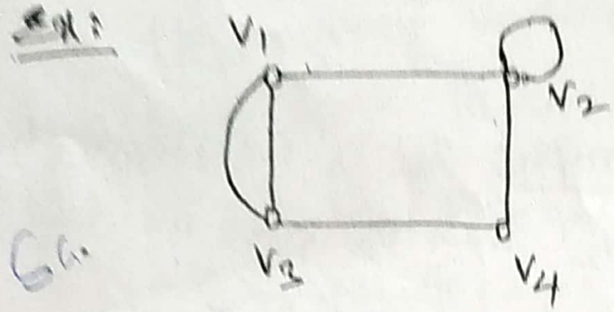
* Every loop on a vertex will be counted in outdegree & Indegree also.

Def:

Minimum Degree: Any vertices which has degree is minimum. It can be denoted by $\delta(G)$.

Maximum Degree: Any vertex which has degree is maximum. It can be denoted by $\Delta(G)$.

Ex:



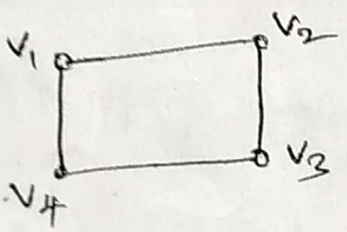
Here
$\deg(v_1) = 3$
$\deg(v_2) = 4$
$\deg(v_3) = 3$
$\deg(v_4) = 2$

$\therefore \delta(G) = 2 \qquad \Delta(G) = 4.$

Vertex 'v_4' has a minimum Degree.
Vertex 'v_2' has a maximum Degree.

Def: If $\delta(G) = \Delta(G) = K$ ie, if each vertex of a graph 'G' has degree K. Then that Graph is called K-regular (or) regular graph of degree K.

Ex:



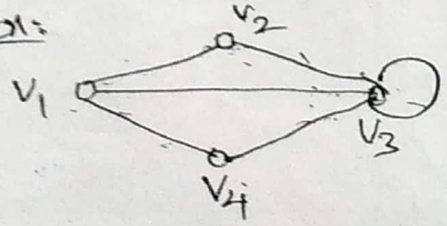
$\deg(v_1) = \deg(v_2) = \deg(v_3) = \deg(v_4) = .$

$\therefore$ It is 2-regular graph.

Degree sequence: Let 'G' be a graph with n-vert then the degree sequence is obtained as

$$\delta(G) = d_1 \leq d_2 \leq d_3 \cdots \leq d_n = \Delta(G)$$

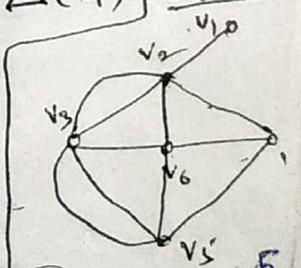
Ex:



$\deg(v_1) = 3$
$\deg(v_2) = 2$
$\deg(v_3) = 5$
$\deg(v_4) = 2$

The degree sequence is $(2, 2, 3, 5)$

Ex: 2

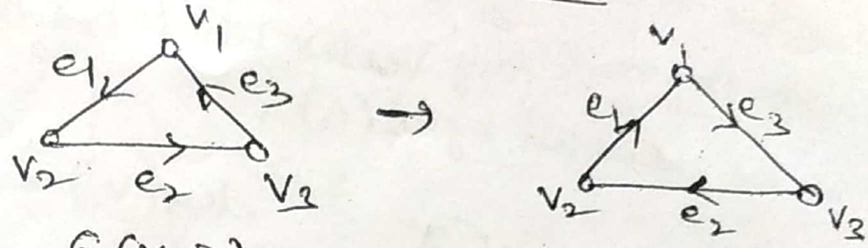


5

Ans: $\{1, 4, 4, 5, 5\}$

➤ **Converse Directed graph:** The converse is obtained from reversing the directions of the edges in G. It can be denoted by $\tilde{G}(V, \tilde{E})$.

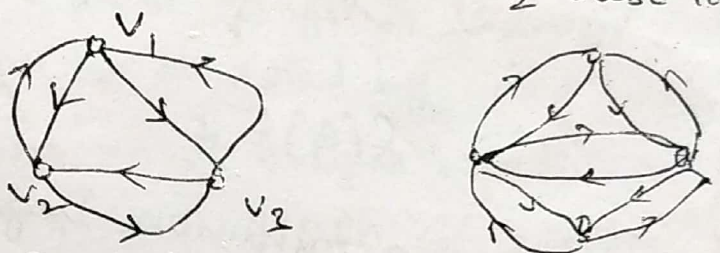
Ex:



G(V,E) $\qquad\qquad$ $\tilde{G}(V,E)$

➤ **Symmetric Directed graph:** ~~Let~~ 'G' is a diagraph. If V_1 is incident to V_2 then V_2 also incident to V_1.

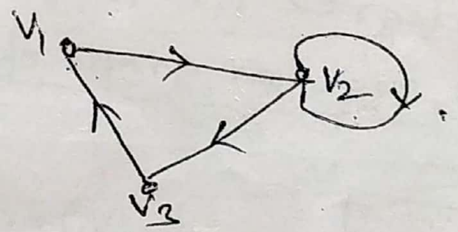
Ex:



➤ **Asymmetric Directed graph:**

If V_1 is incident to V_2 then V_2 is not incident to V_1. Loops are allowed in Asymmetric diagraph.

Ex:

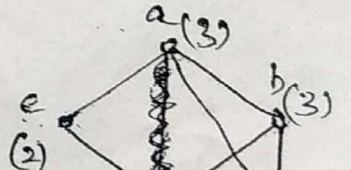


Ex: 1 Does there exists a simple graph with the Degree Sequence {3, 3, 3, 3, 2}?.

Sol: The no. of vertices − 5.

Min degree − 2 − $\delta(G)$

Max degree − 3 − $\Delta(G)$



Ex: 2 Is there a simple graph with the Degree Sequence {1, 1, 3, 3, 3, 4, 6, 7} [Jntu 2000/s]

The no. of vertices − 8.

Min. degree − 1.

Max. degree − 7.

Ex:3 How many vertices does a regular graph of degree 4 with 10 edges?

Sol: Degree of regular graph = 4.

No. of edges in a graph = 10.

No. of vertices = n (assume)

$$\sum \deg(v_i) = 2 \cdot |E|$$

$$n \times 4 = 2 \times 10$$

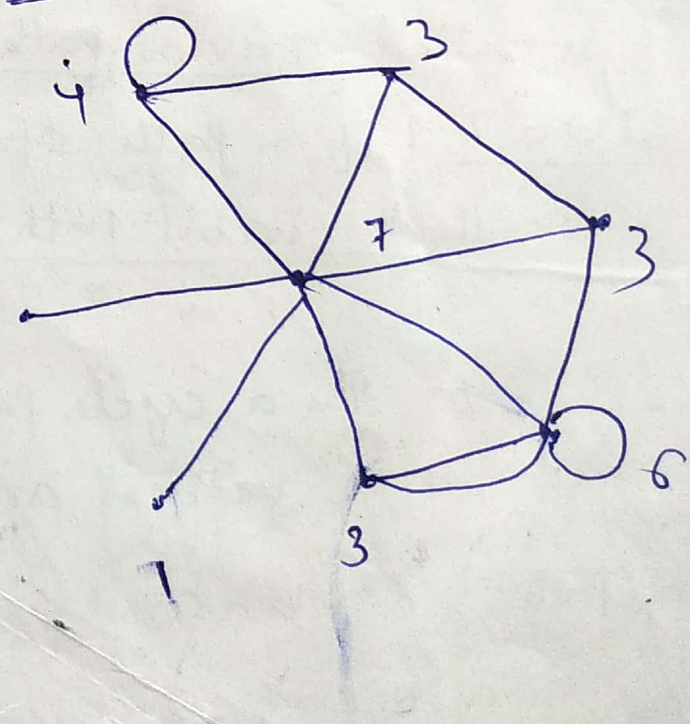
$$n = \frac{20}{4} = 5$$

$$\boxed{n = 5}$$

Ex:2 Answer

~~Not possible.~~

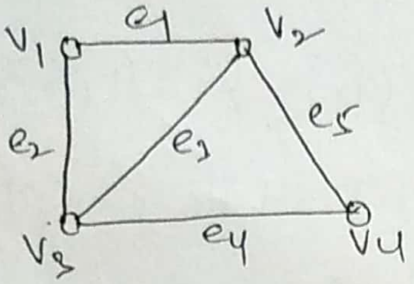
ex:2 Answer



Subgraphs: Let 'G' be a graph. And 'H' is said to subgraph if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.

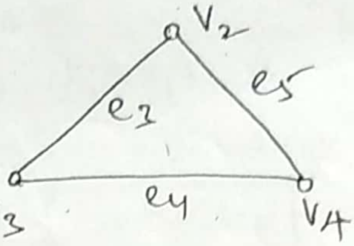
The subgraph of 'G' is denoted by $\underline{H \subseteq G}$.

Ex:



Graph 'G'.

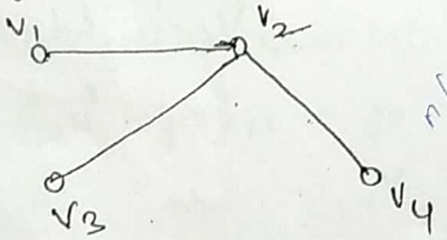
Subgraph is:



Sub Graph 'H'

Def: If $V(H) = V(G)$ then the graph is called spanning sub graph of 'G'.

Ex:

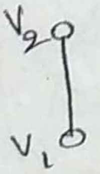


Complete simple graph: A simple graph 'G' is said to be complete simple graph if it has an edge between every pair of vertices.

Ex: If a graph has n-vertices ~~and~~ then it have $nc_2 = \frac{n(n-1)}{2}$ edges. Then it is complete.

some examples are

(a)



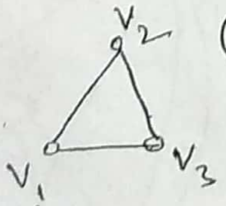
$n=2$

$|V|=2$

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

$= 1$

(b)

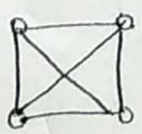


$n=3$

$|V|=3$

$|E|=3$

(c)

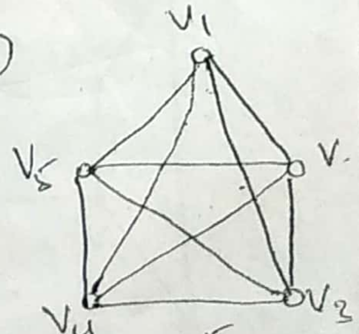


$n=4$

$|V|=4$

$|E|=6$

(d)



$n=5$

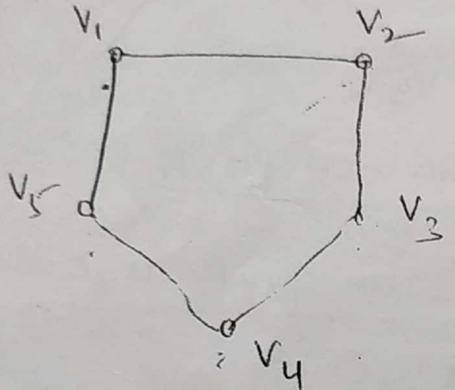
$|V|=5$

$|E|=10$

Complement of a Graph: Let 'G' be a Graph said to be complement if the ~~Adjacent~~ Vertices of 'G' ~~Adjacent, those vertices are~~ not AdJacent in the Graph. ie, $\overline{G}$.

ie, $V(\overline{G}) = V(G)$ but $E(\overline{G}) = \{(x,y) : (x,y) \notin E(G)\}$

Ex:1 A Graph 'G' is below? find its complement?



Sol: V_1 is incident (or) AdJacent to V_2 and V_5.

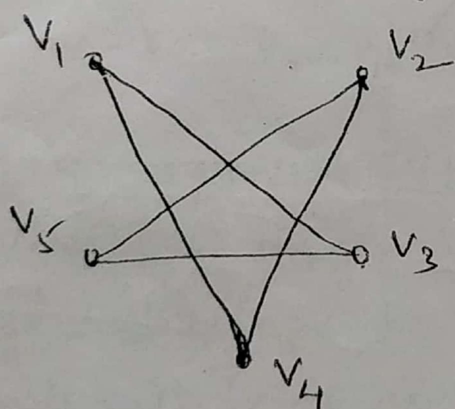
V_2 is AdJacent to V_1 and V_3

V_3 is AdJacent to V_2 and V_4

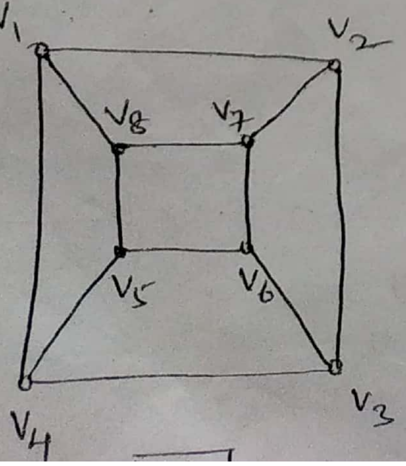
V_4 is AdJacent to V_3 and V_5.

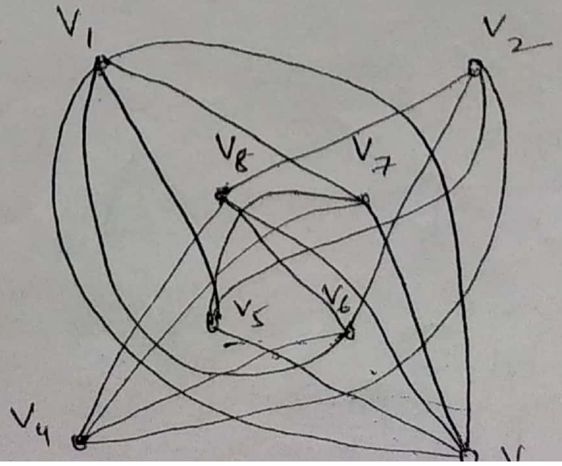
V_5 is AdJacent to V_1 and V_4.

So, the complement of a Graph is.



Ex:2 find the complement of the graph?





$\boxed{\overline{G}}$

Scanned by CamScanner

1) Adjacency Matrix: Let 'G' be a graph with n-

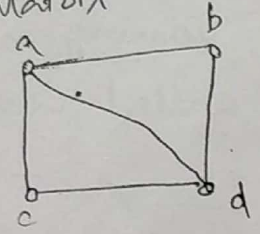
The adjacency Matrix is defined by. * It can be de by A_G.

$$A_G = [a_{ij}]_{n \times n}$$

if $a_{ij} = 1$ if v_i, v_j are adjacent.

$= 0$ if v_i, v_j are not adjacent.

Ex: 1 consider the graph. Represent the graph with Adjacency Matrix



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

Ex: 2 without constructing graph. P.T the graph whose adjacency Matrix is given by $X = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ is connected.

Def: If 'G' is connected means ~~atleast one~~ all entry in the Matrix $Y = X + X^2 + X^3 + \cdots X^{n-1}$ is a non-zero. Then the graph is connected.

Sol: $X = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

The order of $X = 3$, we find X^2.

$$X^2 = X * X = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0+1+0 & 0+0+0 & 0+0+0 \\ 0+0+0 & 1+0+0 & 0+0+0 \\ 0+0+0 & 0+0+0 & 0+0+0 \end{bmatrix}$$

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

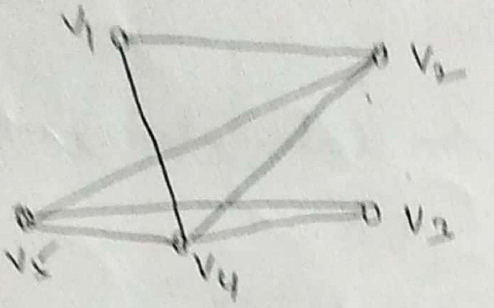
$$Y = X + X^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Some Entries in Y is equal to zero. So, its not Conn

Ex: 2 Draw the graphs for which the following is the Adjacency Matrix:

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

Sol:

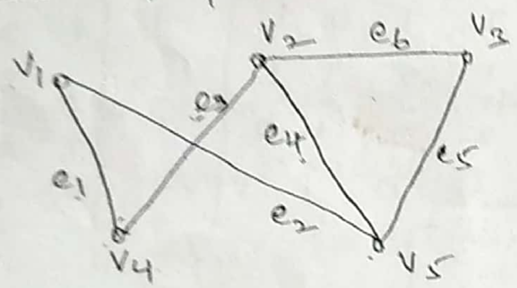


2) **Incidence Matrix:** Let 'G' be a undirected graph with n-vertices and n-edges. Then the incident Matrice is obtained as

$a_{ij} = 1$ if e_j is incident to v_i
$= 0$ if not.

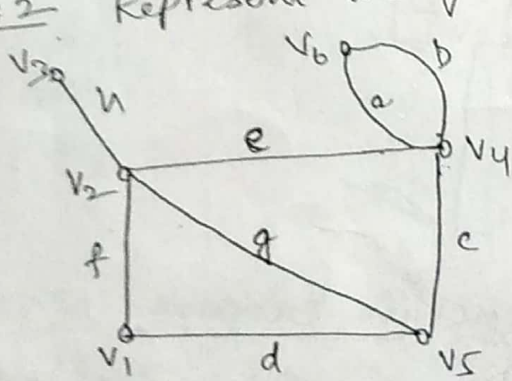
* It is denoted by I_G

Ex: 1 Represent the graph with an incident Matrix.



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

Ex: 2 Represent the graph with an Incidence Matrix.



$$I_G = \begin{array}{c|cccccccc} & a & b & c & d & e & f & g & h \\ \hline v_1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ v_2 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ v_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ v_4 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ v_5 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ v_6 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

***
Ex: A graph has the following Adjacency. check whether it is connected (or) not. [JNTU 200

$$A(G) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

order = 5.

$B = A + \cancel{A^2} A^3 + A^4.$

$$B = \begin{bmatrix} 3 & 1 & 3 & 1 & 4 \\ 1 & 3 & 1 & 3 & 4 \\ 3 & 1 & 7 & 5 & 4 \\ 1 & 3 & 5 & 7 & 4 \\ 4 & 4 & 4 & 4 & 8 \end{bmatrix}$$

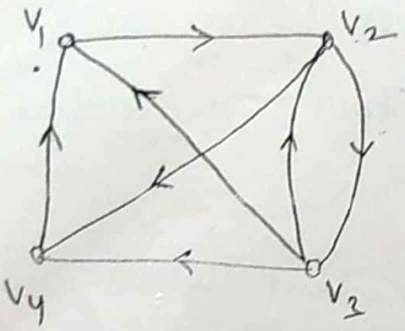
It is not

3) **Path Matrix :-** Let 'G' be a simple diagraph with ...
then the path Matrix is obtained as

$$P_{ij} = \begin{cases} 1 & \text{if elements of } i^{th} \text{ row \& } j^{th} \text{ coloumn of } B_n \ldots \\ 0 & \text{if not.} \end{cases}$$

*NOTE: The path Matrix can be calculated from t[he] Matrix $P_G = A + A^2 + A^3 + \ldots + A^n$.

Ex:1 Consider the graph. Find path Matrix.



$$A = \begin{array}{c c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}_{4\times 4} \end{array}.$$

To Find P_G

$$P_G = A + A^2 + A^3 \cancel{+A^4} \quad \text{B'cos order} = 4.$$

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

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

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

Another Method: Transitive closure

$A = [\ ]$

$A^2 = A \cap A$

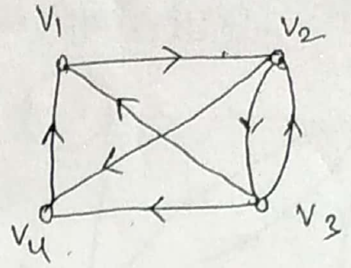
$A^3 = A^2 \cap A$

$A^4 = A^3 \cap A$

$$P_{ij} = A \cup A^2 \cup A^3 \cup A^4$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} = A^+ \text{ Same as Path Matrix}$$

Ex: Consider the graph. Find Path Matrix by using Warshall's Algorithm.



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

Assign $M_0 = A$.

$$M_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Step 1: Consider 1^{st} row & 1^{st} coloumn elements of M_0 which are equal to 1.

$M_0(V_1, V_2) = M_0(V_3, V_1) = M_0(V_4, V_1) = 1$

From the above ordered pairs form a new ordered pair

We get,

$M_1(V_3, V_2) = M_1(V_4, V_2) = 1$

$$M_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}.$$

Step 2: Consider 2^{nd} row & 2^{nd} coloumn elements of M_1 which are equal to 1.

$M_1(V_1, V_2) = M_1(V_3, V_2) = M_1(V_4, V_2) = M_1(V_2, V_3) = M_1(V_2, V_4) = 1$

We get,

$M_2(V_1, V_3) = M_2(V_1, V_4) = M_2(V_3, V_3) = M_2(V_3, V_4) = M_2(V_4, V_3)$

$M_2(V_4, V_4) = M_2(V_2, V_2) = 1$

$$M_2 = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 3: Consider 3^{rd} row & 3^{rd} coloumn elements equal to 1

$M_2(V_1, V_3) = M_2(V_2, V_3) = M_2(V_3, V_3) = M_2(V_4, V_3) = M_2(V_3, V_1)$

$= M_2(V_3, V_2) = M_2(V_3, V_4) = 1$

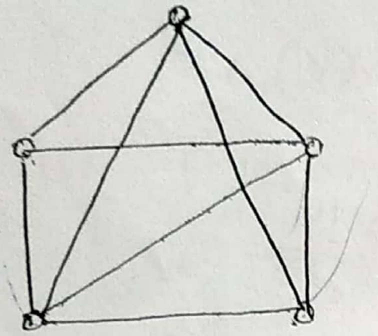
We get $M_3(V_1, V_3) = M_3(V_1, V_1) = M_3(V_1, V_2) = M_3(V_1, V_4) = M_3(V_2, V_1) = 1$

$$M_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \text{Path Matrix of } A.$$

Scanned by CamScanner

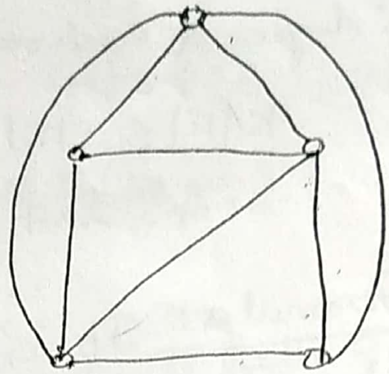
Planar Graphs: A Graph 'G' is said to be planar if it doesn't have any edge crossing, otherwise it is Non-planar Graph.

Ex: 1 Let the Graph 'G' is



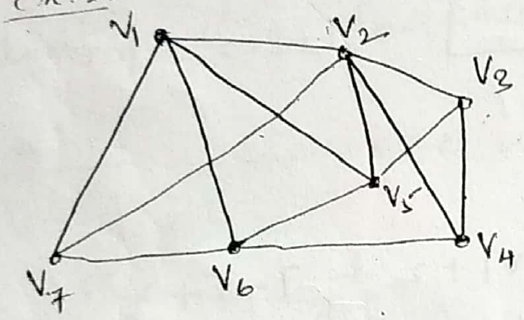
Graph 'G'.

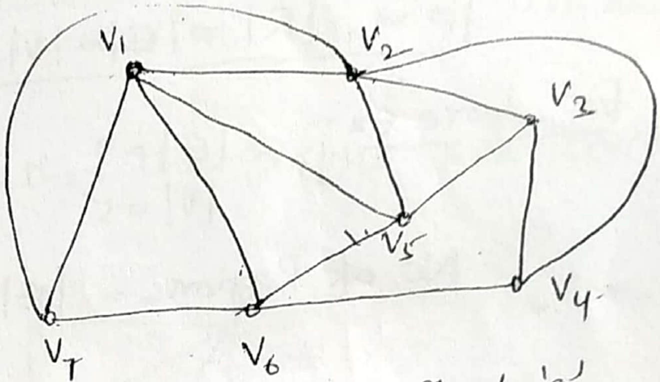
$\Rightarrow$



Planar Graph of 'G'.

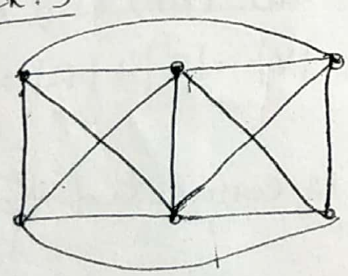
Ex: 2





Planar Graph of 'G'.

Ex: 3



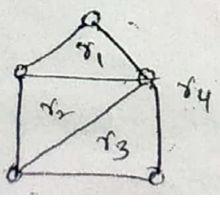
It is a Non-planar Graph b'cos it cannot drawn without crossing of edges.

NOTE: Planar Graph is also called as Plane Graph

A Plane Graph divides the plane in to Regions.

Region: A region is a cycle that forms its boundary.

Ex:



A Graph has Four Regions.

r_1, r_2, r_3 and r_4.

r_1, r_2, r_3 are Interior regions.

Result: In a plane graph 'G', if the degree of each is K then $K|R| \le 2|E|$.

for above example.

$|R|$ = no. of Regions = 4

$|E|$ = no. of edges = 7

K = degree of each region = 3

$$K|R| \le 2|E|$$

$$3 \times 4 \le 2 \times 7 \Rightarrow 12 \le 14$$

Euler's Formula: If 'G' is a connected Planar Graph, then it forms $|E| - |V| + 2$ regions. (Including Exterior)

$$\boxed{\therefore\ |R| = |E| - |V| + 2} \rightarrow \text{Euler's Formula.}$$

For above ex: $|E| = 7$
$|V| = 5$

No. of Regions $= |E| - |V| + 2 = 7 - 5 + 2$
$= 4$ regions.

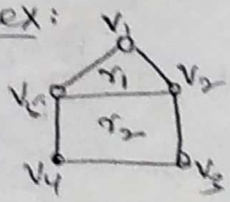
Theorm: Prove that Euler's formula (or) If 'G' is a Connected plane graph then P.T $|V| - |E| + |R| = 2$.

proof:

Assume that suppose 'G' is connected plane graph having $(K+1)$ Regions.

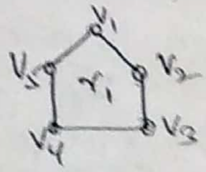
Now Delete an edge which is common to both Regions.

For ex:



Here the common edge for both regions r_1 and r_2 is $(v_2 - v_5)$. Delete it.

We get,



In the resulting graph, we have same no of vertices and lack of one edge & one region.

Isomorphism on Graphs :-

Let G_1 and G_2 are two graph, then $f: G_1 \to G_2$ is called an isomorphism if

(i) 'f' is one to one [no.of vertices in G_1 = no.of vertices G_2]

ie, $G_1(V) = G_2(V)$ (or) $V(G_1) = V(G_2)$

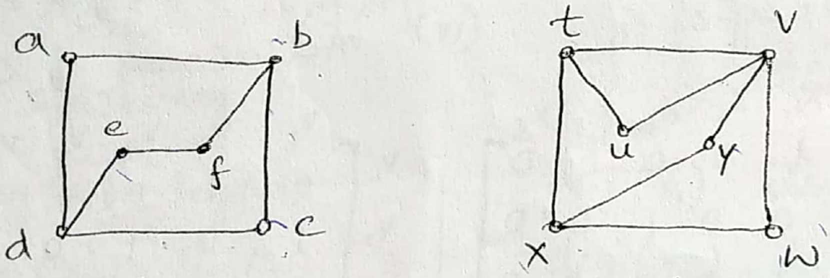
(ii) 'f' is on to [no.of Edges in G_1 = no.of Edges in G_2

ie, $E(G_1) = E(G_2)$

(iii) The degree sequence of both graphs G_1 and G_2 are same.

(iv) The ~~Ad~~ Jacency Matrix of G_1 & G_2 are same

Ex: 1 Determine whether the graphs are isomorf (or) Not.



'G_1' 'G_2'

Sol:

Graph G_1

(i) $|V| = 6$

(ii) $|E| = 7$

Graph G_2.

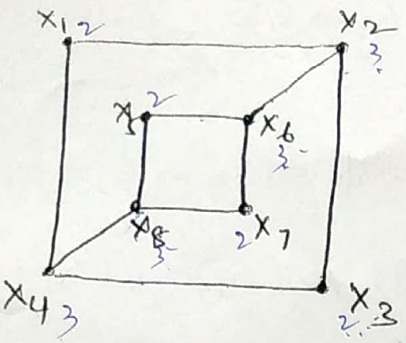
(i) $|V| = 6$

(ii) $|E| = 8$

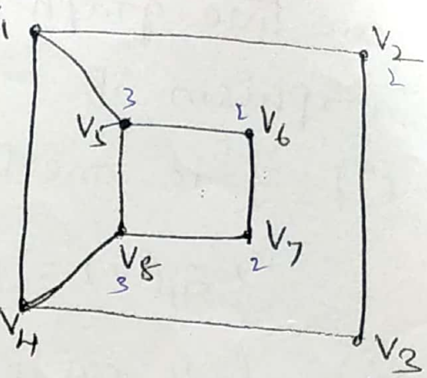
The no.of Edges in G_1 and no.of Edges in G_2 are not Equal. So, G_1 is not isomorphic to G_2

$\therefore$ They are not isomorphic.

Ex: 2 Determine whether the given Graphs are isomorphic (or) Not.



(i)



(ii)

Sol:

Graph 'G_1'

(i) $|V| = 8$

(ii) $|E| = 10$

(iii) Degree sequence of G_1
$= \{2, 2, 2, 2, 3, 3, 3, 3\}$

(iv) Adjacency of G_1

$$
\begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{array}
\begin{array}{c}
\begin{array}{cccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

Graph 'G_2'

(i) $|V| = 8$

(ii) $|E| = 10$

(iii) Degree sequence of G_2
$= \{2, 2, 2, 2, 3, 3, 3, 3\}$

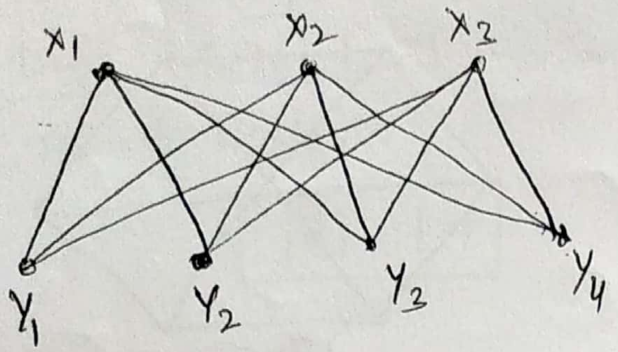
(iv) Adjacency of G_2.

$$
\begin{array}{c} \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \end{array}
\begin{array}{c}
\begin{array}{cccccccc} V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 & V_8 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

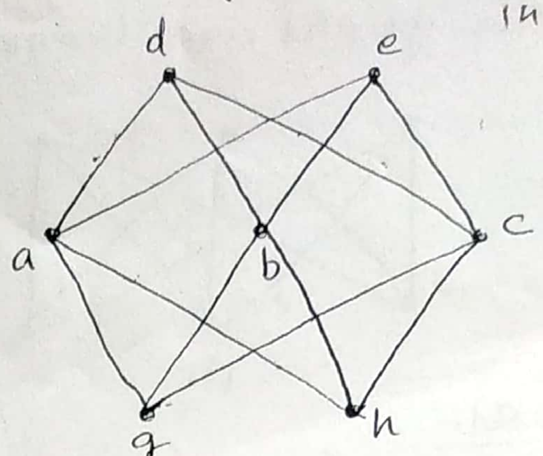
The AdJancey of Graphs G_1 & G_2 are not Equal.

$\therefore$ They are Not isomorphic.

EX:3 Determine whether the given graphs are isomorphic (or) Not.



'G_1'



'G_2'

Sol:

Graph 'G_1'

(i) $|V| = 7$

(ii) $|E| = 12$

(iii) Degree sequence is $= \{3,3,3,3,4,4,4\}$

(iv) Adjency

$\deg(x_1) = \deg(a)$
$\deg(x_2) = \deg(b)$
$\deg(x_3) = \deg(c)$

Graph 'G_2'

(i) $|V| = 7$

(ii) $|E| = 12$

(iii) Degree sequence is $= \{3,3,3,3,4,4,4\}$

(iv) Adjacency.

$\deg(Y_1) = \deg(d)$
$\deg(Y_2) = \deg(e)$
$\deg(Y_3) = \deg(g)$
$\deg(Y_4) = \deg(h)$

So, The Adjacency of two Graphs G_1 and G_2 are same.

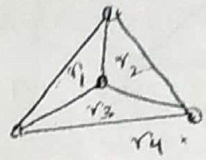
$\therefore$ They are isomorphic

Theorm: Show that if a plane graph is self dual, then $|E| = 2|V| - 2$.

Def: Self dual means if no. of vertices is equal to no. of Regions.

$$\boxed{|R| = |V|}$$

Ex:



Proof:

From Euler's formula, we have

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

But 'G' is self dual, ie $|R| = |V|$

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

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

$$2|V| - |E| = 2$$

$$\boxed{\therefore \quad |E| = 2|V| - 2}$$

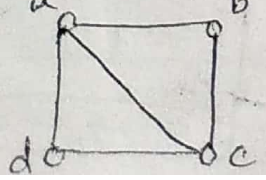
$\therefore$ Hence Theorm proved.

Eulerian Graphs:-

Euler Path: A path in the Graph 'G' is called an Euler path if it visit every edge exactly once and each vertex of the graph is atleast once. Then it is the Euler path of the graph.

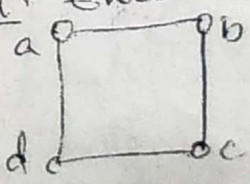
Euler circuit: The Euler Path starts at one point and ends at the same. Then the graph is Euler circuit.

Ex:



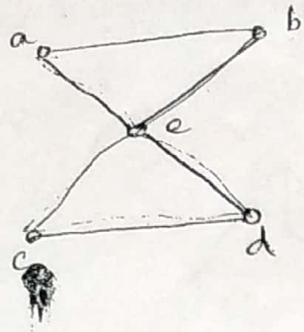
Euler Path is $a-b-c-d-a-c$.

Ex: Euler circuit



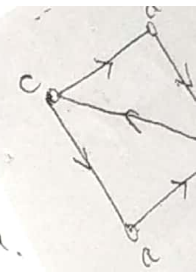
$a-b-c-d-a$

Ex: Consider the graph in the following figure.



This graph has an Euler circuit of the path a - b - e - c - d - e - a.

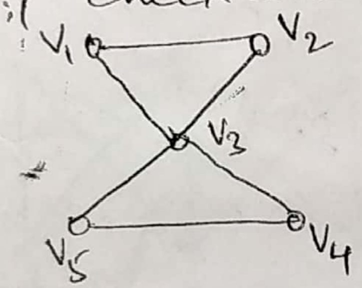
* Starting & End points are identical
So, it is an Euler circuit.



NOTE: A Directed Multigraph 'G' has an Euler circuit iff the indegree of each vertex is equal to outdegree of that vertex.

The indegree of one vertex is one larger than its outdegree & indegree of one vertex is one less than its outdegree. Then it has an Euler path. but it doesnot have euler circuit.

Ex: 1 check whether the graph is Eulerian [JNTU]



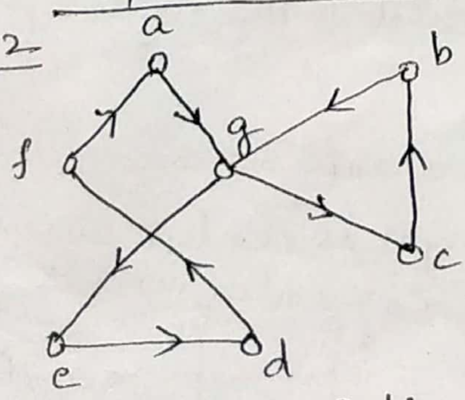
Vertex	Indegree	outdegree
V_1	2	2
V_2	2	2
V_3	4	4
V_4	2	2
V_5	2	2

So, the indegree & outdegree of each vertex is same.

So, it is an Euler circuit of path $V_1 - V_2 - V_3 - V_5 - V_4 - V_3 - V_1$.

So, it is Eulerian

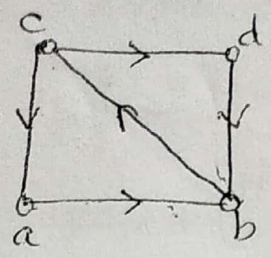
Ex: 2



V(G)	Indegree	outdegree
a	1	1
b	1	1
c	1	1
d	1	1
e	1	1
f	1	1
g	2	2

Euler circuit of path
a - g - c - b - g - e - d - f - a.

... whether the graph is Eulerian (or) Not.



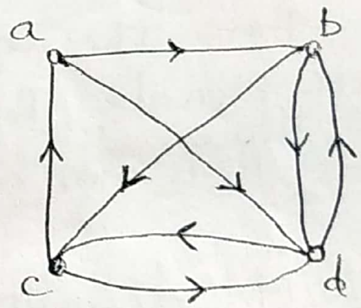
V(G)	Indegree	outdegree
a	1	1
b	2	1
c	1	2
d	1	1

Here the indegree of Vertex 'b' is one greater the outdegree. and indegree of vertex 'c' is one less than the outdegree.

It is not an Euler circuit but the Euler path is

c – a – b – c – d – b.

Ex:4 consider the directed graph. check whether it is Eulerian (or) Not:



Sol:

V(G)	Indegree	out Degree
a	1	2
b	2	2
c	2	2
d	3	2

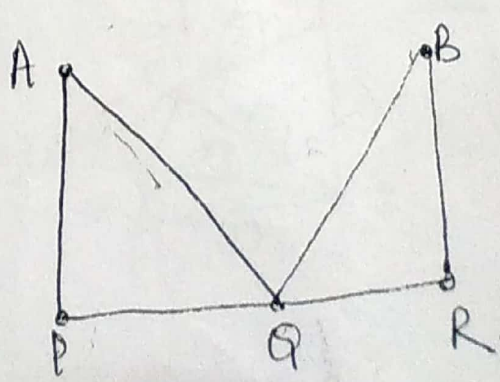
Hence, the indegree & outdegree of the vertices are not equal. So, it doesn't form an Euler's Circuit. But the Euler's path is a-b-d-b-c-d-c-a-d.

Ex-5: S.T the following graph is Eulerin.

[JNTU June 2003]

Sol:

V(G)	Indegree	outdegree
A	2	2
B	2	2
P	2	2
Q	4	4
R	2	2



The indegree & outdegree of all the Vertices are ...

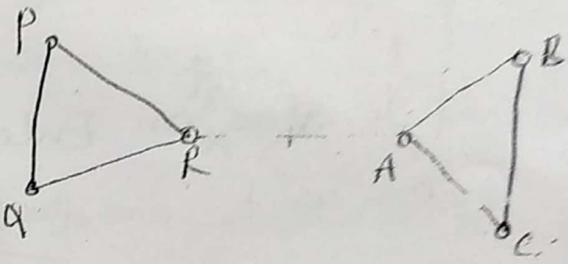
So, it is an Euler's circuit (or) Eulerian.

The Euler path of the given graph is

A – P – Q – R – B – Q – A.

Ex:6: Verify the following graph is not Eulerian? [JNTU June 2003]

V(G)	Indegree	outdegree
P	2	2
Q	2	2
R	3	3
A	3	3
B	2	2
C	2	2



Here the indegree & outdegree of all the vertices are Equal. But it doesn't have an Eulers path. So, it is not an Euler circuit (or) Eulerian.

Theorm: The number of Vertices of odd degree in a graph is always even.

Proof: we consider the vertices with odd degree & Even degree separately.

We know that $\sum_{i=1}^{n} deg(v_i) = 2 \cdot |E|$

where $|E|$ is number of Edges.

v_i is vertices from $1, 2, 3 \ldots n$.

$$\sum_{i=1}^{n} deg(v_i) = \sum_{i=Even} deg(v_j) + \sum_{i=odd} deg(v_k)$$

$$2 \cdot E = \underbrace{\sum_{Even} deg(v_j)} + \sum_{odd} deg(v_k)$$

This Expression is always Even [sum of Even numbers]

$$\therefore \sum_{odd} deg(v_k) = 2 \cdot E - \sum_{even} deg(v_j)$$

RHS is always Even.

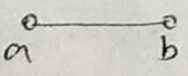
So, No. of vertices of odd degree = Even.

Hence Theorm proved.

Theorm:- Every Tree with 2 (or) More vertices is 2-chromatic.

Proof: Tree is a Acyclic graph. It does not have any Cycles.

Ex: Consider two vertices



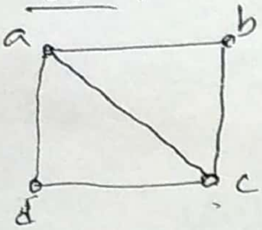
It is a Tree.

→ Chromatic means min. no. of colors required to coloring the vertices of

Hamilton graph: Let 'G' be the Graph is said to be Hamilton graph if it Contains Hamilton ~~path~~ cycle.

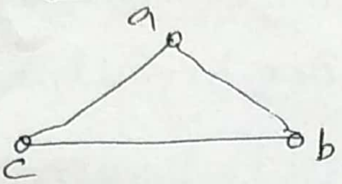
Hamilton cycle is nothing but, it is a path which Covers all the vertices of a Graph 'G' and forms a cycle. (It may (or) maynot Covers all the Edges of the Graph 'G').

Ex: 1



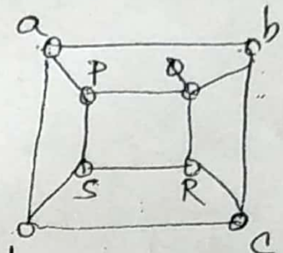
a-b-c-d-a → Cycle.
So, 'G' is Hamilton Graph.

Ex: 2

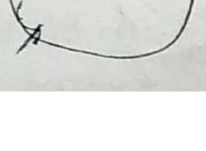


a-b-c → Cycle
So, Hamilton Graph

Ex: 3



a-b-c-d-S-R-Q-P-a is a cycle.
So, it is Hamilton graph



Chromatic Numbers:-

The chromatic Number of a graph 'G' is the minimum number of colors to color the vertices of the Graph 'G'.

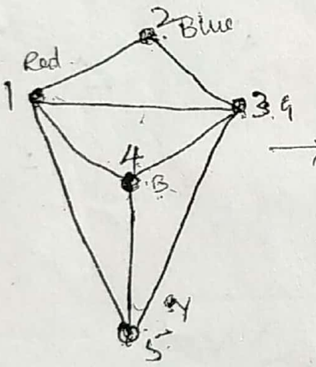
Chromatic number of the graph is denoted by χ $\chi(G)$.

If $\chi(G) = K$ then it is K-chromatic graph.

If $\chi(G) = 2$ then it is 2-chromatic.

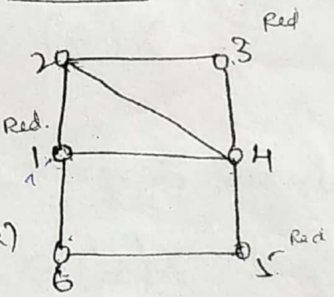
Vertex Coloring: The assignment (assigning) of colors to the vertex is called vertex colouring, ~~But~~ that is one color to the each vertex, such that Adjacent vertices are assigned by different colours.

Ex: 1 What is the chromatic Number of given graph.



$\rightarrow \chi(G) = 4$

Ex: 2



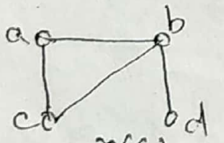
1, 3, 5 → same colour

2, 6 → same color

4 → one col

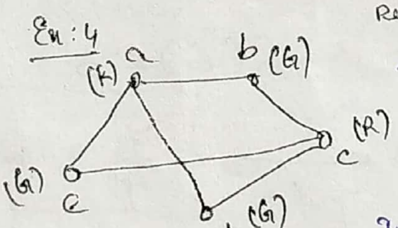
2, 6 $\chi(G) = 3$

Ex: 3



$\chi(G) = 3$

Ex: 4



$\chi(G) = 2$

NOTE: It is very difficult to find the chromatic number of the graph 'G', if 'G' has more no. of vertices.

So, the following rules are helpful to find the chromatic number of the graph.

(i) $\chi(G) \leq |V|$, where $|V|$ is the number of vertic of 'G'.

(ii) A Triangle requires 3 colours. Ily, A complet graph with n-vertices requires n-colours.

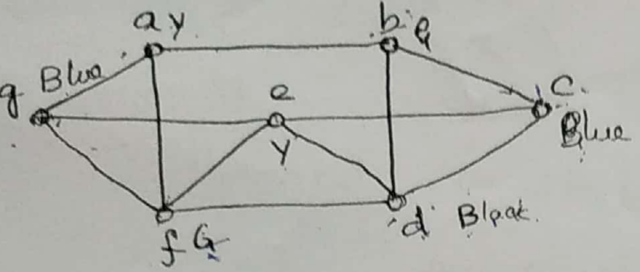
(3) ... then atmost 'd' colors are required to colour the ve...

iii) For any graph G, $\chi(G) \le 1+\Delta(G)$, where $\Delta(G)$ is the largest degree of the vertex of 'G'.

iv) For any graph G, $\chi(G) \geq \frac{|V|}{|V|-\delta(G)}$, where $\delta(G)$ is the minimum degree of the vertex of 'G'.

v) Degree of all vertices are same 'n', it has atmost n - colors.

Ex: 3 Find the chromatic number of the following graph.



$\deg(a) = \deg(b) = \deg(c) = \deg(g) = 3$

$\deg(e) = \deg(f) = \deg(d) = 4.$

The maximum degree of the Vertex of G is

$$\Delta(G) = 4.$$

By rule (3)

$$\chi(G) \le 1+\Delta(G)$$

$$\chi(G) \le 5 \checkmark$$

By rule (4)

$$\chi(G) \geq \frac{|V|}{|V|-\delta(G)}$$

minimum degree is $\delta(G) = 3$.

$$\chi(G) \geq \frac{7}{7-3} = \frac{7}{4} = 1.7$$

$$\chi(G) \geq 2.$$

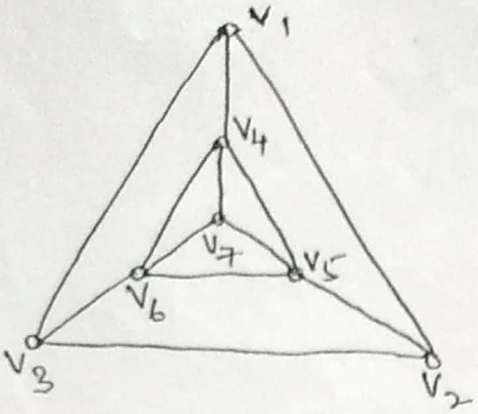
$$2 \le \chi(G) \le 5.$$

a, e → same colour.
b, f → same colour
g, c → same colour.
d → one colour.

2- coloring is not Possible.
3 - coloring is not Possible.
5- coloring is also not possible.

so, $\boxed{\chi(G) = 4}$ It is 4- chromatic

EX: 4 Consider the graph. Find the chromatic number.



$\deg(v_1) = \deg(v_2) = \deg(v_3) = \deg(v_7) = 3$

$\deg(v_4) = \deg(v_5) = \deg(v_6) = 4$

The Maximum degree $\Delta(G) = 4$.

$\chi(G) \leq 5$.

The minimum degree is $\delta(G) = 3$

$\chi(G) = \frac{7}{7-3}$

$\chi(G) \geq 2$

$2 \leq \chi(G) \leq 5$.

$v_1, v_7 \rightarrow$ same colour

$v_3, v_4 \rightarrow$ same colour

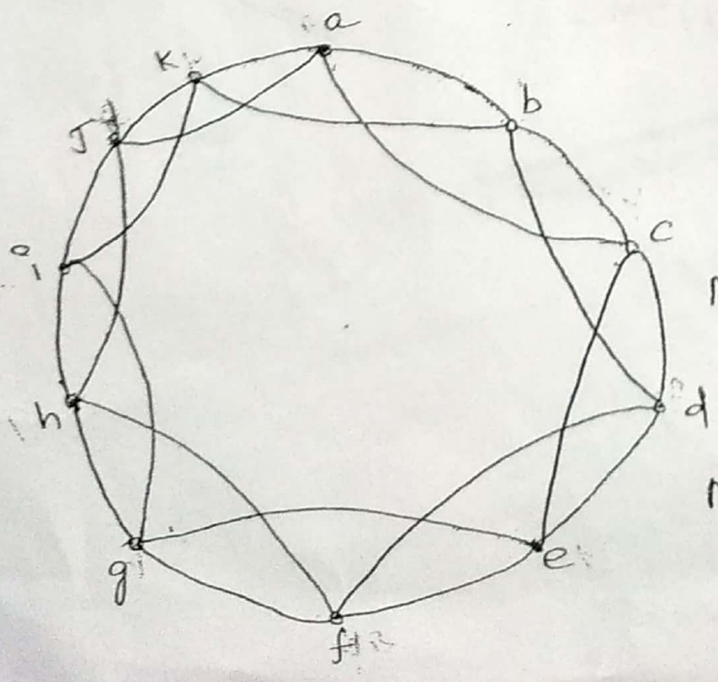
$v_2, v_6 \rightarrow$ same colour

$v_5 \rightarrow$ one colour.

2-coloring, 3-coloring, 5-colouring is not possible.

So, $\boxed{\chi(G) = 4}$

EX: 5 Find the chromatic number of the graph.



Degree of each vertex

$\deg(v) = 4$.

max. degree

$\Delta(G) = 4$

$\chi(G) \leq 5$

min degree $\delta(G) = 4$

$\chi(G) \geq \frac{11}{11-4}$

$\chi(G) \geq 3$

Scanned by CamScanner

Degree of all vertices are same.

ie, $deg(v) = 4$.

$$3 \le \chi(G) \le 4.$$

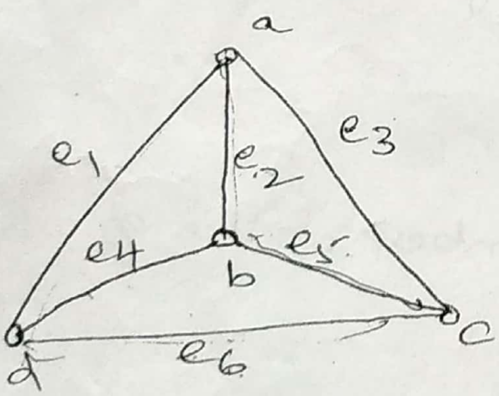
3 - coloring is not possible.

so $\boxed{\chi(G) = 4}$

Edge chromatic Number:-

The Edge chromatic Number of 'G' is the minimum number of colors required to color the all edges of 'G', But the edges with common end points are colored different colors.

Ex: What is the edge chromatic Number of the following graph.



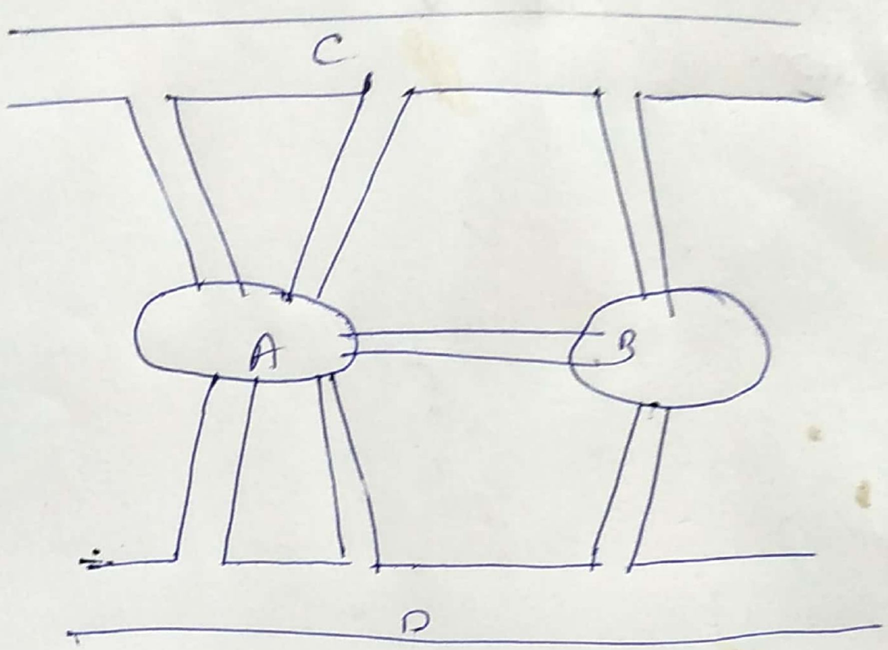
$e_1, e_5 \rightarrow$ same colour

$e_2, e_6 \rightarrow$ same colour

$e_3, e_4 \rightarrow$ " "

Konigsberg Bridge problem:

There are two islands A & B formed by a river. They are connected to each other and to the river banks C & D by means of 7- bridges.

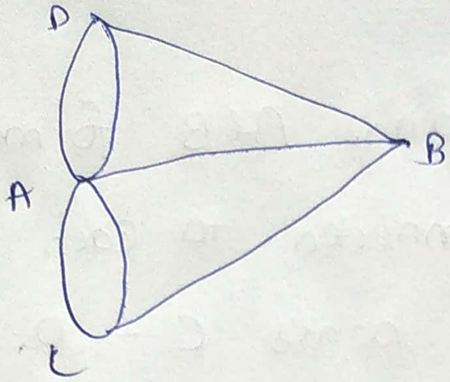


The problem is to start from any one of the 4 land areas A, B, C, D walk across each bridge exactly once and return to the starting point.

This problem has no solution.

when the situation is represented by a Graph with vertices representing the lands

~~Here~~



Here we can't find a Eulerian Circuit.

Hence Konigsberg Bridge problem has no solution.

Graph Traversal

Graph traversal is the problem of visiting all the vertices of a graph in some systematic order. There are mainly two ways to traverse a graph.

- Breadth First Search
- Depth First Search

Breadth First Search

Breadth First Search (BFS) starts at starting level-0 vertex X of the graph G. Then we visit all the vertices that are the neighbors of X. After visiting, we mark the vertices as "visited," and place them into level-1. Then we start from the level-1 vertices and apply the same method on every level-1 vertex and so on. The BFS traversal terminates when every vertex of the graph has been visited.

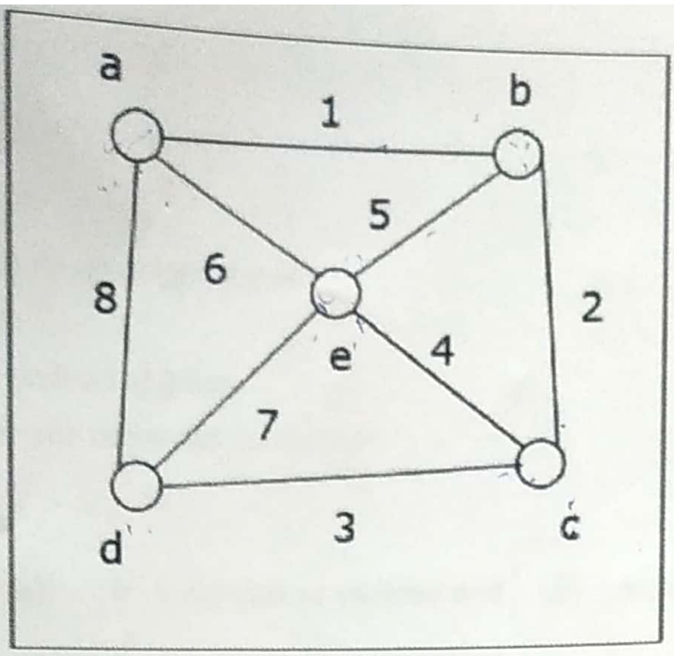
BFS Algorithm

The concept is to visit all the neighbor vertices before visiting other neighbor vertices of neighbor vertices.

- Initialize status of all nodes as "Ready".
- Put source vertex in a queue and change its status to "Waiting".
- Repeat the following two steps until queue is empty –
  - Remove the first vertex from the queue and mark it as "Visited".
  - Add to the rear of queue all neighbors of the removed vertex whose status is "Ready". Mark their status as "Waiting".

Problem

Let us take a graph (Source vertex is 'a') and apply the BFS algorithm to find out the traversal order.



Solution –

- Initialize status of all vertices to "Ready".
- Put *a* in queue and change its status to "Waiting".
- Remove *a* from queue, mark it as "Visited".
- Add *a*'s neighbors in "Ready" state *b*, *d* and *e* to end of queue and mark them as "Waiting".
- Remove *b* from queue, mark it as "Visited", put its "Ready" neighbor *c* at end of queue and mark *c* as "Waiting".
- Remove *d* from queue and mark it as "Visited". It has no neighbor in "Ready" state.
- Remove *e* from queue and mark it as "Visited". It has no neighbor in "Ready" state.
- Remove *c* from queue and mark it as "Visited". It has no neighbor in "Ready" state.
- Queue is empty so stop.

So the traversal order is –

$$\boxed{a \to b \to d \to e \to c}$$

The alternate orders of traversal are –

$$\boxed{a \to b \to e \to d \to c}$$

Or, $\boxed{a \to d \to b \to e \to c}$

Or, $\boxed{a \to e \to b \to d \to c}$

Or, $\boxed{a \to b \to e \to d \to c}$

$a \to d \to e \to b \to c$

Application of BFS

- Finding the shortest path
- Minimum spanning tree for un-weighted graph
- GPS navigation system
- Detecting cycles in an undirected graph
- Finding all nodes within one connected component

Complexity Analysis

Let $G(V, E)$ be a graph with $|V|$ number of vertices and $|E|$ number of edges. If breadth first search algorithm visits every vertex in the graph and checks every edge, then its time complexity would be −

$$O(|V| + |E|). O(|E|)$$

It may vary between $O(1)$ and $O(|V2|)$

Depth First Search

Depth First Search (DFS) algorithm starts from a vertex v, then it traverses to its adjacent vertex (say x) that has not been visited before and mark as "visited" and goes on with the adjacent vertex of x and so on.

If at any vertex, it encounters that all the adjacent vertices are visited, then it backtracks until it finds the first vertex having an adjacent vertex that has not been traversed before. Then, it traverses that vertex, continues with its adjacent vertices until it traverses all visited vertices and has to backtrack again. In this way, it will traverse all the vertices reachable from the initial vertex v.

DFS Algorithm

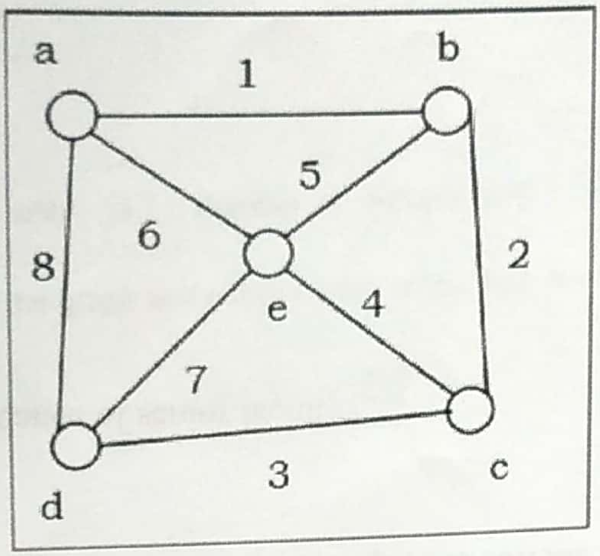
The concept is to visit all the neighbor vertices of a neighbor vertex before visiting the other neighbor vertices.

- Initialize status of all nodes as "Ready"
- Put source vertex in a stack and change its status to "Waiting"
- Repeat the following two steps until stack is empty −
    - Pop the top vertex from the stack and mark it as "Visited"

- Push onto the top of the stack all neighbors of the removed vertex whose status is "Ready". Mark their status as "Waiting".

em

us take a graph (Source vertex is 'a') and apply the DFS algorithm to find out the traversal order.



Solution

- Initialize status of all vertices to "Ready".
- Push *a* in stack and change its status to "Waiting".
- Pop *a* and mark it as "Visited".
- Push *a*'s neighbors in "Ready" state *e*, *d* and *b* to top of stack and mark them as "Waiting".
- Pop *b* from stack, mark it as "Visited", push its "Ready" neighbor *c* onto stack.
- Pop *c* from stack and mark it as "Visited". It has no "Ready" neighbor.
- Pop *d* from stack and mark it as "Visited". It has no "Ready" neighbor.
- Pop *e* from stack and mark it as "Visited". It has no "Ready" neighbor.
- Stack is empty. So stop.

So the traversal order is −

$$\boxed{a \to b \to c \to d \to e}$$

The alternate orders of traversal are −

$$\boxed{a \to e \to b \to c \to d}$$

Or, $\boxed{a \to b \to e \to c \to d}$

$$a \rightarrow d \rightarrow e \rightarrow b \rightarrow c$$

Or,

$$a \rightarrow d \rightarrow c \rightarrow e \rightarrow b$$

Or,

$$a \rightarrow d \rightarrow c \rightarrow b \rightarrow e$$

Complexity Analysis

Let $G(V, E)$ be a graph with $|V|$ number of vertices and $|E|$ number of edges. If DFS algorithm visits every vertex in the graph and checks every edge, then the time complexity is –

$$\Theta(|V| + |E|)$$

Applications

- Detecting cycle in a graph
- To find topological sorting
- To test if a graph is bipartite
- Finding connected components
- Finding the bridges of a graph
- Finding bi-connectivity in graphs
- Solving the Knight's Tour problem
- Solving puzzles with only one solution