Graph Theory Notes

ICT 6th semester

by

Mukesh Singh

ICT 5th batch 2073

Sukuna Multiple Campus

Sundarharaincha-12, Morang

प्रश्न नं. उत्तर

Unit-1

Fundamental Concepts of Graph Theory

Graph theory is a branch of mathematics and Computer Science, which is the study of graphs that concerns with the relationship between edges and vertices.

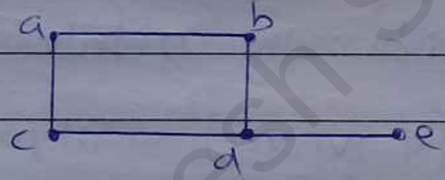
1.1. Introduction of Graph

A graph is a pictorial ~~form~~ and mathematical representation of a set of objects where some pairs of objects are connected by links.

Formally, a graph is an ordered pair $G = (V, E)$, where G specifies the graph.

V is set of vertices denoted by $V(G)$ or ($V = \{v_1, v_2, v_3,\}$)

E is the set of edges denoted by $E(G)$ or ($E = \{E_1, E_2, E_3\}$)



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

$E = \{ab, ac, bd, ed, de\}$

$\therefore G(5,5)$

Terms in graph

Point → Is a particular position in one/two of three-dimensional space. Represented by dot and labelled by alphabet or numbers.

Eg: • P

Line → It is connection or link between two points. It is represented by a solid line.

Example

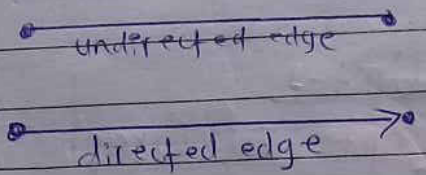
Vertex → It is a point where multiple lines meet. It is also called node, point or junction.

Example :-

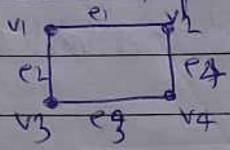
प्रश्न नं.

Edge → It is connection between two vertices. It is also called line or link or arc. It may be directed or undirected.

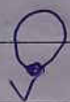
Examples



Graph → A graph G is defined as G = (V, E) where V is set of vertices and E is set of edges.



Loop → In a graph, if an edge is drawn from vertex to itself, it is called a loop.

Example: 

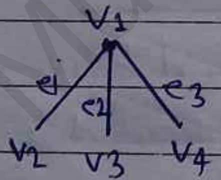
Edge (v, v)

1.2. Representation of Graph

Generally, we ~~draw~~ represent a graph by diagram and refer to the diagram itself as the graph.

We represent the graph pictorically as follows:

$v_1 \xleftrightarrow{e} v_2$ $[e = \{v_1, v_2\}]$ (2,1) graph

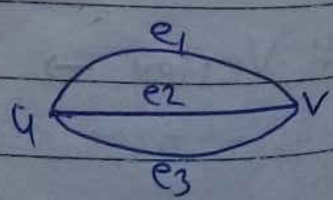


[(4,3) graph] $\cdot V_1$ & V_2 are adjacents, where V_2, V_3, V_4 are non-adjacents.

Where, starting point = ending point, then it is called loop.

(u, u) An edge joins vertex itself is called loop.

Parallel edges - Two or more edges (e_1, e_2, e_3) are said to be parallel edges, if they have same end points (u & v in figure).



Isolated vertex – ~~Two vertices are~~

Vertex with degree zero is isolated vertex.

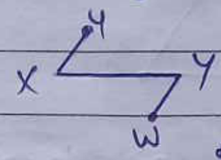
Isolated vertex has no edge incident on it.

Pendant Vertex – Vertex with degree one is pendant vertex.

There is an end point of only one edge.

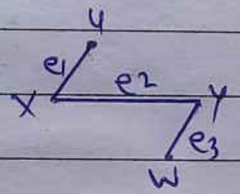
Example

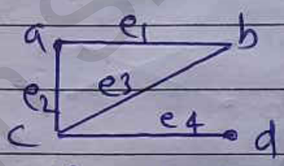
① 

② 

Pendant vertex :(u, w)

Pendant Edge – An edge having one of its vertices a pendant vertex.

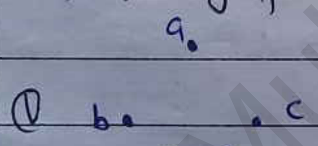
Example ① 

② 

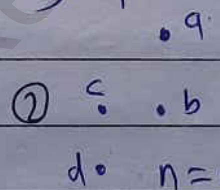
Pendant edge

1.3. Different types of graph

1) Null graph – A graph having no edge between its vertices.

① 

$n=3$

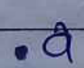
② 

$n=4$

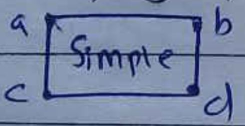
Where n = no. of vertices

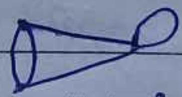
2) Trivial graph – A graph with only one vertex.

Example



3) Simple graph – A graph with no loops and no parallel edges.





Not simple

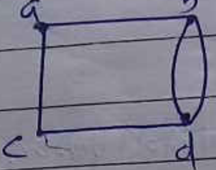
Maximum no. of edges possible,

$^nC_2 = n(n-1)/2$

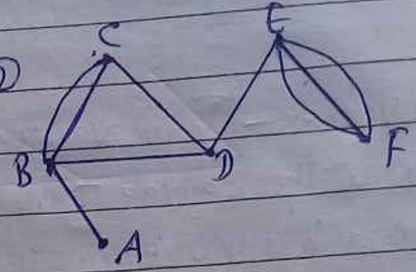
प्रश्न नं.

4) Multigraph ⇒ A graph having multiple edges between any pair of vertices.

Examples ①

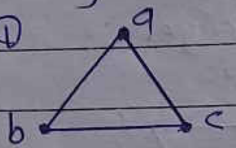


②

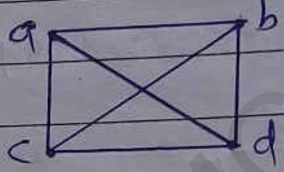


5) Complete graph → A graph in which every pair of vertices is joined by exactly one edge.

Examples ①



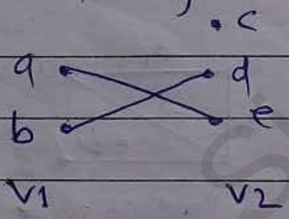
②



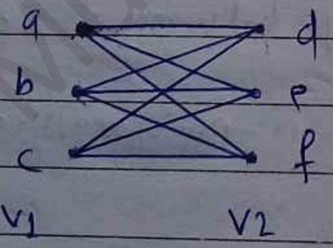
All vertices connects to all other vertices.

6) Bipartite graph → A simple graph ~~with~~ $G=(V,E)$ with vertex partition $V=\{V_1, V_2\}$ is bipartite graph if an edge connects any vertex in set V_1 to any vertex in set V_2.

Example

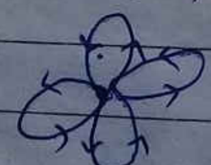


7) Complete Bipartite graph → A bipartite graph $G=(V,E)$ with vertex partition $V=\{V_1,V_2\}$ is complete bipartite graph if every vertex in set V_1 is connected to every vertex of V_2



8) Pseudo graph → A graph containing loops or multiple edges.

①

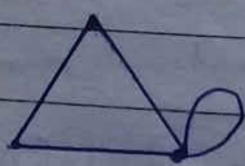


(G_1)

②

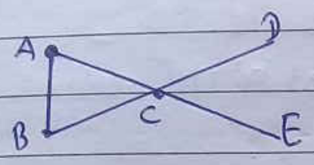
(G_2)

③

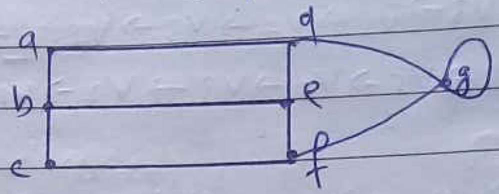


(G_3)

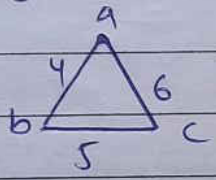
9) Labelled graph - If each vertex of a graph is given a unique name or label, it is called labelled graph.

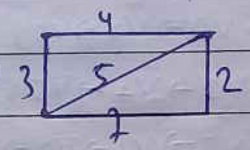
① 

(bn)

② 

10) Weighted graph - If each edge of a graph is assigned by a weight (numbers), it is called a weighted graph.

① 

② 

Order of graph = no. of vertices of the graph

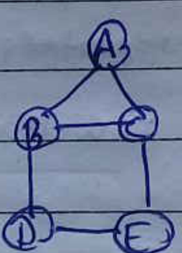
Size of graph = no. of edges of graph

1.4 Walks, paths and Cycles

1. Walk - It is sequence of vertices and edges of a graph. Total no. of edges covered in a walk is called length of the walk.
Vertex and edges can be repeated.

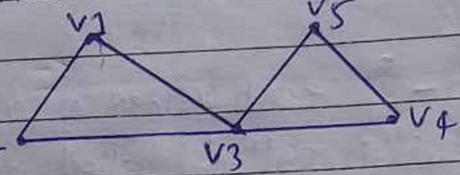
a) Open walk - If start vertex and end vertex are not same.

b) Close walk - If start vertex and end vertex are same.



- a → b → c → e → d (Length = 4)
- d → b → a → c → e → d → e → c (Length = 7)
- e → c → b → a → c → e → d (Length = 6)

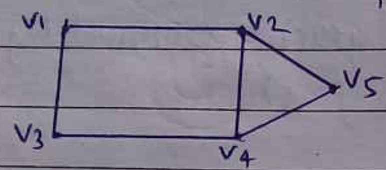
2.) Trial → An open walk in which edges cannot be repeated.



{V1 → V3 → V5 → V4 → V3 → V2} – trial

{V1 → V3 → V5 → V4 → V3 → V2 → V1} – Closed trial

3. Path → An open walk (or trial) in which all vertices are different, then the trial is called path. Edges are also not repeated.



Paths are

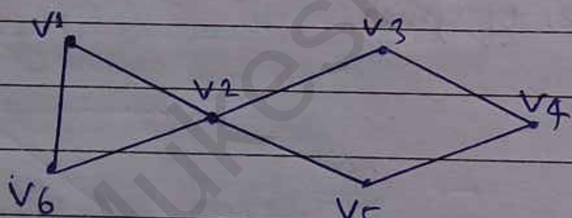
{V1 → V2 → V5}

{V1 → V2 → V4 → V5}

{V1 → V3 → V4 → V5}

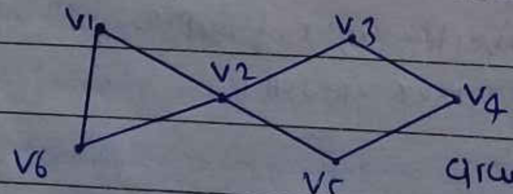
{V1 → V3 → V4 → V2 → V5}

4. Circuit → A closed trial/walk in which vertices may be repeated but edges are not repeated.



Here the closed trial, {V1 → V2 → V3 → V4 → V5 → V2 → V6 → V1} is a circuit.

5. Cycle → It is a circuit (or closed trial) in which neither vertices (except start & end) nor edges are repeated:



Here, Closed trial, {V1 → V2 → V3 → V4 → V5 → V2 → V6 → V1} is circuit but not cycle as V2 is repeated in middle.

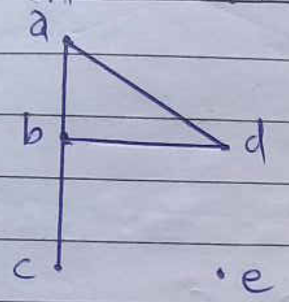
The closed trial, {V2 → V3 → V4 → V5 → V2} is a cycle as well as a circuit both.

प्रश्न नं. | उत्तर | प्राप्ताङ्क

1.5 ## The degree of a vertex

It is the number of edges connected in a vertex.
It is denoted by deg (v) or d(V).

In an undirected graph

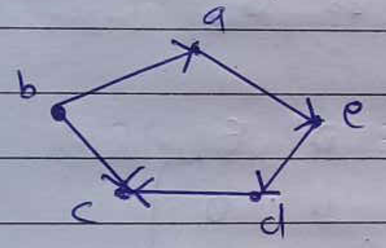


deg(a) = 2
deg(b) = 3
deg(c) = 1 - c is pendent vertex
deg(d) = 2
deg(e) = 0 - e is isolated vertex

$\therefore \sum deg(V) = 2+3+1+2+0 = 8$

In a directed graph

- Indegree of graph - no. of edges coming to the vertex. [$deg^-(v)$]
- Outdegree of graph - No. of edges going out from the vertex. [$deg^+(v)$]



Vertex	Indegree	Outdegree
a	1	1
b	0	2
c	2	0
d	1	1
e	1	1

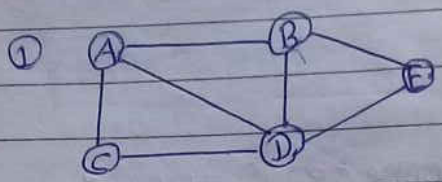
6. ## Matrix representation of graph

1. Adjacency Matrix System

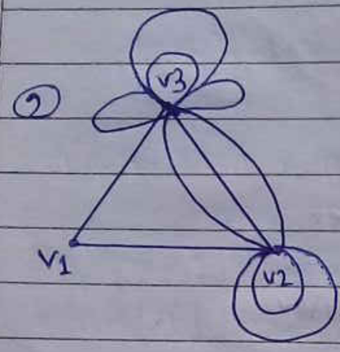
- It is a sequential representation.
- It is used to present which nodes are adjacent to each other.
- In a weighted graph we can store weight of edges instead of 1s & 0s.
- We have to construct a n×n matrix A. If any vertex i to vertex j, then $a_{ij} = 1$ else $a_{ij} = 0$.

Mukesh Singh

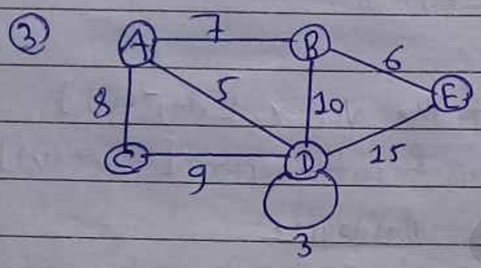
Undirected graph examples:

① 

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

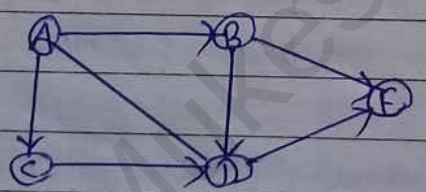
② 

$$A(G) = \begin{array}{c} \\ V_1 \\ V_2 \\ V_3 \end{array} \begin{array}{c} \begin{array}{ccc} V_1 & V_2 & V_3 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 4 \end{bmatrix} \end{array}$$

③ 

$$\rightarrow \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{ccccc} A & B & C & D & E \end{array} \\ \begin{bmatrix} 0 & 7 & 8 & 5 & 0 \\ 7 & 0 & 0 & 10 & 6 \\ 8 & 0 & 0 & 9 & 0 \\ 5 & 10 & 9 & 3 & 15 \\ 0 & 6 & 0 & 15 & 0 \end{bmatrix} \end{array}$$

Directed graph example



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

2. Incident Matrix System

→ In this system, graph can be represented by:

- Total no. of vertices by total no. of edges.
- It means if a graph has 4 vertices and 6 edges then we use a matrix of 4x6 class.

→ If there are $v_1, v_2 \ldots v_m$ vertices & $e_1, e_2 \ldots e_n$ edges in a graph G, then incident matrix of G denoted by $I(G) = (m_{ij})_{m \times n}$

परीक्षा मिति: .... मूल उ.पु. नं. : ....

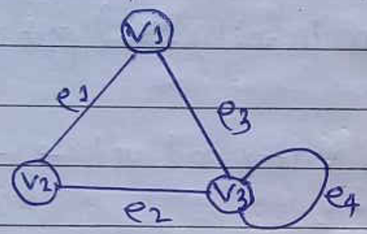
प्रश्न नं. | उत्तर | प्राप्ताङ्क

is defined as,

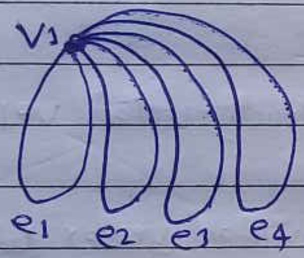
$$m_{ij} = \begin{cases} 0 \text{ if } v_i \text{ is not linked with } e_j \\ 1 \text{ if } v_i \text{ is linked with } e_j \\ 2 \text{ if } v_i \text{ has one loop } e_j \end{cases}$$

Here, m_{ij} denote the number of times as the vertex 'v_i' is linked with edge e_j.

Examples

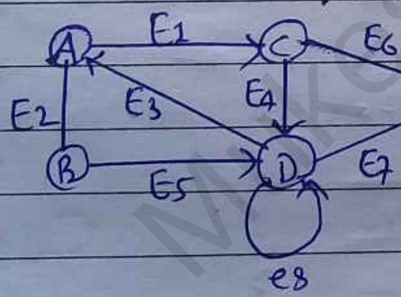
① 

$$\rightarrow I(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \end{array} \begin{array}{c} \begin{array}{cccc} e_1 & e_2 & e_3 & e_4 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 \end{bmatrix} \end{array}$$

② 

$$\Rightarrow I(G) = \begin{array}{c} \\ v_1 \end{array} \begin{array}{c} \begin{array}{cccc} e_1 & e_2 & e_3 & e_4 \end{array} \\ \begin{bmatrix} 2 & 2 & 2 & 2 \end{bmatrix} \end{array}$$

In case of directed graph:



$$\Rightarrow I(G) = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{cccccccc} E_1 & E_2 & E_3 & E_4 & E_5 & E_6 & E_7 & E_8 \end{array} \\ \begin{bmatrix} 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \end{bmatrix} \end{array}$$

1.7. The shortest path problem

It is about finding a path between two vertices in a graph such that the total sum of the edges weight is minimum.

Shortest Path using Dijkstra's Algorithm:

Step 1: Remove all parallel loops and edges if present.

Step 2: Start with a weighted graph.

Step 3: Choose starting vertex and assign infinity (∞) path values to all other vertices.

Step 4: Go to each vertex adjacent to this vertex and update its path length.

Step 5: If path length of adjacent vertex is lesser than new path length then don't update it.

Step 6: Avoid updating path length of already visited vertices.

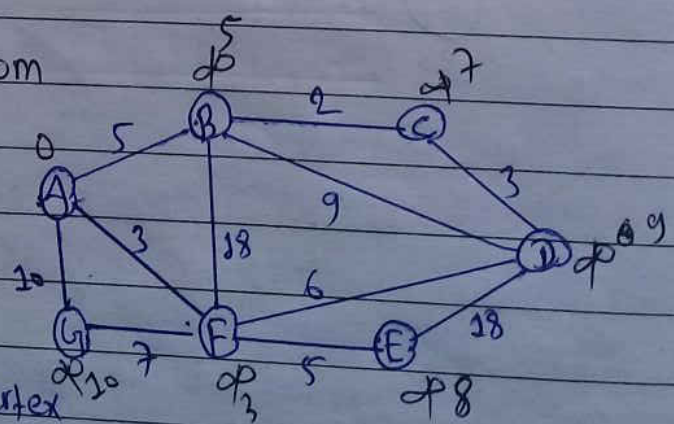
Step 7: After each iteration, we pick the unvisited vertex with least path length.

Step 8: Notice how the rightmost vertex has its path length updated.

Step 9: Repeat all the vertices have been visited.

Questions

1. find the shortest path from A to G.



A = Source vertex

G = Destination vertex

Vertices	A	B	C	D	E	F	G
A	$\boxed{0}$	∞	∞	∞	∞	∞	∞
F		5	∞	∞	∞	$\boxed{3}$	10
B		$\boxed{5}$	∞	9	8		10
C			$\boxed{7}$	9	8		10
E				9	$\boxed{8}$		10
D				$\boxed{9}$			10
G							$\boxed{10}$

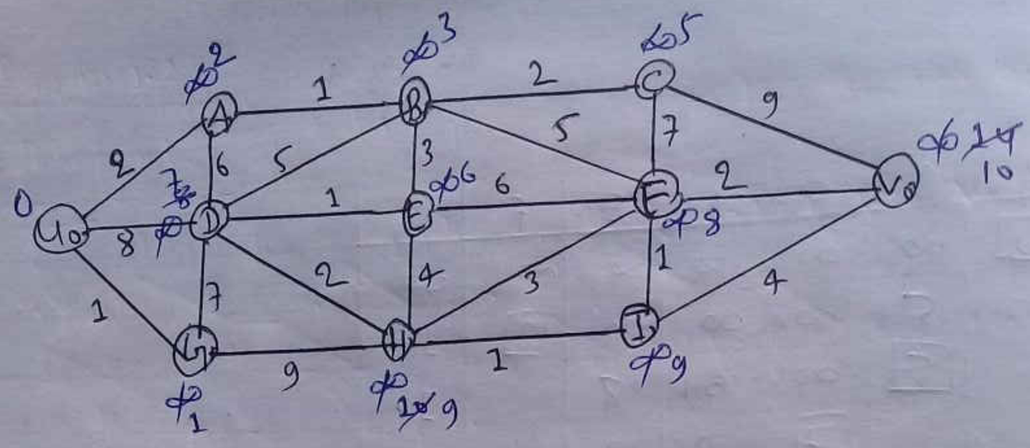
$S = \{A, F, B, C, E, D, G\}$
0 3 5 7 8 9 10

A	F	B	C	E	D	G
0	3	5	7	8	9	10

= ~~0+3+5+7+8+9+1~~

[illegible].

(2)



Vertices	u_0	A	B	C	D	E	F	G	H	I	V_0
u_0	$\boxed{0}$	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
G		2	∞	∞	8	∞	∞	$\boxed{1}$	∞	∞	∞
A		$\boxed{2}$	∞	∞	8	∞	∞		10	∞	∞
B			$\boxed{3}$	∞	8	∞	∞		10	∞	∞
C				$\boxed{5}$	8	6	8		10	∞	∞
E					8	$\boxed{6}$	8		10	∞	14
D					$\boxed{7}$		8		10	∞	14
F							$\boxed{8}$		9	∞	14
I									9	$\boxed{9}$	10
H									$\boxed{9}$		10
V_0											$\boxed{10}$

S =

u_0	G	A	B	C	E	D	F	I	H	V_0
0	1	2	3	5	6	7	8	9	9	10

प्रश्न नं. | उत्तर | प्राप्तांक

Theorems

Theorem 1.1.

If there is a walk from any two vertices to u and v of a graph G, then there is a path from these vertices after the deletion, if necessary, of some vertices and edges.

Solution:

Proof

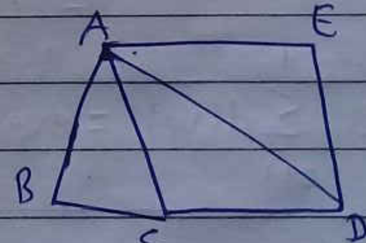
Since path is a walk (without repeated vertices), if there is a walk from u to v then there is is path from u to v.

Now suppose that there is a walk ~~from~~, W from u to v which is not a path - that is, a walk containing repeated vertices. Between any two occurences of a repeated vertex in W, there is a closed walk. Remove from W all such closed walks and the remaining edges will form a path from u to v.

Theorem 1.2

The sum of the degrees of the vertices of a graph is equal to twice the number of edges.

Proof



Let G be a graph having 'n' vertices $v_1, v_2, \ldots v_n$ and 'e' be no. of edges. It is clear that each each contributes degree of 2. Therefore, the sum of degree of all vertices in a graph is twice the no. of edges.

$$\therefore \boxed{\sum_{i=1}^{n} d(V_i) = 2e}$$

From fig

$$\sum d(v) = d(A)+d(B)+d(C)+d(D)+d(E)$$

$$= 4+2+3+3+2$$

$$\sum d(v) = 14$$

$$e = 7$$

Theorem 1.3

The number of odd vertices in a graph is always even.

Proof

Let there be a graph G with $V = \{v_1, v_2, v_3 \ldots v_n\}$ vertices and $E = \{e_1, e_2, e_3 \ldots e_n\}$ edges.

We know that sum of degree of all vertices in a graph G is twice the no. of edges.

i.e. $$\sum_{i=1}^{n} d(v_i) = 2e \quad \text{---} (1)$$

We have

$$\sum_{i=1}^{n} d(v_i) = \sum_{even} d(v_i) + \sum_{odd} d(v_i) \quad \text{---} (2)$$

from eqn (1) and (2)

$$\sum_{even} d(v_i) + \sum_{odd} d(v_i) = 2e$$

or, $$\sum_{odd} d(V_i) = 2e - \sum_{even} d(V_i)$$

$$= even - even$$
$$= even$$

$$\boxed{\therefore \sum_{odd} d(v_i) = even}$$

Hence, proved.

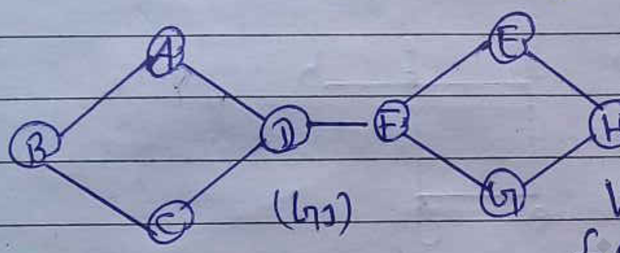
Unit - 2

Isomorphism and Operations

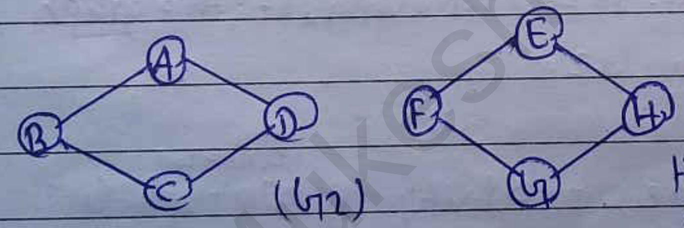
2.1. Connectivity

A graph is said to be connected if there is a path between ~~any two~~ every pair of vertices. From every vertex to any other vertex there must be some path to traverse. This is called connectivity of a graph.

A graph is disconnected, if there exists multiple disconnected vertices and edges.



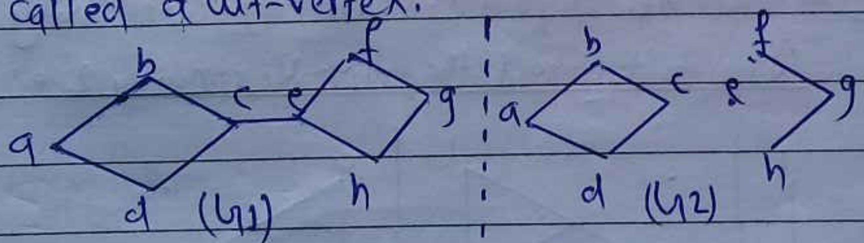
In this graph, G1, it is possible to travel from one vertex to another vertex. We can travel from B to H using path: {B→A→D→F→E→H}. Hence, it is a connected graph.



In this graph it is not possible to traverse from B to H like in G1, because there is no path between them. Hence, it is a disconnected graph.

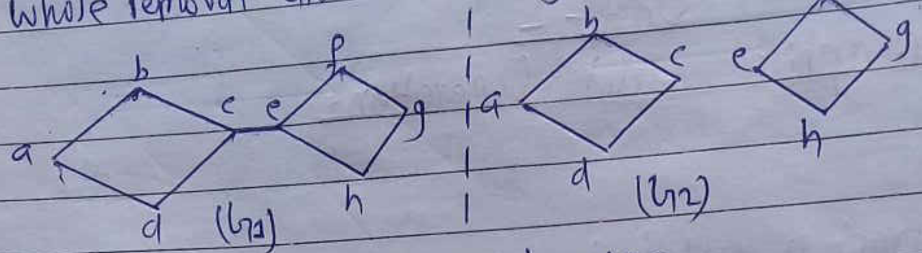
Some Basic Concepts of Connectivity:

1. Cut Vertex → A single vertex whose removal disconnects a graph is called a cut-vertex.



Here, e is a cut vertex. After removing vertex 'e', the graph will become disconnected graph.

2. Cut Edge (Bridge) → A cut edge or bridge is a single edge whose removal disconnects a graph.



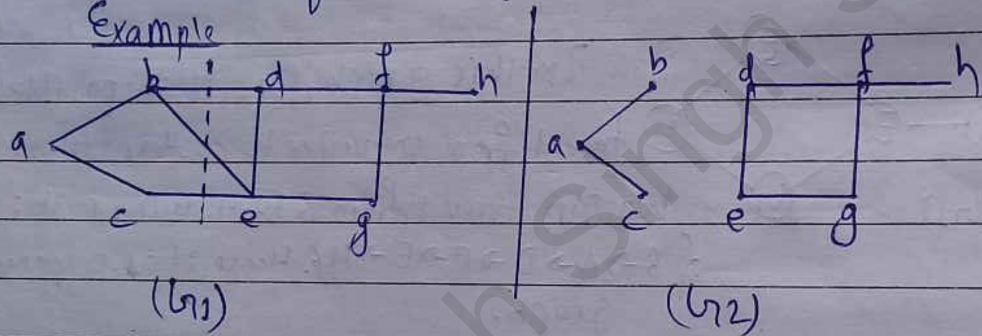
Here, edge 'ce' is a cut edge.

3. Cut Set → In a connected graph G, a cutset is a set of edges with the following properties:

→ The removal of all edges in S disconnects graph G.

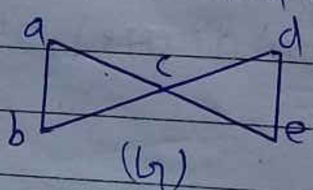
→ The removal of some edges (but not all) does not disconnect G.

Example



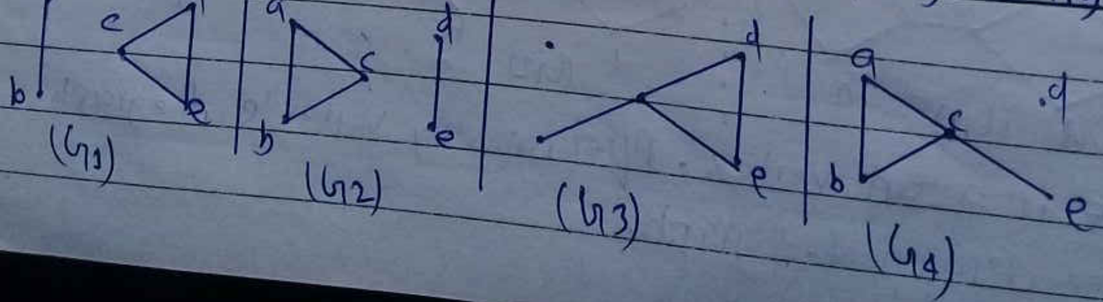
Here (bd, be, ce) is a cut set. After removing it, the graph G_1 gets disconnected as in G_2.

4. Edge Connectivity → The edge connectivity of a connected graph G is the minimum number of edges whose removal makes G disconnected. It is denoted by $\lambda(G)$ or E(G).



In this graph removing 2 minimum edges disconnects a graph. So, its edge connectivity is 2.

Ways to disconnect the graph by removing 2 edges:

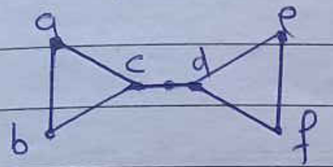


विषय: ....

परीक्षा मिति: .... मूल उ.पु. नं. : ....

प्रश्न नं.	उत्तर	प्राप्ताङ्क

5. Vertex Connectivity → The vertex connectivity of a connected graph G is the minimum number of vertices whose removal makes G disconnected or reduces to trivial graph. It is denoted by $K(G)$

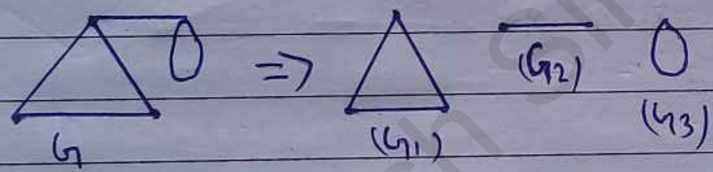


The graph can be disconnected by removing the single vertex either 'c' or 'd'. Hence, its vertex connectivity is 1.

2.2 Blocks

A connected non-trivial graph having no cut point (vertex) is called block. A block of graph is a sub-graph. Every graph is the union of its blocks.

Example



$\therefore G = G_1 \cup G_2 \cup G_3$

2.3. Isomorphism

Graph isomorphism is the phenomenon of existing the same graph in more than one forms. Such graphs are called isomorphic graphs.

Two graphs G_1 & G_2 are isomorphic if:

→ No. of vertices in both graphs are same (i.e. if $V(G_1) = V(G_2)$.)

→ No. of edges in both graphs are same (i.e. if $E(G_1) = E(G_2)$)

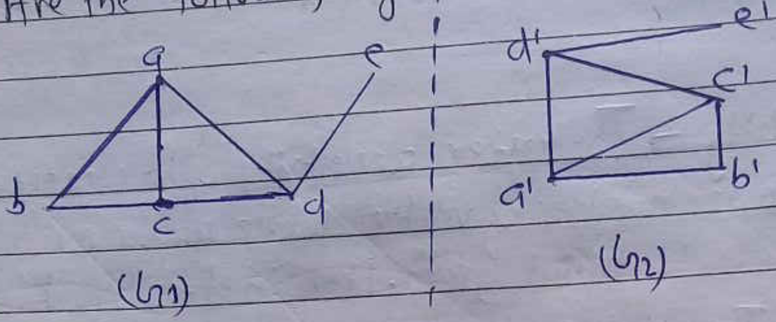
→ Degree sequence of both graphs are same (i.e. $\psi_{G_1} = \psi_{G_2}$) or $G = H$

→ If cycle length k is formed in one graph then cycle of same length k ~~will~~ must be formed by the vertices in the other graph as well.

प्रश्न नं.

उत्तर

Q. Are the following graphs isomorphic ?



Soln:

$V(G_1) = 5$, $E(G_1) = 6$

$V(G_2) = 5$, $E(G_2) = 6$

Degree of G_1, $d(G_1)$ and G_2, $d(G_2)$

$d(a)=3$, $d(b)=2$, $d(c)=3$, $d(d)=3$, $d(e)=1$

$d(a')=3$, $d(b')=2$, $d(c')=3$, $d(d')=3$, $d(e')=1$

Corresponding relation

$\phi(a) = a'$

$\phi(b) = b'$ Since, both graphs satisfies all conditions, hence

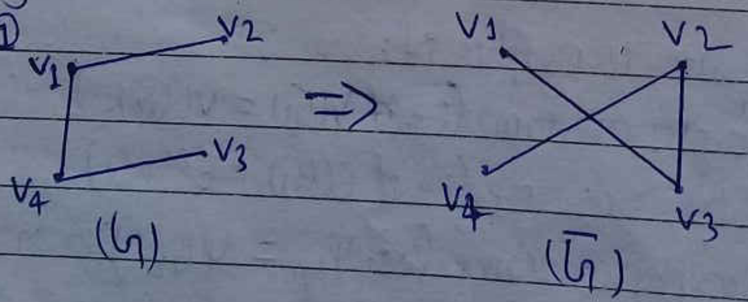
$\phi(c) = c'$ they are isomorphic

$\phi(d) = d'$

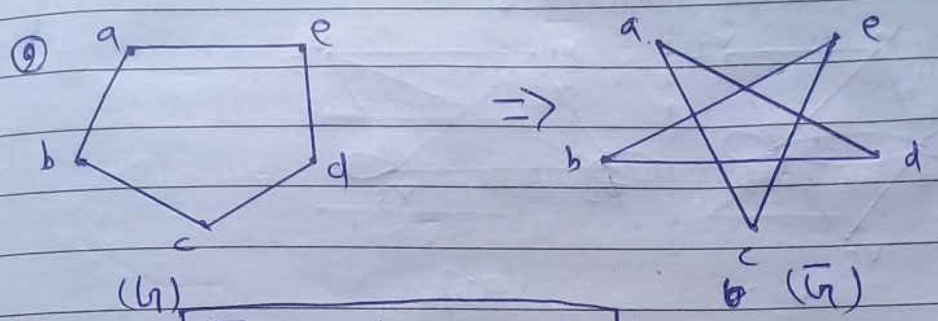
$\phi(e) = e'$

2.4. Some Special Graphs

1. Complement of a graph → The complement of a graph G is the simple graph with same vertex set V & with two vertices adjacent if and only if they are not adjacent in G. It is denoted by G^c or $\overline{G}$.

① 

②



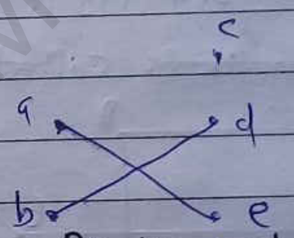
(G) $\qquad$ $(\bar{G})$

$$\boxed{\bar{G}_{com} = G_{complete} - G}$$

2. Self Complementary graph → A graph G is said to be self complementary if $G \cong G^c$.

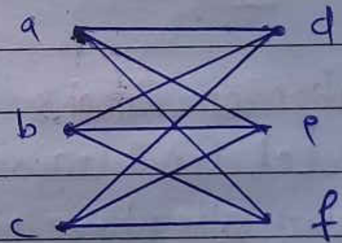
3. Complete Bipartite Graph

A graph $G=(V,E)$ with vertex partition $V=\{V_1, V_2\}$ is called bipartite graph if an edge connects any vertex in V_1 to any vertex in set V_2.



Bipartite graph

But, a bipartite graph $G=(V,E)$ with vertex partition $V=\{V_1, V_2\}$ is complete bipartite, if every vertex in set V_1 connects to every vertex in V_2.

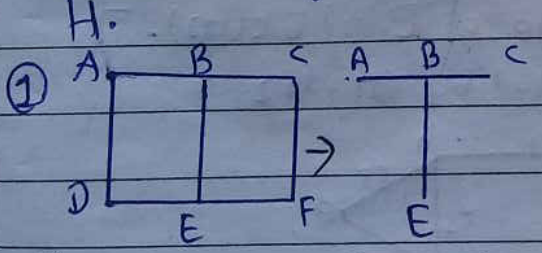


Complete Bipartite graph

2.5. Subgraphs

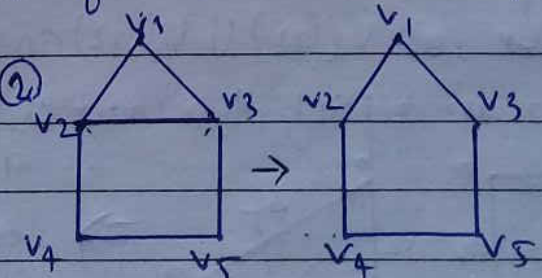
A graph $H=\{V_1, E_1\}$ is called subgraph of $G=\{V,E\}$ if $V_1 \subseteq E$. If H is a subgraph of G, then G is a super graph of H.

①



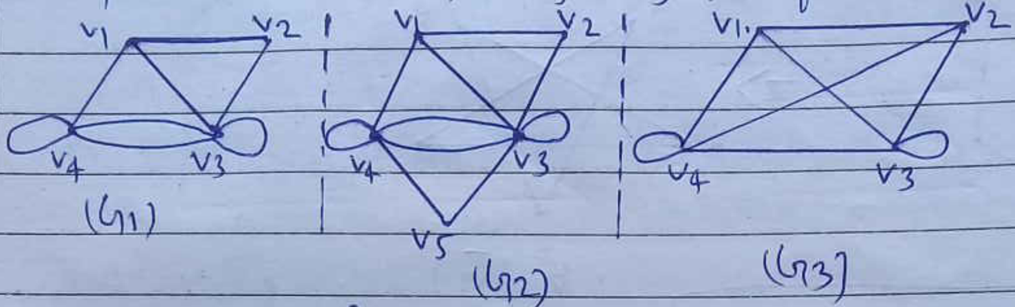
Supergraph G $\qquad$ Subgraph H

②



Supergraph G $\qquad$ Subgraph H

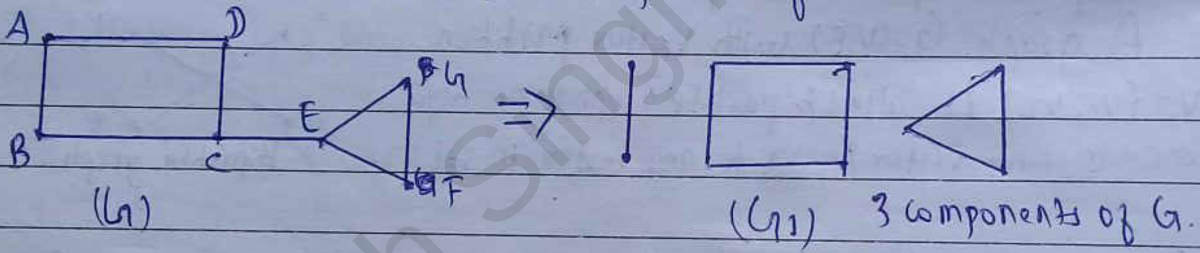
Spanning Subgraph – Consider H be a subgraph of G with $V(H) = V(G)$. It means H and G have exactly the same vertex set. Then, H is called the spanning subgraph of G.



In the figure, G_1 is the proper subgraph of G_2 and G_1 is the spanning subgraph of G_3.

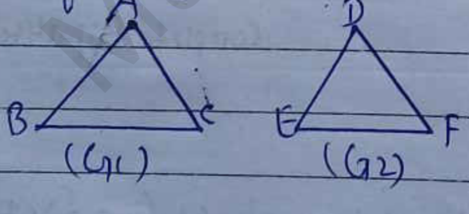
Component of a graph:

Let G_1 is a maximal connected subgraph of a graph G, then G_1 is called connected component of G.

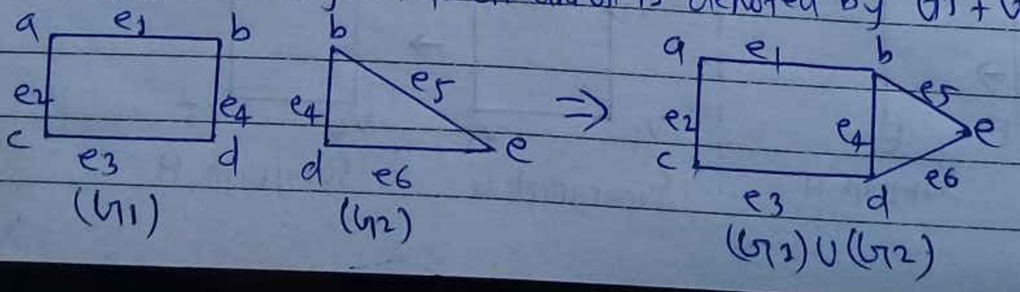


2.6. Operation on Graphs

Let G_1 & G_2 are two subgraphs of G. G_1 & G_2 are disconnected if they have no common vertex.

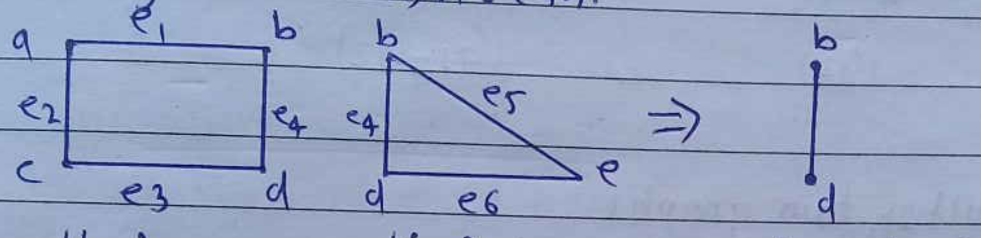


1. **Union of Graphs** → The union of G_1 and G_2 is the subgraph with vertex set $V(G_1) \cup V(G_2)$ and edge set $E(G_1) \cup E(G_2)$. If G_1 and G_2 are disjoint their union is denoted by $G_1 + G_2$



2. Intersection of Graphs

If G_1 and G_2 have at least one common vertex then their intersection is the subgraph with vertex set $V(G_1) \cap V(G_2)$ and edge set $E(G_1) \cap E(G_2)$.



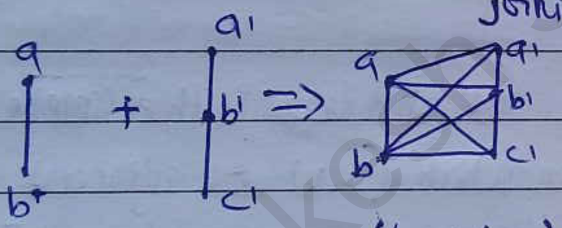
(G_1) (G_2) $(G_1) \cap (G_2)$

3. Sum of two graphs

If G_1 and G_2 are two graphs then sum of two graphs is denoted by $G = G_1 + G_2$.

if $V(G_1 + G_2) = V(G_1) + V(G_2)$

$E(G_1 + G_2) =$ ~~$E(G_1) + E(G_2)$~~, those edges in G_1 & G_2 obtained by joining each vertex of G_1 to each vertex of G_2.



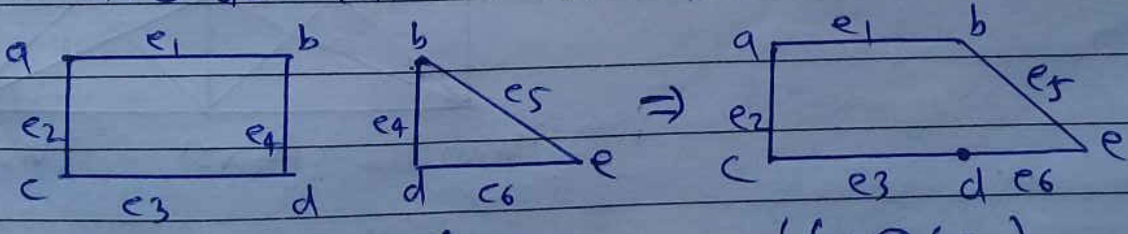
All vertices of G_1 matched to all vertices of G_2

(G_1) (G_2) $(G_1 + G_2)$

Ringsum of G_1 and G_2

It contains all the edges either in G_1 or G_2 but not in both

[i.e. $G_1 \oplus G_2 = (G_1 \cup G_2) - (G_1 \cap G_2)$]

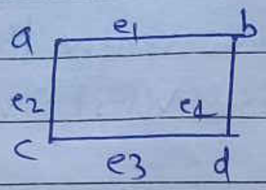


(G_1) (G_2) $(G_1 \oplus G_2)$

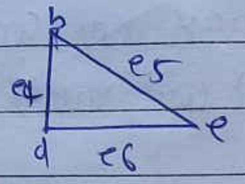
4. Difference of two graphs

If G_1 and G_2 are two graphs, then their difference is denoted by:

$G = G_1 - G_2$ [Contains all edges in G_1 but not in G_2]

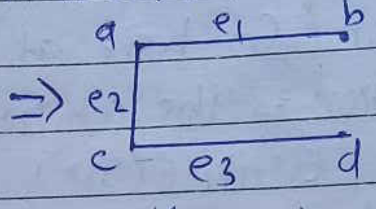


(G_1)



(G_2)

$\Rightarrow$



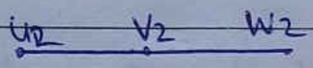
($G_1 - G_2$)

5. Cartesian Product of two graphs

$V(G_1 \times G_2) = V(G_1) \times V(G_2)$

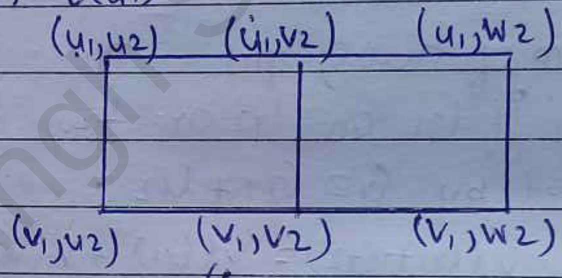
$E(G_1 \times G_2) = V(G_1) \times E(G_2) \cup V(G_2) \times E(G_1)$

(G_1)



(G_2)

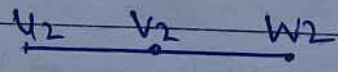
$\Rightarrow$

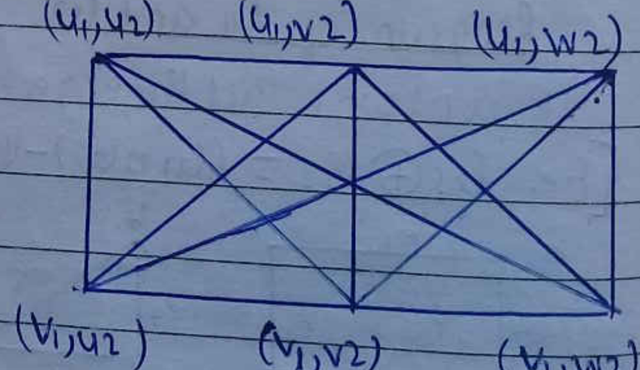


($G_1 \times G_2$)

6. Composition of two graphs

The composition of two graphs G_1 and G_2 is the simple graph with vertex set $V(G_1) \times V(G_2)$ in which (u,v) is adjacent to (u',v') if and only if either $uv' \in E(G)$ are $u = u'$ and $w' \in E(G_2)$. It is denoted by $G_1(G_2)$.





$G_1(G_2)$

Theorems

Theorem 2.1

Let G_1 and G_2 be the isomorphic graphs, with isomorphism ϕ. Then,

(i) G_1 and G_2 have the same number of vertices.

(ii) G_1 and G_2 have the same number of edges.

(iii) If u is adjacent to v is G_1, then $\phi(u)$ is adjacent to $\phi(v)$ is G_2,

(iv) If u has degree k is G_1, then $\phi(u)$ has degree k is G_2.

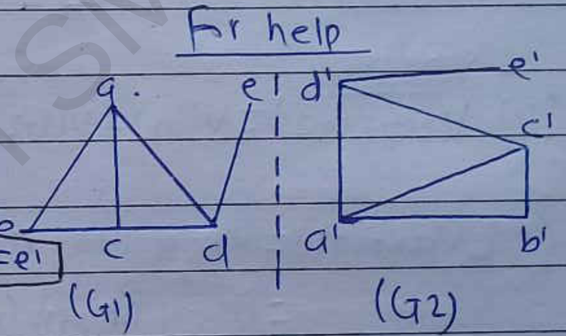
$\boxed{V(G_1)=5,\ E(G_1)=6,\ V(G_2)=5,\ E(G_2)=5}$

$\boxed{d(a)=3, d(b)=2, d(c)=3, d(d)=3, d(e)=1}$

$\boxed{d(a')=3, d(b')=2, d(c')=3, d(d')=3, d(e')=1}$

$\boxed{\phi(a)=a',\ \phi(b)=b',\ \phi(c)=c',\ \phi(d)=d',\ \phi(E)=e'}$

For help



(i) Since, G_1 and G_2 are isomorphic graphs, the vertices of G_1 and G_2 are one-to-one correspondance. Hence, the number of vertices in G_1 and G_2 are same.

(ii) Here, each edge $\{u,v\}$ in G_1 is correspondent with unique edge $\{\phi(u), \phi(v)\}$ in E_2 and vice-versa. It implies that $|E_1|=|E_2|$. Hence, G_1 and G_2 have same no. of edges.

(iii) If u is adjacent to v, then $\{u,v\}$ belongs to G_1. Then, the vertex $\{\phi(u), \phi(v)\}$ belongs to G_2. Therefore, $\phi(u)$ is adjacent to $\phi(v)$ in G_2.

(iv) If u has degree k in G_1, then no. of edges are k in G_1. The no. of edges are also k in G_2. Therefor degree of G is also k.

[If $u=k$, then $\phi(u)=k$]

प्रश्न नं.

Theorem 2.2

Let G_1 be a (p_1,q_1) and G_2 be a (p_2,q_2) graph. Then prove that:

(i) $G_1 \cup G_2$ is a (p_1+p_2, q_1+q_2) graph.

(ii) $G_1 + G_2$ is a $(p_1+p_2, q_1+q_2+p_1p_2)$ graph.

(iii) $G_1 \times G_2$ is a $(p_1p_2, q_1p_2+q_2p_1)$ graph.

(i) G_1 be a (p_1,q_1) and G_2 be a (p_2,q_2)

$V(G_1)=p_1,\ V(G_2)=p_2$

$E(G_1)=q_1,\ E(G_2)=q_2$

$V(G_1 \cup G_2)=V(G_1)\cup V(G_2)=p_1+p_2$

$E(G_1 \cup G_2)=E(G_1)\cup E(G_2)=q_1+q_2$

$\therefore G_1 \cup G_2$ is a (p_1+p_2, q_1+q_2) graph.

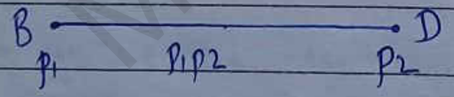
(ii) $V(G_1+G_2)=V(G_1)+V(G_2)=p_1+p_2$

$E(G_1+G_2)=$ no. of lines in (G_1+G_2)

$=$ (no. of lines in G_1) + (no. of lines in G_2) + (no. of lines joining points of V_1 to the points of V_2)

$= q_1+q_2+p_1p_2$

$\therefore (p_1+p_2, q_1+q_2+p_1p_2)$ is a graph.



(iii) No. of points in $G_1 \times G_2$ is p_1p_2

i.e. $V(G_1 \times G_2) = V(G_1)\times V(G_2) = p_1p_2$

Now,

Let $(u_1,u_2) \in V_1 \times V_2$

The points adjacent to (u_1,u_2) are (v_1,v_2), where u_2 is adjacent to v_2 (v_1,v_2) where adjacent to u_1.

$\therefore \deg(u_1,u_2) = \deg(u_1)+\deg(u_2)$

$\therefore$ Total no. of lines is $G_1 \times G_2$

$$= \frac{1}{2}\left[\sum_{i,j} \deg(u_i) + \deg(v_j)\right] \quad \text{where } u_i = V_1, \; v_j = V_2$$

$$= \frac{1}{2}\left[\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \left(\deg u_i + \deg v_j\right)\right], \text{ where } u_i \in V_1, v_j \in V_2$$

$$= \frac{1}{2}\left[\sum_{i=1}^{p_1} (p_2 \deg u_i) + \sum_{j=1}^{p_2} (\deg v_j)\right]$$

$$= \frac{1}{2} \sum_{i=1}^{p_1} \left(p_2 \deg u_i + 2q_2\right)$$

$$= \frac{1}{2}\left(2p_2q_1 + 2p_1q_2\right)$$

$$= p_2q_1 + p_1q_2$$

$\therefore (p_1p_2, p_2q_1 + p_1q_2)$ is a graph.

Unit-3
Eulerian Tours and Hamiltonian Cycles

3.1. Eulerian graphs

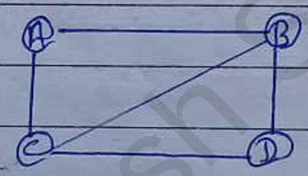
Euler path → An Euler path is a path that passes through every edge exactly once. Euler path is also known as Euler Trial or Euler Walk.

Euler trial → If there exists a trial in a connected graph containing all the edges of the graph, then that trial is called Euler trial.

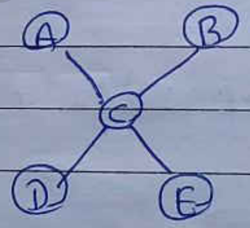
Euler Walk → If there exists a walk in a connected graph, that visits every edge once with or without repeating the vertices, then such walk is called an Euler Walk.

Note

A graph contains an Euler path if it contains at most two vertices of odd degree.



Euler Path = BCDBAD

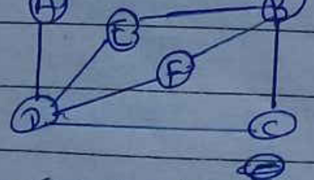


Euler path doesn't exist.

Euler Circuit → If there exists a circuit in the connected graph that contains all the edges of the graph, then that circuit is called as an Euler Circuit. Euler circuit is also known as Euler Cycle or Euler Tour.

Or,

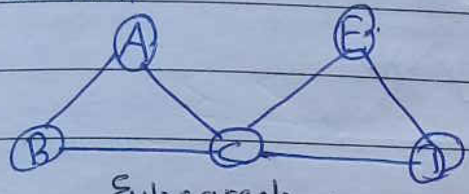
An Euler trial that starts and ends at the same vertex (i.e. Closed Euler trial) is called an Euler Circuit.



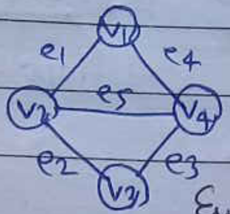
Note: A graph contains an Euler circuit iff all its vertices are of even degree

Euler Circuit = ABCDFBEDA

Euler graph → A connected graph is called an Euler Graph iff all its vertices are of even degree. It contains an Euler Circuit or trial.



Euler graph

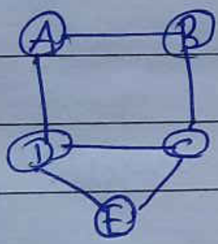


Euler graph

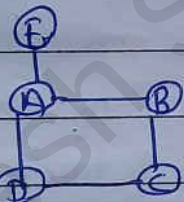
2. Hamiltonian Graph

Hamiltonian path → If there exists a path in a connected graph containing all the vertices of the graph, then such a path is called as a Hamiltonian path.

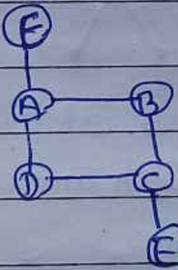
Note – Edges may or may not be covered but must not repeat.
Visits every vertex exactly once.

① 

Hamiltonian path = ABCDE

② 

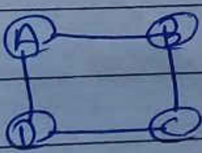
H. Path = EABCD

③ 

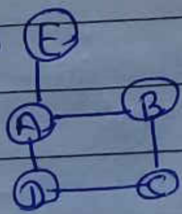
Does not exist H. Path

Hamiltonian Circuit → A hamiltonian path (closed H. path) which starts and ends at the same vertex is called as a Hamiltonian Circuit.

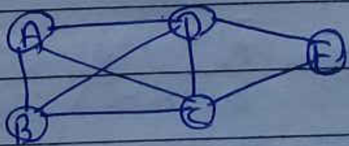
→ Also called Hamiltonian cycle. → No edge repeat.

① 

H. Circuit = ABCEDA

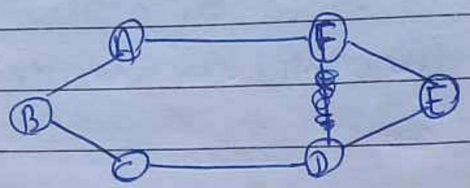
② 

Does not contain ~~H. Circuit~~

③ 

H. Circuit is: A→B→C→E→D→A
Or, A→C→E→D→B→A

Hamiltonian graph → Any connected graph that contains a Hamiltonian circuit is called as a Hamiltonian graph.
~~No.~~It is a closed walk • No edge repeat •



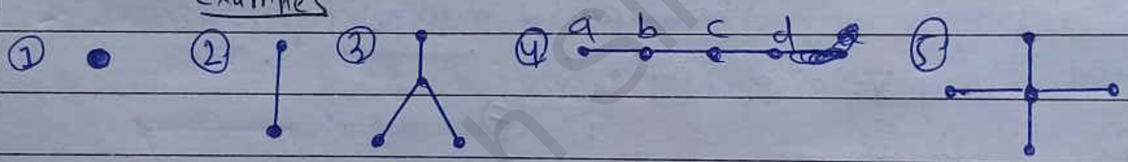
It contains a closed walk or H. circuit, A → B → C → D → E → F → A, therefore it is a Hamiltonian graph.

33. Trees and forests

Trees → A tree is an undirected, connected and acyclic graph having no cycle.

- Elements of trees are called nodes.
- Edges of trees are called branches.
- ~~Edges~~ Nodes without child nodes are called leaf nodes.

Examples

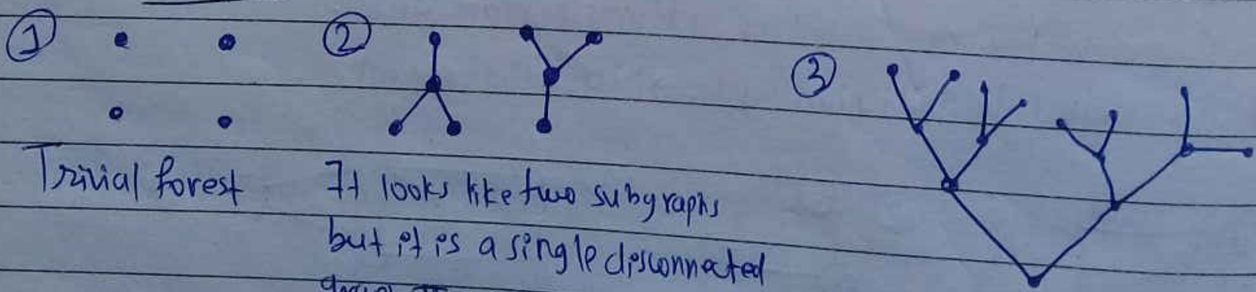


→ Every tree has at least two vertices of degree 1.
→ But in fig ④ b and c has degree 2 but do not form a cycle.

Forests → forest is an undirected, disconnected and acyclic graph. It is disjoint collection of trees.

- Each component of a forest is tree.

Examples



Trivial forest

It looks like two subgraphs but it is a single disconnected graph. There are no cycles. Hence, it is a forest.

3.4. Spanning Trees

Let G be a connected graph, then the subgraph H of G is called a spanning tree of G if:

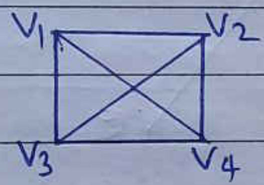
→ H is a tree.

→ H contains all vertices of G.

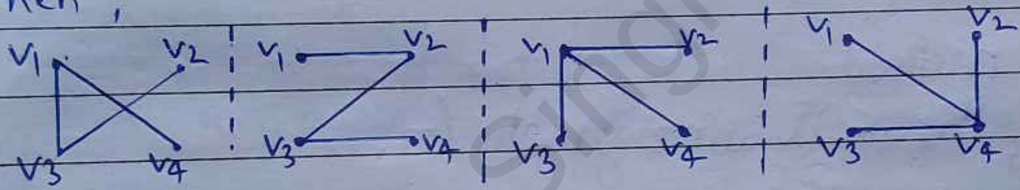
A Spanning tree of an undirected graph G is a subgraph that includes all of the vertices of G.

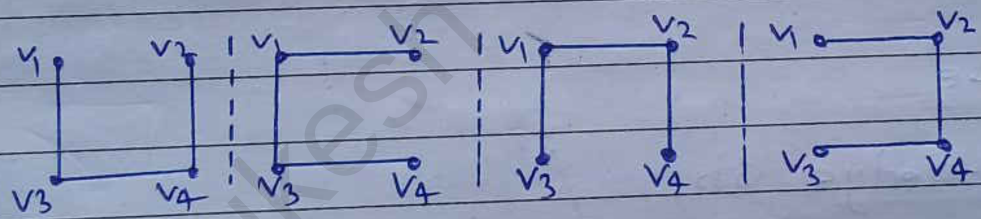
Example

① If G =



Then,



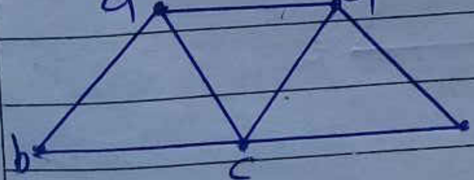


etc. and so on are spanning trees (H) of graph (G).

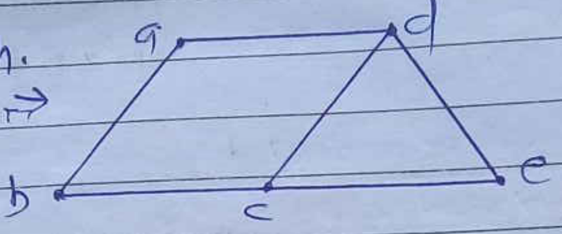
Methods of finding the Spanning trees

1. Cutting-down method:
- Start choosing any cycle in graph G.
- Remove one of cycle's edges.
- Repeat this process until there are no cycles left.

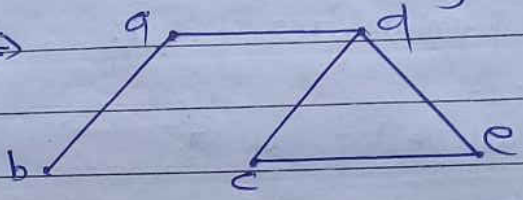
Example: Consider a graph G:



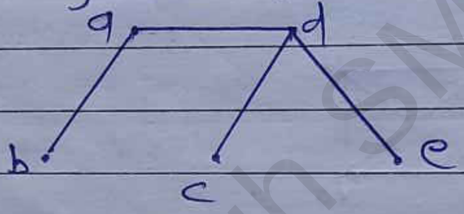
If we remove the edge ac which destroy the cycle a-d-c-a in the above graph, then we get following graph. ⇒



Remove the edge eb which destroy the cycle a-d-c-b-a. ⇒

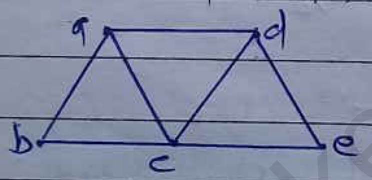


If we remove the cycle ec, which destroy the cycle d-e-c-d →

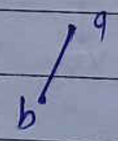


2. Build up method

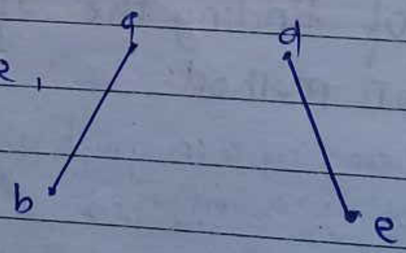
Consider the following graph G.



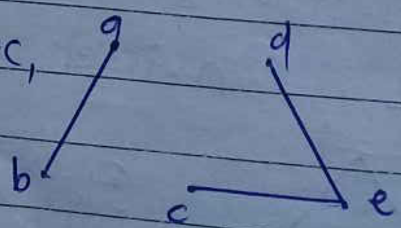
- Choose the edge ab,



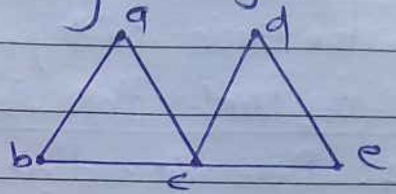
- Choose the edge de,



- After that choose the edge ec,



Next choose the edge ch, then finally we get the following spanning tree.



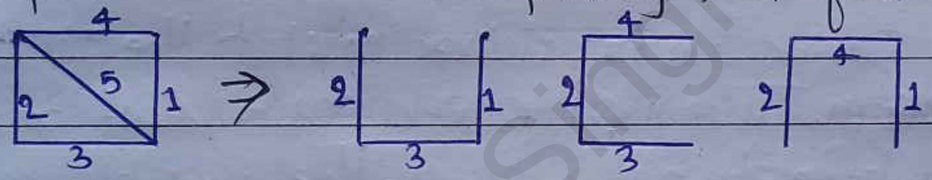
· Minimal Spanning Trees

A minimal spanning tree for a weighted connected graph G is defined as a spanning tree T which has the smallest possible weight so that if T' is any other spanning tree in G then:

$$W(T) \leq W(T')$$

Example   Some spanning trees of G are:

G =



~~W(H1)=6~~   W(H1)=9   W(H2)=9   W(H3)=7

Among them, H3 is the minimal spanning tree because it has minimum weight than other spanning trees.

Applications

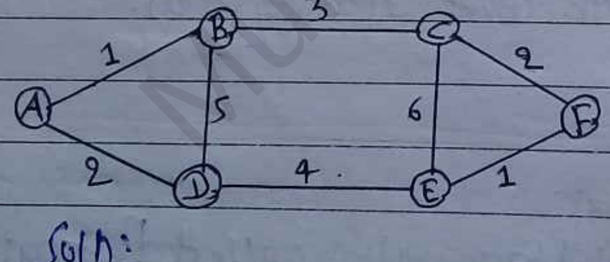
.5 The Chinese Postman Problem

The Chinese Postman Problem, also called Postman Tour or Route Inspection ~~Proth~~ Problem, is a famous mathematical problem in Graph Theory. The postman's job is to deliver all of the town's mail using the shortest route possible. In order to do so, he/she must pass each street once and then return to the origin.

Algorithm for Solving Chinese Postman Problem:

Step 1: list/find all the odd vertices in the graph.
Step 2: Determine & list all possible pairings of odd vertices.
Step 3: For each pairing, find the edge that connect the odd vertices with the shortest possible path (Dijkstra's)
Step 4: Determine the combination of pairings that has the shortest total length.
Step 5: Add the combination of pairings (edges) found previously (in step 4), to the original graph.
Step 6: The length of optimal Chinese postman route is the sum of all the edges in augmented/modified graph.
Step 7: A route corresponding to this minimum weight can then be easily found.

Q1. A mail carrier has to deliver mail to ~~the~~ all streets in a subdivision. The mail carrier has to start and end at point A, where the post office is. The weight of each edge represents the length of each street. What is the most efficient route for the mail carrier?



Soln:

1. The odd vertices are BCDE.
2. There are 3 ways of pairing these odd vertices.
3. The shortest way of joining pairs is using the path/

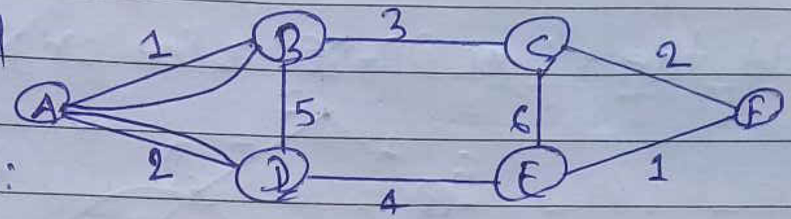
Pairs:

- BC - DE ⇒ 3+4 = 7
- BD - CE ⇒ 3+3 = 6
- BE - CD ⇒ 6+6 = 12

प्रश्न नं. उत्तर

4. Draw these edges onto the graph:

Find a route round all points from A with at least cost:



Smallest + Total = 6+24 = 30

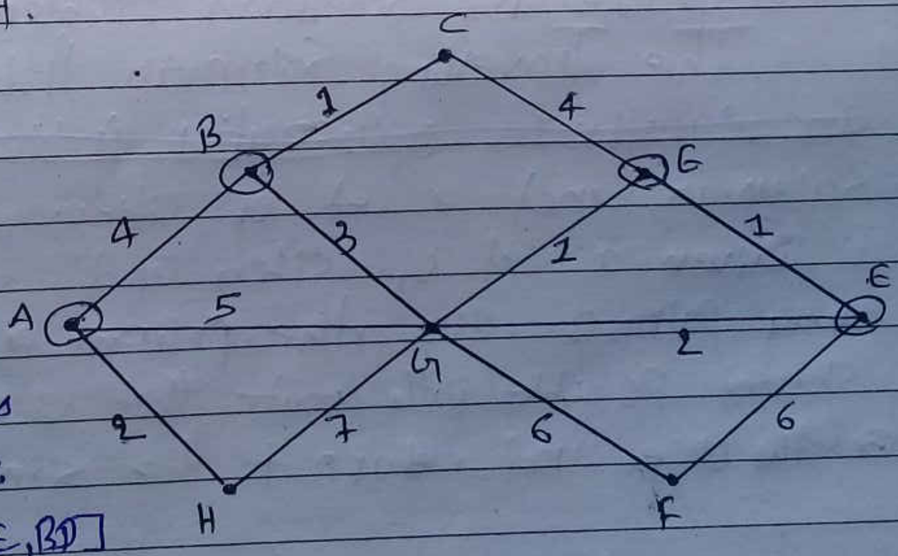
5. The length of the optimal Chinese postman route is the sum of all the edges in the original graph, which is 24, plus the answer found in step 4, which is 6. So the length is 30.

6. Find out the route:

i.e. A->B->C->F->E->E->F->D->B->A->D->A

Traversing all the edges exactly once with the total Cost 30.

Q2. Find the most efficient route for the mail carrier starting at point A.



Total weight = 42

Soln:

1. The odd vertices are ABDE.

2. The possible pairs of odd vertices are:
[AB, DE], [AD, BE], [AE, BD]

3. Find the pair with minimum weight:

~~AB~~ -

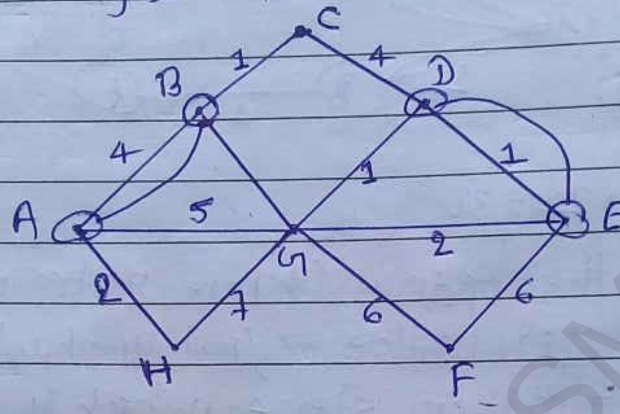
AB - DE ⇒ 4+1 = 5

AD - BE ⇒ 6+5 = 11

AE - BD ⇒ 7+4 = 11

Here, the pair AB-DE has minimum weight 5.

4. Add the edges AB, DE to the original graph.



5. Optimal Chinese Postman route is of length :
5 + 42 = 47 (42 = Sum of all edges of modified graph)

6. The Chinese Postman route is:
A-B-C-D-E-D-G-E-F

3.6 The Travelling Salesman Problem

The Travelling Salesman Problem (TSP) consists of ~~an algorithmic optimization problem in graph theory~~ a salesman and a set of cities.

Given a set of cities and distance between every pair of cities the problem is to find the shortest possible path/route that visits every city exactly once and returns to the same starting city.

Q. Given the set of cities and distance between every pair of cities. Find the shortest route that visits every city exactly once and returns to the

मूल उ.पु. नं. :

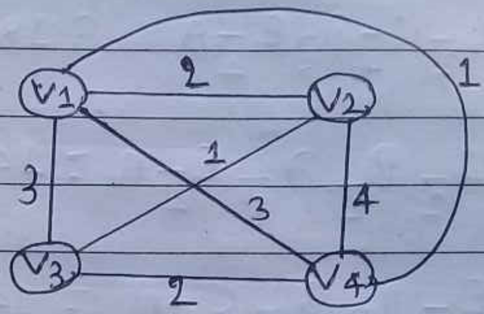
प्रश्न नं. उत्तर

starting city using TSP.

Soln:

Using matrix

$$\begin{array}{c|cccc} & & & \text{city} & \\ & v_1 & v_2 & v_3 & v_4 \\ \hline v_1 & \infty & 2 & 3 & 1 \\ v_2 & 2 & \infty & 1 & 4 \\ \text{city}\ v_3 & 3 & 1 & \infty & 2 \\ v_4 & 1 & 4 & 2 & \infty \end{array}$$



$V_1 \xrightarrow{1} V_4 \xrightarrow{2} V_3 \xrightarrow{1} V_2 \xrightarrow{2} V_1$

1) Starting at V1 look the shortest distance and mark it, i.e. V4

2.) From v4 shortest distance is 1 at v1 which is already visited. Next short distance is 2 at v3.

3) from V3 shortest is 1 at V2 and mark V2.

4) From V2 shortest distance is 1 which is already visited. The next is 2 at V1 and mark V1.

The final shortest route is:

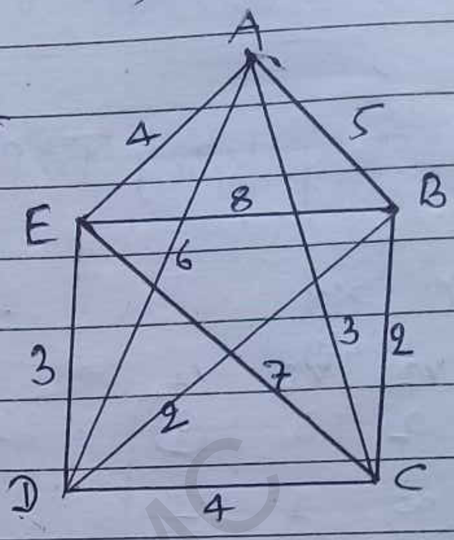
$V_1 \xrightarrow{1} V_4 \xrightarrow{2} V_3 \xrightarrow{1} V_2 \xrightarrow{2} V_1$

Total Cost $= 1+2+1+2 = 6$

Q8.

प्रश्न नं. उत्तर

Q2. Find the shortest route starting at A.



Routes	Total distance
① $A \underset{5}{-} B \underset{2}{-} C \underset{4}{-} D \underset{3}{-} E \underset{4}{-} A$	$= 18$
② $A \underset{3}{-} C \underset{2}{-} B \underset{2}{-} D \underset{3}{-} E \underset{4}{-} A$	$= 14$
③ $A \underset{4}{-} E \underset{3}{-} D \underset{2}{-} B \underset{2}{-} C \underset{3}{-} A$	$= 14$

∴ Shortest routes are ② and ③

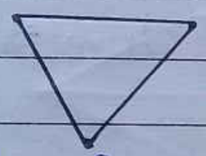
Theorems

Unit-4
Planar Graphs and Colouring

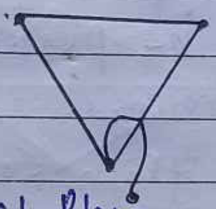
4.1. Plane and Planar Graphs

A planar graph is a ~~planar~~ graph that ~~can be~~ has been drawn in the plane without any edge crossing. A graph is planar if it is isomorphic to a plane graph.

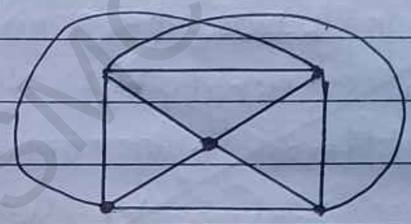
Plane vs Planar



Plane & Planor



not Plane but planar

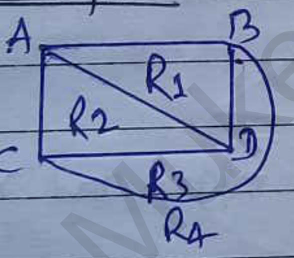


neither plane nor planar

A plane graph divides the plane into ~~two~~ regions:

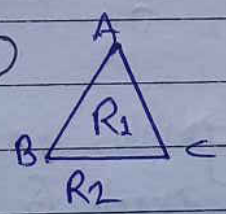
Examples

①



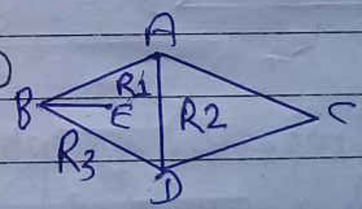
$v = 4$
$e = 6$
$r = 4$
$v - e + r = 2$

②



$v = 3$
$e = 3$
$r = 2$
$v - e + r = 2$

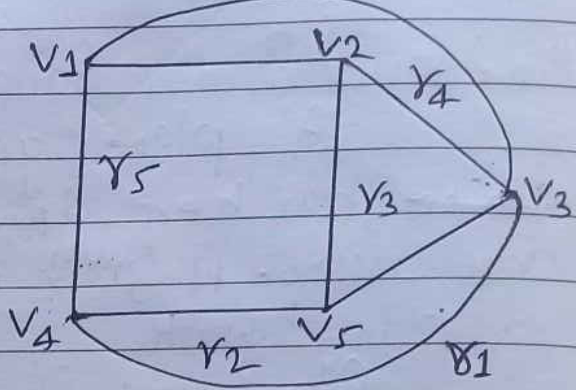
③



$v = 5$
$e = 6$
$r = 3$
$v - e + r = 2$

Region of a graph → An area of a plane that is bounded by edges and cannot be further sub-divided.
i) finite region → It is interior region.
ii) Infinite region → It is exterior region.

Example: Determine number of regions, finite and infinite regions.

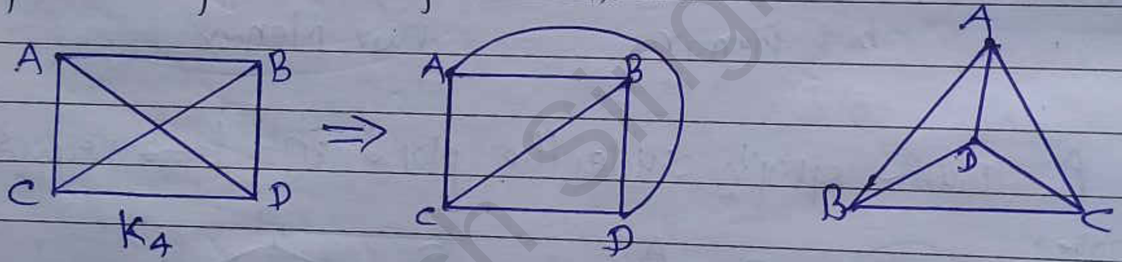


Total regions $= 5$ (i.e. r_1, r_2, r_3, r_4, r_5)
Finite regions $= 4$ (i.e. r_2, r_3, r_4, r_5)
Infinite region $= 1$ (i.e. r_1)

Note

A graph may be planar even if it is drawn with edge crossings, but it may be possible to draw it in a different way without crossings.

Example- Consider complete graph K_4 and its two possible planar representations:



Properties of planar graphs:

1) If a connected planar graph G has e edges and r regions, then $r \leq \frac{2}{3}e$.

2) If a connected graph G has e edges, v vertices and r regions, then $v - e + r = 2$.

3) If a connected planar graph G has e edges and v vertices, then $3v - e \geq 6$.

4) A complete graph K_n is a planar if and only if $n < 5$.

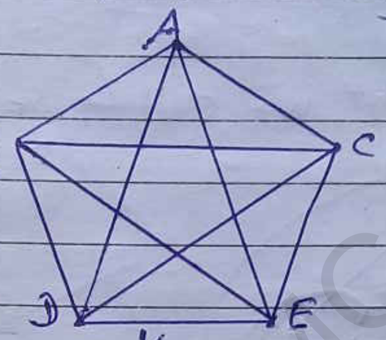
5) A complete bipartite graph K_{mn} if and only if

$m<3$ or $n>3$.

Non-Planar Graph

A graph is said to be non-planar if it cannot be drawn in a plane so that no edges cross.

Properties of non-planar graph



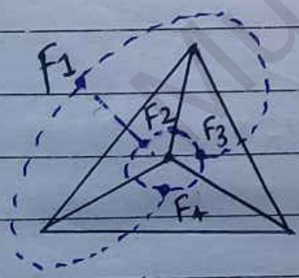
K_5

→ A graph is non-planar if and only if it contains a subgraph homeographic to K_5 or $K_{3,3}$.

4.2 ## Dual Graphs

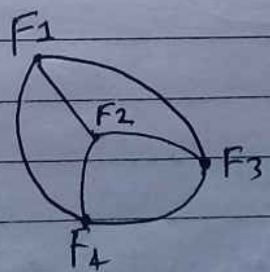
The dual graph of a plane graph G is a graph that has vertex for each face of G. It is denoted by G^*.

Each edge 'e' of 'G' has a corresponding dual edge, whose endpoints are the dual vertices corresponding to the faces on either side of 'e'.



(G)

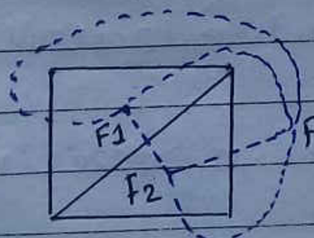
⇒



(G^*)

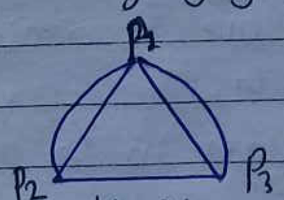
Q. Draw the dual graphs of following graphs:

a)



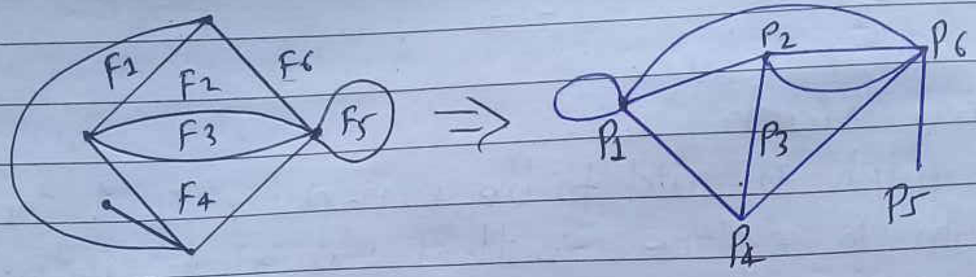
(G)

⇒



(G^*)

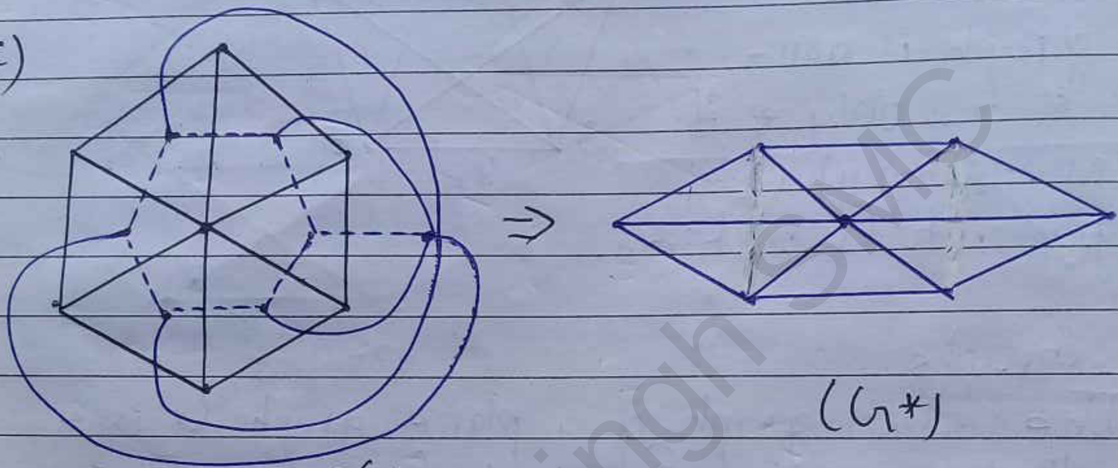
b)



(G) (G*)

If $G = G^*$ we call G self-dual.

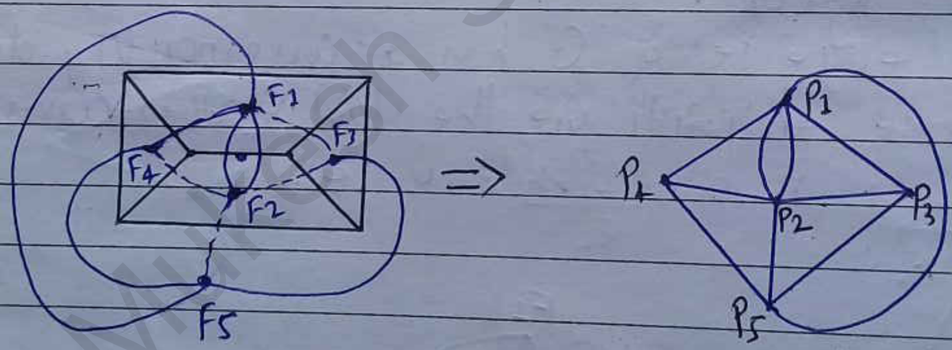
c)



(G*)

(G)

d)



(G) (G*)

Observations

i) An edge forming a loop in G yields a pendant edge in G^*.

ii) A pendant edge in G yields a self loop in G^*.

iii) Edges that are in series in G produce parallel edges in G.

iv) Parallel edges in G produce edges in Series in G^*.

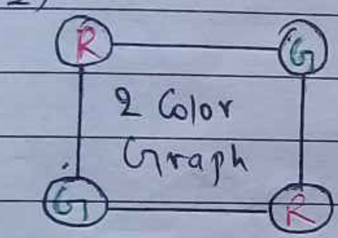
Graph Colouring

4.3 Coloring Vertices

Graph colouring is also called as vertex colouring.
Graph colouring is a process of assigning colors to the vertices of a graph such that no two adjacent vertices of it are assigned the same colour.
It ensures that there are no edges in the graph whose end vertices are coloured with the same colour.
Such a property is called a properly coloured Graph.

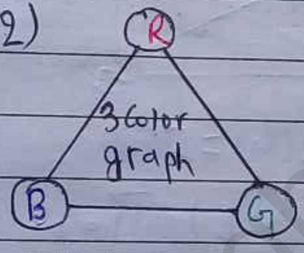
Examples

1)



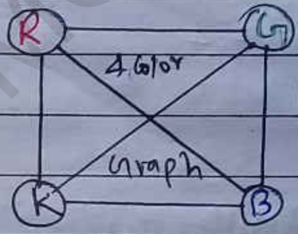
$X(G) = 2$

2)



$X(G) = 3$

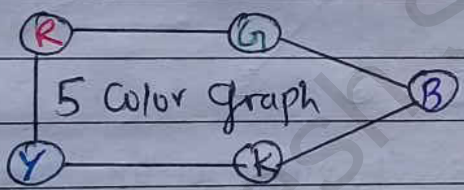
3)



K = black

$X(G) = 4$

4)



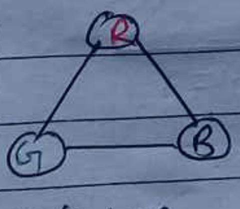
Chromatic Number

It is the minimum number of colors required to properly color any graph such that no two adjacent vertices of it are assigned the same color. It is denoted by $X(G)$
For example the chromatic number of K_n is n.

Examples

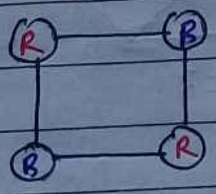
A. Cycle Graphs

1)



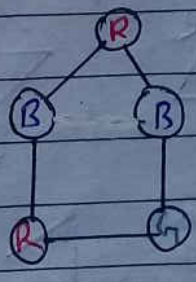
$X(G) = 3$

2)



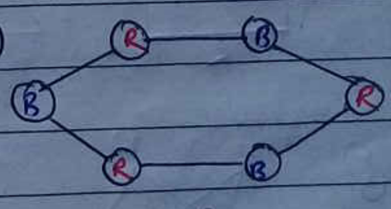
$X(G) = 2$

3)



$X(G) = 3$

4)



$X(G) = 2$

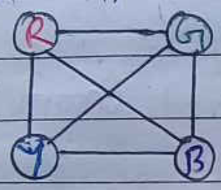
In cycle graph, even vertices $X(G) = 2$, odd vertices, $X(G) = 3$.

B. Planar Graphs

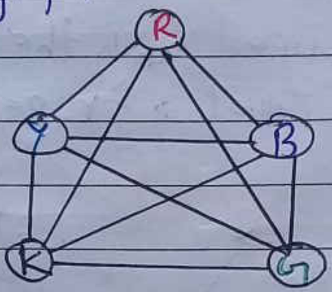
Chromatic number of any planar graph = less than or equal to 4. Above/previous graphs are planar.

C. Complete Graphs Colouring

Chromatic number of any complete graph = no. of vertices in that complete graph.

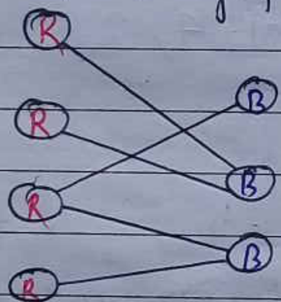
① 

$\chi(G) = 4$

② 

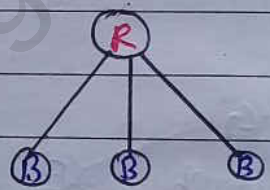
$\chi(G) = 5$

D) Bipartite graph Coloring



$\chi(G) = 2$

E) Tree Colouring

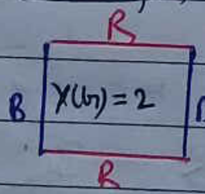


Chromatic number of any tree = 2.

4.4 Coloring Edges

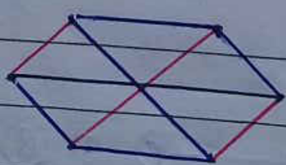
Edge Colouring is the assignment of "Colors" to edges so that no two adjacent edges have the same Colors.

Examples

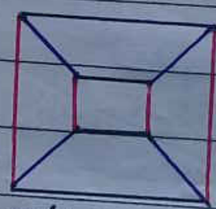
① 

② 

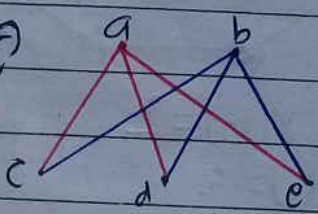
$\chi(G) = 2$

③ 

$\chi(G) = 3$

④ 

$\chi(G) = 3$

⑤ 

⑥ 

4.5 Colouring Maps

- Color a map such that two regions with a common border are assigned different colors.
- Each map can be represented by a graph.
- Each region of a map is represented by a vertex.
- Edges connect two vertices if the regions represented by these vertices have a common border.
- The resulting graph is called the dual graph of the map.

The four Color Map theorem, states that, given any separation of a plane into contiguous regions, producing a figure called map, no more than four colors are required to colour the regions of the map so that no two adjacent regions have the same color.

Theorems

Theorem 1: If G is a connected planar graph with $|V| = v$, $|E| = e$ and r number of regions, then $v - e + r = 2$.

Proof

Let us prove the theorem by means of induction.

First suppose, $e = 0$ then $v = 1, r = 1$.

So, $v - e + r = 1 - 0 + 1 = 2$

If $e > 0$, it has two cases:

(i) G has no cycles.

(ii) G has at least one cycle.

Case 1: Let G has no cycle, it has no closed regions.

Then $v = n, e = n - 1, r = 1$

$\therefore v - e + r = n - (n-1) + 1$

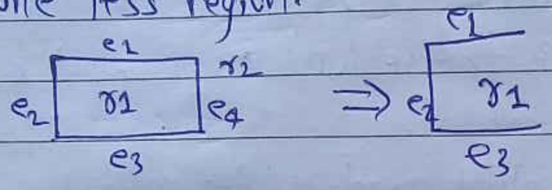
$= \cancel{n} - \cancel{n} + 1 + 1 = 2$

$v = 4, e = n - 1 = 3$

$r = 1$

$\therefore v - e + r = 2$

Case 2: Let G has finite cycle. It means it has finite regions. Let us choose a region that has a common edge with the outer region and then delete such an edge. Thus, the new graph has one less edge and one less region.



Original graph where $e=4$, $r=2$

After deleting one edge e4: $e=3=e-1$, $r=1=r-1$

$$v=v,\ e=e-1,\ r=r-1$$

Then, we have

$$~~v-e~~\ v-(e-1)+r-1=2$$

$$\text{or, } v-e+\cancel{1}+r-\cancel{1}=2$$

$$\therefore v-e+r=2$$

It completes the proof.

Theorem 2: If G is a planar graph with K components and r regions such that $|V|=v$ and $|E|=e$, then $v-e+r=k+1$

Proof

The result is obtained by applying the above theorem 4.1 to each component separately. The infinite region is counted only once to be noted well.

Theorem 3: If G is a connected planar graph with vertices v and edges e, where $v\geq 3$, then $e\leq 3v-6$.

Proof

For $v=3$, $e\leq 3$. It is happened because a connected graph with three vertices has at most three edges.

As there are at least 3 edges belonging to the boundary of every region, it gives $n \geq 3r$.

Thus, $3r \leq n \leq 2e$

or, $3r \leq 2e$ or, $r \leq \frac{2e}{3}$

or, $-r \geq -\frac{2e}{3}$

Again, we have

$v - e + r = 2$

or, $v = e - r + 2$

Thus,

$v = e - r + 2 \geq e - \frac{2e}{3} + 2$

or, $v \geq \frac{3e - 2e + 6}{3}$

or, $3v \geq e + 6$

Hence, $e \leq 3v - 6$.

It completes the proof.

Q1. Find the maximum number of edges in the planar graph with 9 vertices.

Soln:

$v = 9$

Now, $e < 3v - 6$

$\Rightarrow e \leq 3 \times 9 - 6$

$\Rightarrow e \leq 27 - 6$

$\Rightarrow e \leq 21$

Hence, the maximum number of edges possible in the planar graph with 9 vertices is 21.

Q2. Find the minimum number of vertices necessary for a graph with 21 edges to ~~the~~ be planar?

Soln:

We have,

$$v \geq \frac{e}{3} + 2$$

For, $e = 21$

$$v \geq \frac{21}{3} + 2 \Rightarrow v \geq 7 + 2$$

$$\Rightarrow v \geq 9.$$

Hence, the minimum number of vertices required is 9.

Theorem 4 (Kuratowski)

The graphs K_5 and $K_{3,3}$ are not planar.

(a) K_5 is not planar [or K_n is planar if & only if $n \leq 4$]

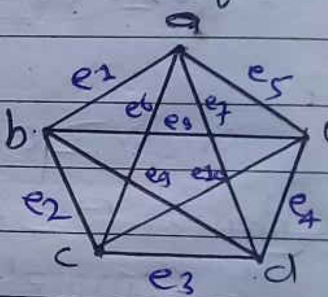
Proof

It is clear that K_5 has 5 vertices and 10 edges, so that: $3v - 6 = 9$

That is $e > 3v - 6$. So, the graph K_5 cannot be planar.

It completes the proof.

$[e \leq 3v - 6]$



K5 graph

(b) $K_{3,3}$ is not planar. [or, A complete bipartite graph $K_{m,n}$ is planar if and only if $m \leq 2$ or $n \leq 2$]

Proof

Note that $K_{3,3}$ is connected and has no triangles. $K_{3,3}$ has 6 vertices and 9 edges.

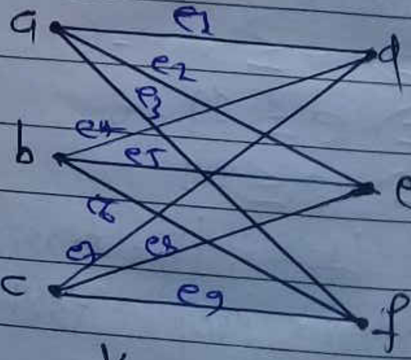
Obviously, $v \geq 3$ and there are no circuits of length 3.

If $K_{3,3}$ were planar, then $e \leq 2v - 4$ would have to be true

i.e. $2v - 4 = 2 \times 6 - 4 = 8$

i.e. $9 \leq 8$ (false) So e must be ≤ 8.

But $e = 9$.



K3,3 graph

So, $K_{3,3}$ is not planar.

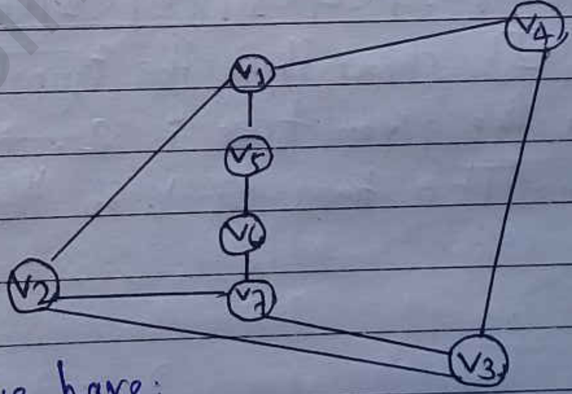
Welsh-Powell Algorithm to find Chromatic Number

Step 1: Find the degree of each vertex.
Step 2: List the vertices in order of descending degrees.
Step 3: Colour the first vertex with Color 1.
Step 4: Move down the list and Colour all the vertices not connected to the coloured vertex with the same Color.
Step 5: Repeat step 4 on all uncoloured vertices with a new colour in descending order of degrees, until all the vertices get coloured.

Examples:

1. In the given figure below, find the chromatic number of the graph G.

Soln:

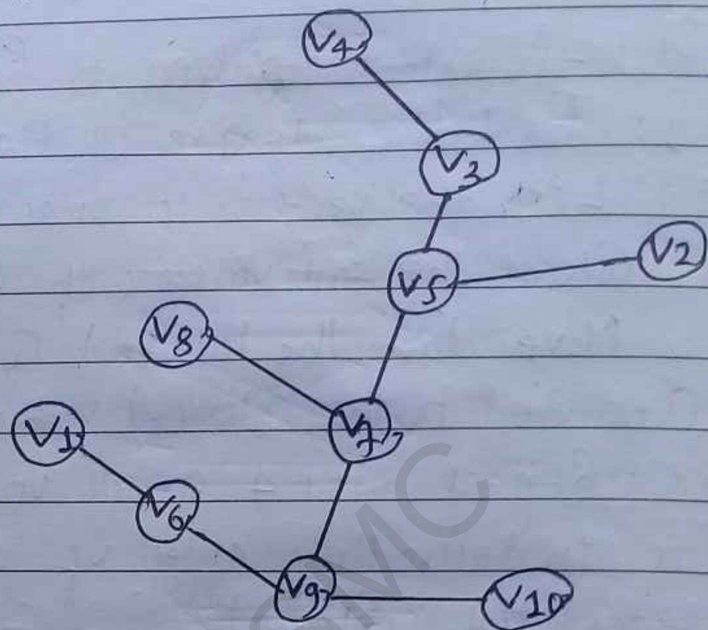


Following the algorithm, we have:

Vertex	V_7	V_3	V_2	V_4	V_1	V_5	V_6
Degree	4	4	3	3	3	3	2
Colour	C_1	C_2	C_3	C_3	C_2	C_1	C_3

The vertices V_2, V_3 and V_7 are connected to each other. So, they must be coloured by different colours. Here, at least 3 colours are required to paint the graph G. Thus, $\chi(G) = 3$.

2. In the given figure of tree (G), find its chromatic number.



Soln:

from the given figure we have

Vertex	V_5	V_7	V_9	V_3	V_6	V_1	V_2	V_4	V_8	V_{16}
Degree	3	3	3	2	2	1	1	1	1	1
Colour	C_1	C_2	C_1	C_2	C_2	C_1	C_2	C_1	C_1	C_2

It shows that the number of colours needed to colour the given tree is 2.

Note: The trees and so the forests are 2-colourable.

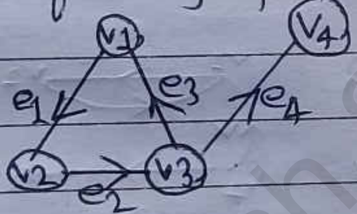
Unit - 5
Diagraphs and Traversability

5.1 Diagraph

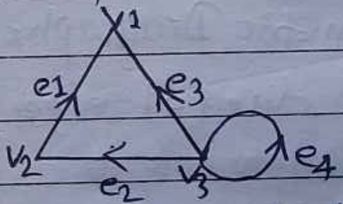
A directed graph (diagraph) is a graph that is made up of a set of vertices connected by edges (or arcs) where each edges have a direction associated with them.

In formal terms, a diagraph is an ordered pair $D=(V,A)$ consisting of two finite sets V and A where
- V is the set of vertices or nodes or points.
- A is the set of directed edges or arcs (sometimes E is used)

Example of diagraph:



Loop

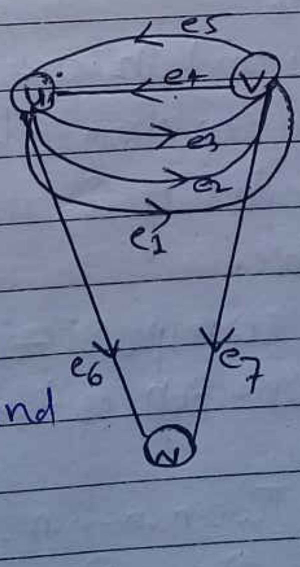


Having same initial & terminal vertex. [Loop $e_4=(v_3,v_3)$]

Parallel edges or Arcs:

Arcs having same initial and terminal vertices.

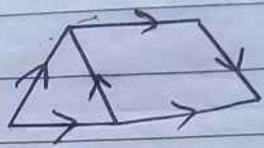
In the diagraph, $e_1=(u,v)$, $e_2=(u,v)$, $e_3=(u,v)$, $e_4=(u,v)$ and $e_5=(u,v)$.



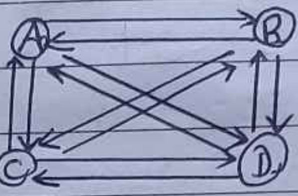
प्रश्न न.

Types of Diagraphs:

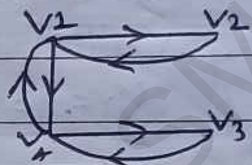
1. Simple Diagraph
   Having no loops ~~..~~ and no parallel edges.



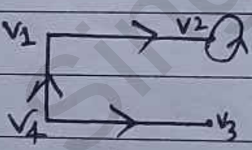
2. Complete Diagraph
   Each ~~edge~~ edge or arc is bidirected.



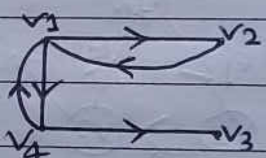
3. Symmetric Digraph
   For every edge (a,b) there is also an edge (b,a).



4. Asymmetric Diagraphs
   Have directed edges and allowed to have a self loop.



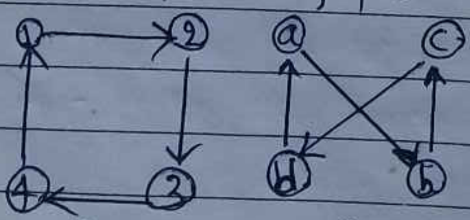
(A)



(B)

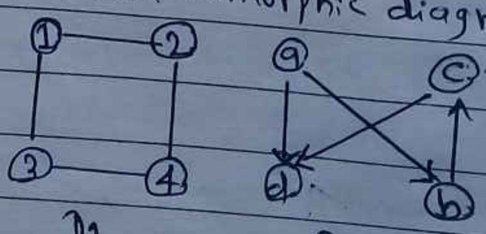
5. Isomorphic Diagraphs
   Two diagraphs D_1 and D_2 are said to be isomorphic if both of the following conditions hold:
   (i) The underlying graphs of D_1 and D_2 are either identical or isomorphic.
   (ii) Under one-to-one correspondence between the edges of D_1 and D_2 the direction of the corresponding edges are preserved.

Two isomorphic diagraphs



D_1 D_2

Two non-isomorphic diagraphs



D_1 D_2

5.2 Relation

Let A and B be two non-empty sets. A relation R from A to B is called a subset of $A \times B$.

i.e. If $R \subseteq A \times B$ and $(a,b) \in R$, we say that a is related to b by R, which is written as aRb.

- If a is not related to b by R, we write $a \not R b$.
- If $R \subseteq A \times A$, we say R is a relation on A.

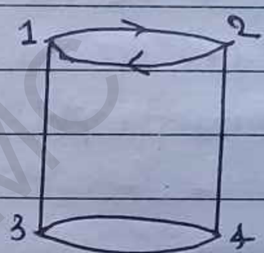
Example 1

In the given figure,

$V = \{1, 2, 3, 4\}$ and

$R = \{(1,2), (2,1), (1,4), (2,3), (4,3), (3,4)\}$

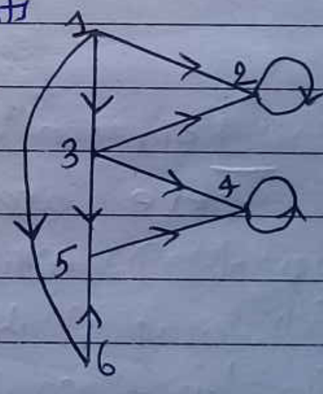
$R \subseteq V \times V$.



Example 2: Determine the relation in a given digraph.

As there are no parallel arcs, it represents a relation defined by:

$R = \{(1,2), (2,2), (1,3), (3,2), (3,4), (4,4), (3,5)$
$(5,4), (6,5), (1,6)\}$



Example 3

Let $A = \{1,2,3\}$ and $B = \{r, s\}$. Then $R = \{(1,r), (2,s), (3,r)\}$ $\subseteq A \times B$ is relation from A to B.

Equivalence Relation

A relation R on a set V is called an equivalence relation if it is reflexive, symmetric and transitive (all 3 are needed).

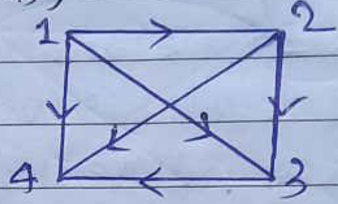
(a) Reflexive relation, if $(u,u) \in R$ for all $u \in V$

(b) Symmetrical relation, if $(u,v) \in R \Rightarrow (u,v) \in R$

(c) Transitive relation, if $(u,v) \in R$ and $(v,w) \in R \Rightarrow (u,w) \in R$

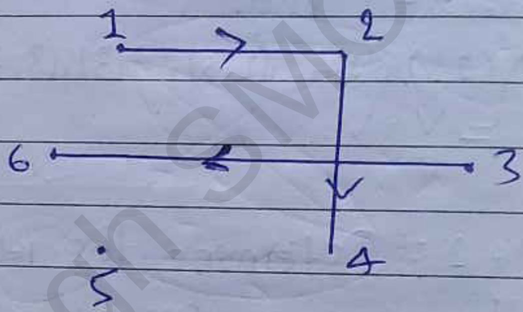
Q1. Draw a digraph D based on the relation R defined as "x is less than y" on V={1,2,3,4} where R={(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)}

Soln:



Q2. Let D be a digraph having V={1,2,3,4,5,6}. Find the ordered pairs of R where "y is double of x". Also draw graph of R.

Soln:



5.3 Matrix Representation

There are 2 principal ways to represent a digraph D with the matrix. They are:

1. Adjacency Matrix representation
2. Incidence matrix representation

1 Adjacency Matrix Representation

The adjacency matrix of a digraph 'D' having 'n' vertices is an $n \times n$ matrix, $A(D) = [a_{ij}]_{n \times n}$, where a_{ij} is the number of edges or arcs between V_i to V_j.

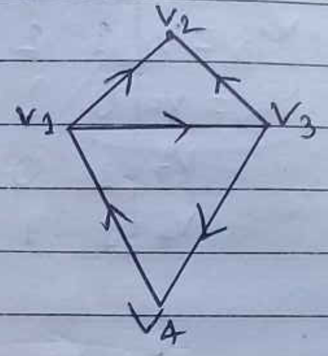
It is defined as:

$a_{ij} =$ 1, if $[V_i, V_j]$ is an edge i.e. V_i initial vertex and V_j final vertex.
0, if there is no edge/arc between V_i and V_j

Examples of Adjacency Matrix:

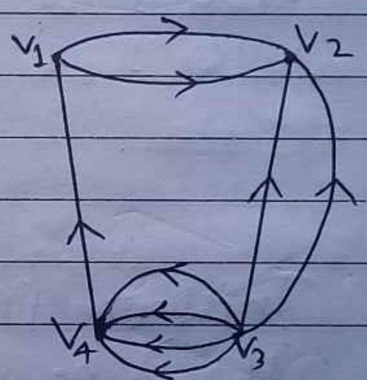
1. Represent the given diagraph by a matrix:

Here, the given diagraph D is represented by the square matrix AD as follows:



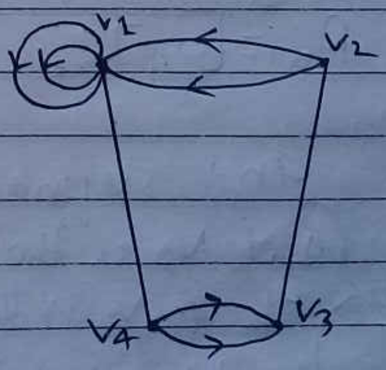
$$A_D = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array}\begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

2. Represent the given diagraph having parallel edges by adjacency matrix.



$$A_D = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array}\begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 4 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

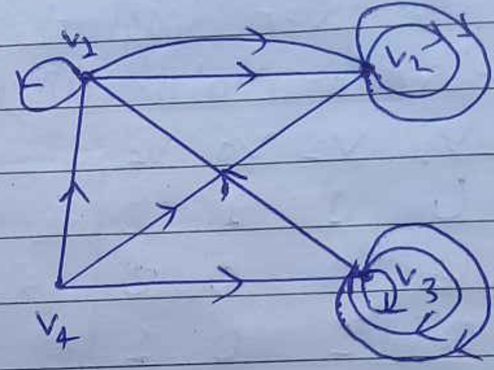
3. Represent the diagraph having loops and parallel edges by adjacency matrix.



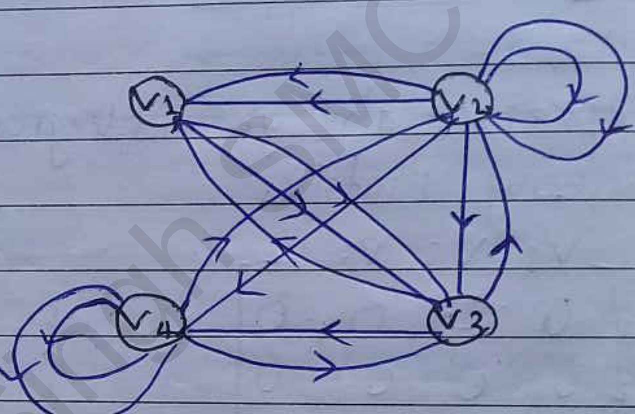
$$A_D = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array}\begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 2 & 0 \end{bmatrix} \end{array}$$

4. Draw the diagraphs from the following adjacency matrices.

(i) $M_D = \begin{array}{c|cccc} & v_1 & v_2 & v_3 & v_4 \\ \hline v_1 & 1 & 2 & 0 & 0 \\ v_2 & 0 & 2 & 0 & 0 \\ v_3 & 1 & 0 & 3 & 0 \\ v_4 & 1 & 1 & 1 & 0 \end{array} \Rightarrow$



(ii) $M_d = \begin{array}{c|cccc} & v_1 & v_2 & v_3 & v_4 \\ \hline v_1 & 0 & 0 & 3 & 0 \\ v_2 & 2 & 2 & 1 & 1 \\ v_3 & 0 & 1 & 0 & 1 \\ v_4 & 0 & 1 & 1 & 2 \end{array} \Rightarrow$



2. Incidence Matrix Representation

If D be a diagraph (with loops and parallel edges) having 'n' vertices and 'e' edges, then incidence matrix is $n \times e$ matrix,

$$A_D = [a_{ij}]_{n \times e} \quad \begin{bmatrix} n \text{ rows or } n \text{ vertices} \\ e \text{ column or } e \text{ edges} \end{bmatrix}$$

and defined as

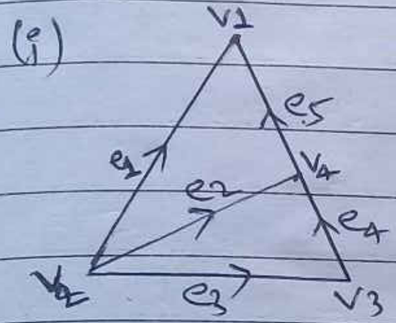
$$a_{ij} = \begin{cases} 1, & \text{if } v_i \text{ is the initial veartex of edge } e_j \\ -1, & \text{if } v_i \text{ is the final vertex of edge } e_j \\ 0, & v_i \text{ is not incident on edge } e_j \end{cases}$$

The number of ones in the incidence matrix is equal to the number of edges in the graph.

Examples

1. Find the incidence matrix of given diagraphs:

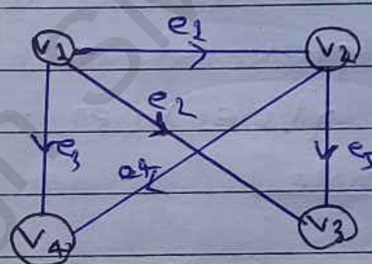
(i)



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

2. Construct a diagraph of following incidence matrix.

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



5.4 Connectivity of Digraphs

Underlying graph

The underlying graph G is obtained from the given diagraph D by deleting all the arrow heads (allocated directions) from the edges of D.

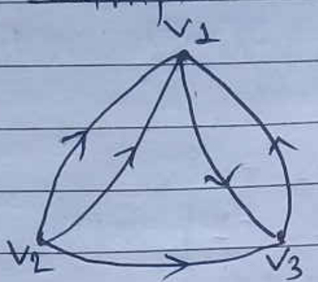
The underlying graph G of diagraph D has the following properties:

(a) G has same direction as that of D but without arrow-heads in the edges.

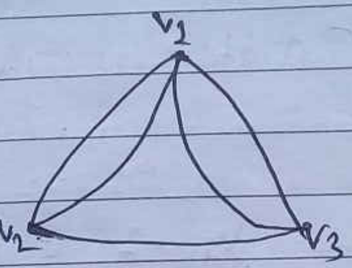
(b) G has same set of vertices as that of D.

(c) G has same set of edges which are connected in similar manner as that of D.

Example



(Diagraph)



(underlying graph)

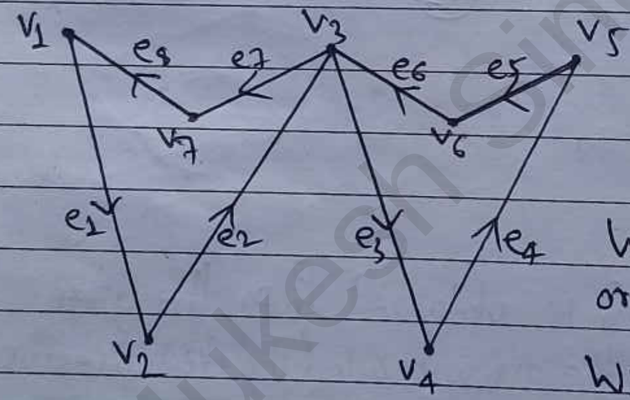
Directed walk

A directed walk W in a diagraph D is defined as a finite or infinite sequence of edges in the same direction which joins a sequence of vertices. It is represented as follows:

$W = \{v_1, e_1, v_2, e_2, \cdots, v_n\}$

Example

Length = no. of edges in walk path



$W_1 = \{v_1, e_1, v_2, e_2, v_3\}$ or $\{v_1, v_2, v_3\}$
Length, $L|W_1| = 2$

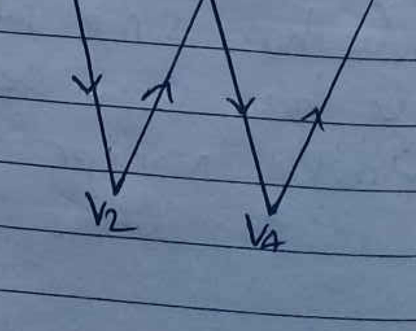
$W_2 = \{v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5\}$
or, $\{v_1, v_2, v_3, v_4, v_5\}$, $L|W_2| = 4$

$W_3 = \{v_1, v_2, v_3, v_4, v_5, v_6, v_3\}$, $L|W_3| = 6$

Directed Path

A directed path is defined as open directed walk (or directed trial) in which all vertices and edges are distinct.

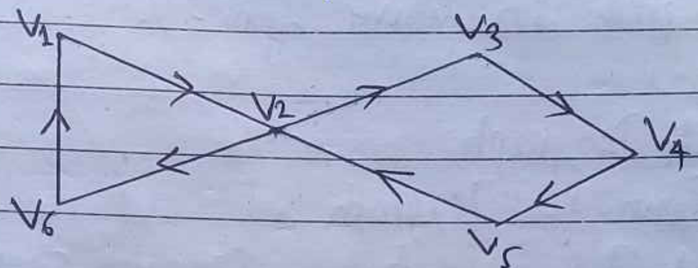
In the given figure,
$\{v_1, v_2, v_3, v_4, v_5\}$ is a directed path.



Directed trial

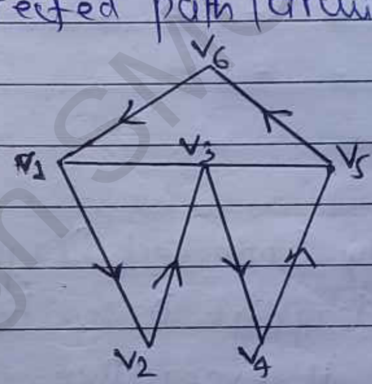
Directed walk in which all edges are distinct.

Directed Circuit → It is a directed closed walk/trial in which vertices may be repeated but edges are not repeated.



Here, {V1, V2, V3, V4, V5, V2, V6, V1} is a directed circuit.

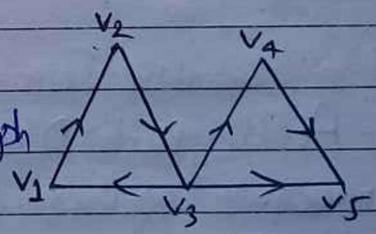
Directed Cycle → It is a directed path/circuit in which all vertices are different except first and last one. Actually, it is a closed directed ~~path~~ path.



In the given figure,
{V1, V2, V3, V4, V5, V6, V1} is a directed cycle.
Other examples are: {V1, V2, V3, V1} and {V3, V4, V5, V3}

Spanning path

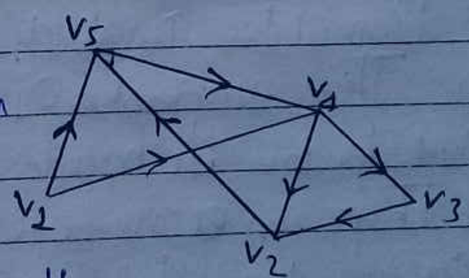
It is a directed walk ~~in~~ which contains all the vertices of a given diagraph D.



Here {V1, V2, V3, V4, V5} is a spanning path.

Reachable

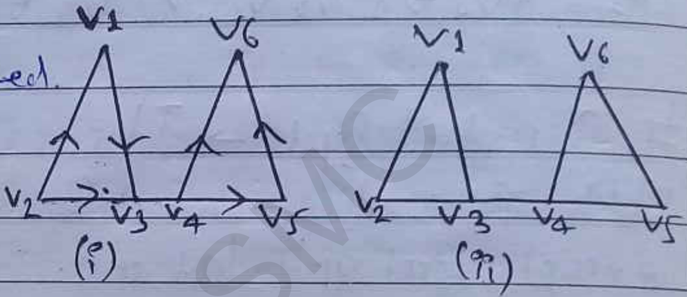
Let D be a diagraph in which a vertex V is said to be reachable from a vertex U if there is a directed path from U to V.



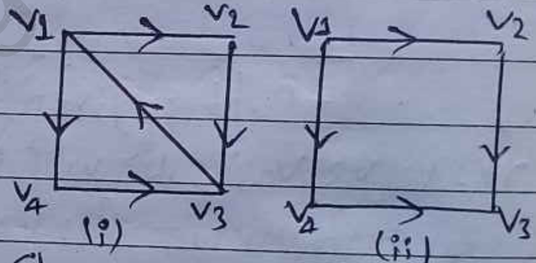
For example, the directed path for V1 to V5 is {V1, V4, V3, V2, V5}. Thus V5 is

reachable from V1.
But V1 is not reachable from V4 (check its directed path in above figure).

Connected Diagraph
(a) Weakly Connected Diagraph
A diagraph D is said to be strongly connected if its underlying graph is connected.
Otherwise, it is called disconnected.



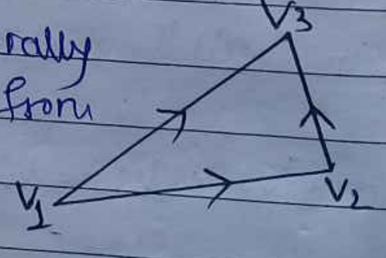
(b) Strongly Connected Diagraph
A diagraph D is strongly connected if every pair of vertices is reachable from one to another.
(i.e V_i to V_j and V_j to V_i)



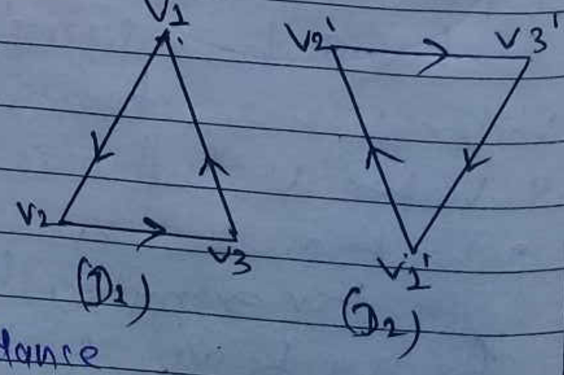
Strongly Connected

Non-Strongly Connected

(c) Unilaterally Connected graph
A diagraph D is said to be unilaterally connected if there is a directed path from V_i to V_j or V_j to V_i but not both the paths.



Isomorphic Diagraphs
The diagraphs D1 and D2 are said to be isomorphic if:
(i) There is one-to-one correspondence between V1 and V2.
(ii) One-to-one and onto correspondance



between E_1 and E_2.

This kind of correspondance is also called diagraph isomorphism.

For example, in the above figures D_1 & D_2:

$V_1 = \{v_1, v_2, v_3\}$ and $V_2 = \{v_1', v_2', v_3'\}$

$E_1 = \{(v_1, v_2), (v_2, v_3), (v_3, v_1)\}$

$E_2 = \{(v_1', v_2'), (v_2', v_3'), (v_3', v_1')\}$

Thus, the isomorphism (f) defined between D_1 and D_2 holds the following properties.

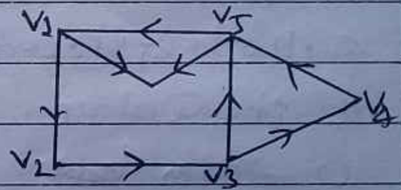
(i) $f(v_1) = v_1'$, $f(v_2) = v_2'$, $f(v_3) = v_3'$

(ii) $f(v_1, v_2) = (v_1', v_2')$, $f(v_2, v_3) = (v_2', v_3')$ and $f(v_3, v_1) = (v_3', v_1')$

5.5 Traversability of diagraphs

Adjacent Vertices

Let $e_m = (v_i, v_j)$. The vertices v_i and v_j of the edge e_m is called adjacent if e_m comes out from v_i and directly enters into v_j.

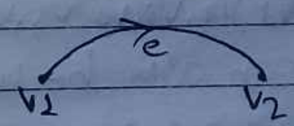


For example, in the figure, (v_4, v_5), (v_1, v_6), (v_1, v_2) (v_3, v_5) are adjacent vertices.

But (v_1, v_2) or (v_2, v_6) are not adjacent vertices pairs.

Incident out and Incident into edges

Here, an edge e has been incident out from initial vertex v_1 and incident into terminal vertex v_2.

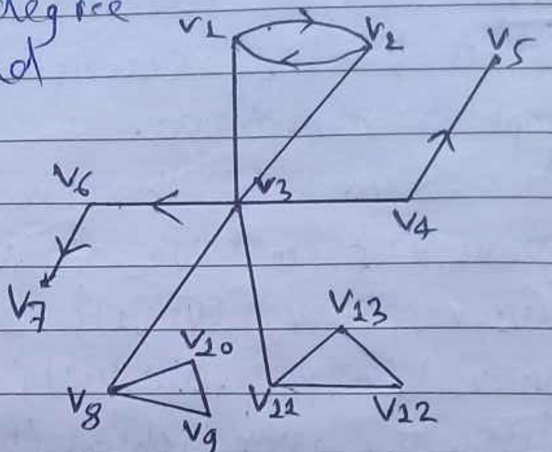


Out degree and In-degree of the Vertex:

The number of edges incident out from the given vertex is called out-degree of it.

Similarly, the number of edges incident into the given vertex is called in-degree of it.

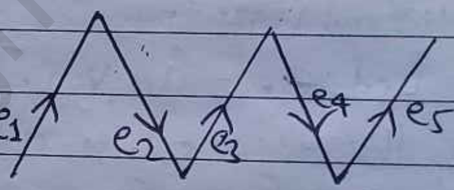
The outdegree and indegree are denoted by $od(v_i)$ and $id(v_i)$ respectively.



Vertex	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}
Out-degree	2	2	2	1	0	1	0	3	0	1	1	1	2
In-degree	1	1	4	1	1	1	1	0	2	1	2	1	0

Directed Euler trial

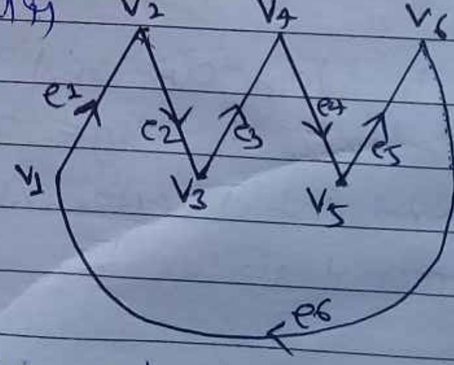
A directed walk in D is called a directed Euler trial if it contains all the arcs or edges ~~bet~~ of D exactly once.



For example a directed Euler trial is given by $\{e_1, e_2, e_3, e_4, e_5\}$.

Direct Euler tour

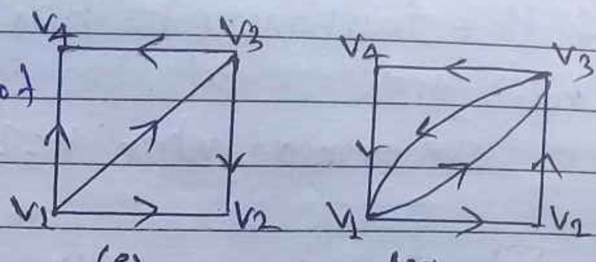
A closed directed Euler trial is called a directed Euler tour in the diagraph D. For example: $\{v_1, v_2, v_3, v_4, v_5, v_6, v_1\}$ which ~~contains~~ can also be written as $\{e_1, e_2, e_3, e_4, e_5, e_6\}$ or $\{v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_5, e_5, v_6, e_6, v_1\}$.



Eulerian Diagraph

An Eulerian diagraph is defined as a diagraph D which contains a directed Euler tour.

For example, in the given figure, the first diagraph is not Eulerian but second one is an Eulerian diagraph.



(i) (ii)

Applications:

5.6 Tournaments Management

A tournament is a diagraph in which each pair of vertices is connected by one directed arc.

A tournament in which every players (or teams) plays against every other player once is called Round-Robin tournament.

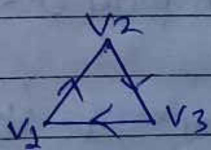
For example, let v1 and v2 be two players where v1 won the game and v2 lost it, then we ~~have~~ represent it in tournament as given alongside.

Note

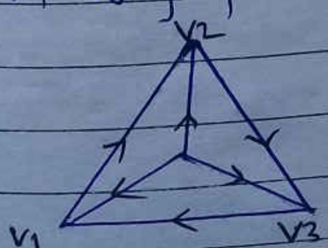
1. If we provide directions to the arcs of a complete diagraph, it gives a tournament.
2. There are different tournament patterns which are given below:

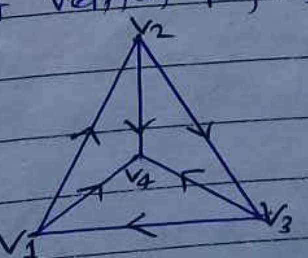
(i) A single tournament is represented by two teams:

(ii) Two tournament in the diagraph is represented with 3 vertices.



(iii) A diagraph having 4 vertices represents 4 tournaments.





3. Tournament is unilaterally connected.

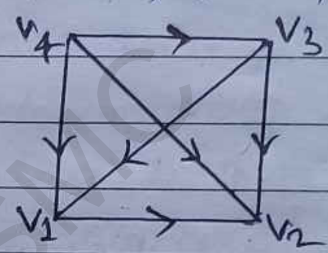
4. The tournament has n vertices consists of $\frac{n(n-1)}{2}$ no. of arcs.

For example, when $n=2$, then it has 1 arc and so on.

5. The score of vertex (V) in the tournament T means the number of out-degree of v. It is denoted by $S(V)$.

6. Scores of vertices (teams) are calculated as follows:

Scores	Won	Lost
$S(v_1) = 1$	1	2
$S(v_2) = 0$	0	3
$S(v_3) = 2$	2	1
$S(v_4) = 3$	3	0



Hence the sequence of scores in ~~the~~ ~~given~~ descending order is: 3, 2, 1, 0.

Q. Find the number of tournaments for n teams.

Let $t_1, t_2 \ldots t_n$ be the playing teams.

Here t_1 can play $(n-1)$ no. of games with $t_2, \ldots t_n$ teams.

t_2 can play $(n-2)$ no. of games with $t_3 \ldots t_n$ teams.

⋮

t_n can play $[n-(n-1)] = 1$ no. of games with t_n team.

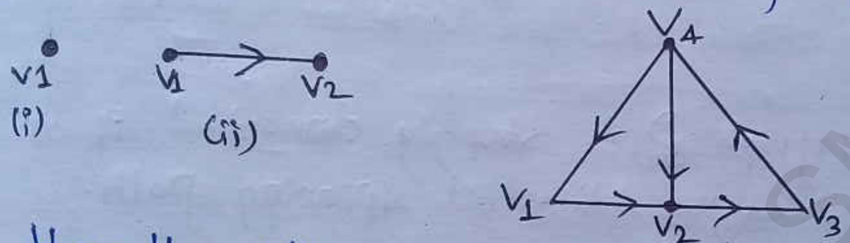
Thus, there are $(n-1)+(n-2)+\cdots+1 = \frac{n(n-1)}{2}$ tournaments.

Hamiltonian Path and Cycle

Let D be a diagraph and P be directed path in it. P is called directed Hamiltonian path if it includes every vertex of D exactly once.

Similarly a directed cycle C in D is called Hamiltonian cycle if C includes all the vertices of D exactly once except the initial and final vertex in which cycle comes to be end.

For example see the following diagraphs:



Here the paths $p_1 = v_1$;
$p_2 = \{v_1, v_2\}$ and
$p_3 = \{v_1, v_2, v_3, v_4\}$ are Hamiltonian paths.

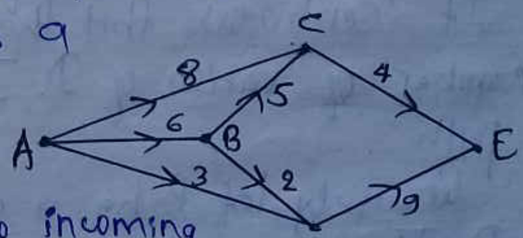
And $C = \{v_1, v_2, v_3, v_4, v_1\}$ is a Hamiltonian cycle.

Note: The diagraph having a directed Hamiltonian cycle is called Hamiltonian.

5.7 Traffic Flow System

The traffic flow system can be re-arranged by changing two-way flow into one-way flow so that it is more systematic, easy and comfortable.

A transportation network is a weighted directed graph satisfying following conditions:



i) There is only one vertex with no incoming edges, called the source (i.e. vertex A).

ii) There is only one vertex with no outgoing edges, called destination/sink (i.e. vertex E).

iii) The weight assigned to each edge is a non-negative number, called its capacity.

In the above graph:

vertices → represents stations or crossroads or intersection of roads.

edges → represents lines or roads or streets

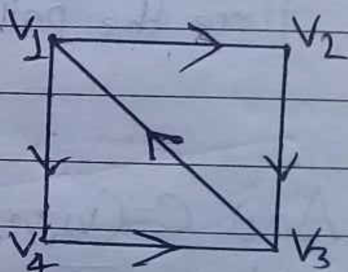
Capacity → is the maximum amount of flow possible per unit of time through that line.

This system follows the concept of orientation or direction provided to the edges of the graph.

Theorems

Theorem 1: A diagraph D is strongly connected if and only if it has a closed directed spanning path.

Proof:



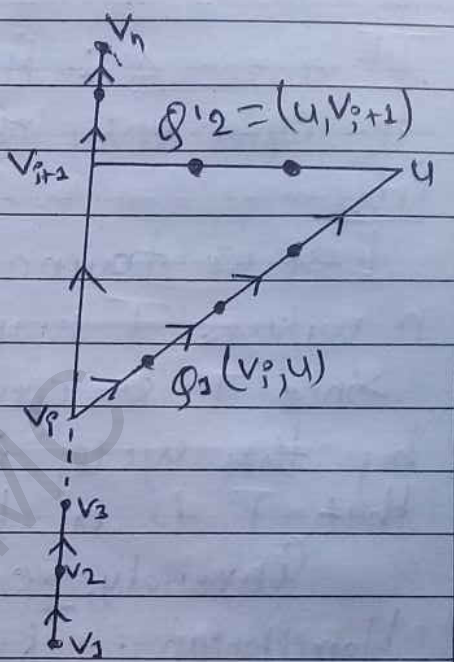
Strongly Connected Diagraph

Let us consider a diagraph D is strongly connected. Let $P = \{V_1, V_2, V_3, V_1\}$ be a closed path containing a maximum number of vertices of diagraph D.

If possible, we take P is not a spanning path. If so, there is a vertex say V' not involved in the closed path P. But D is a strongly connected diagraph, so there are paths $P' = \{V', \ldots V_1\}$ and $Q' = \{V_1, \ldots V'\}$. It implies that the path $(P' \cup Q')$ is a closed path involving the vertices of closed path P and the vertex V'. It contradicts that the closed path P consists of maximum number of vertices of D. It means P is a closed spanning path.

Conversely, let P be a closed spanning path in the diagraph D. Then every vertex of D can be reached from any vertex of D through the path P. It implies that D is strongly connected.

Theorem 2: A diagraph D is unilaterally connected diagraph if and only if there is a directed spanning path in D.



Let $P = \{v_i, v_{i+1}\}$ or $P' = \{v_{i+1}, v_i\}$ is a directed spanning path. Thus, the number of vertices in the directed spanning path and the diagraph D have the same. Therefore, every $\{v_i, v_{i+1}\}$ are directly connected by any small paths of P.

It completes the theorem.

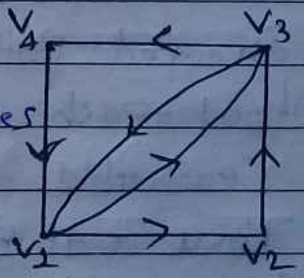
Theorem 3: Let D be a diagraph with vertices $v_1, v_2 \cdots v_n$ and q be the number of directed arcs in D, then

$$\sum_{i=1}^{n} id(v_i) = \sum_{i=1}^{n} od(v_i) = q$$

OR

A weakly connected diagraph D is with at least one arc is Eulerian iff $od(v) = id(v)$ for every vertex v of D.

Proof



Each arc is counted once when it enters into a vertex. Thus, the total no. of indegrees is equal to total number of directed arcs;

i.e. $\sum_{i=1}^{n} id(v_i) = q$ --- (1)

Similarly, each arc is counted exactly once when it goes out from each vertex. Thus, the total number of out-degrees is equal to total number of directed arcs.

i.e. $\sum_{i=1}^{n} od(v_i) = q$ -- (2)

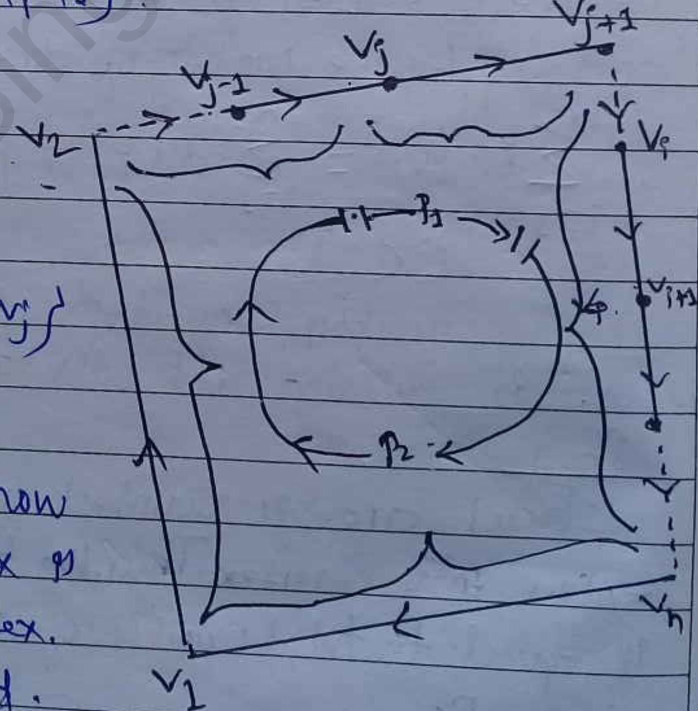
Hence, from ① & ②, $\sum_{i=1}^{n} id(v_i) = \sum_{i=1}^{n} od(v_i) = q$.

Theorem 4: A tournament T is a Hamiltonian if and only if it is strongly connected.

Proof

Let a tournament T is strongly connected with n vertices. T must have a directed cycle of length n'. Since T consisting all the vertices are used by this cycle, it is a Hamiltonian cycle. It implies that T is a Hamiltonian.

Conversely, we assume that the tournament T is Hamiltonian. If so, it consists of Hamiltonian cycle given as $C = \{v_1, v_2, \dots, v_n, v_1\}$.



From the figure,
We have

$P_1 = \{v_j, v_{j+1} \dots v_i\}$

$P_2 = \{v_i, v_{i+1} \dots v_n, v_1, v_2, \dots v_{j-1}, v_j\}$

Both P_1 and P_2 are directed paths and they show that each and every vertex is reachable from either vertex. Thus, T is strongly connected. Hence, the theorem is completed.

Unit-6
Matchings

6.1 Matching

Matching is a set of edges (without self loop) without common vertices.

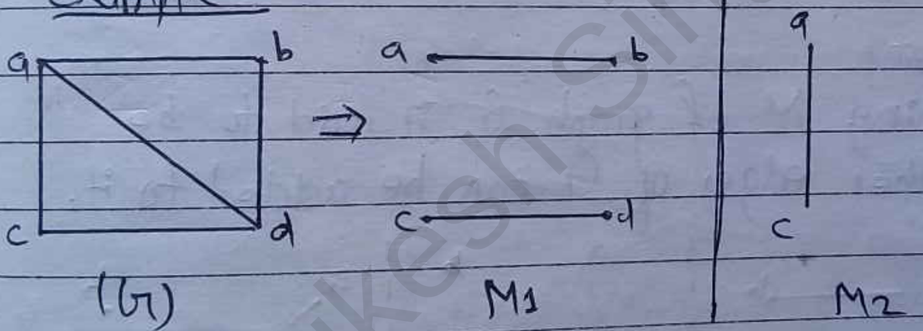
Mathematical definition

Let $G = (V, E)$ be a graph. A subgraph M is called matching if each vertex of G is incident with at most one edge in M.

i.e. $\deg(v) \leq 1$

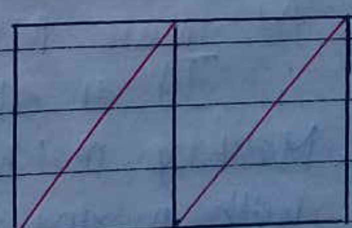
Which means in the matching M, the vertices should have degree 1 or 0, where the edges should be incident from the graph G.

Example



Note

1) The size of matching is given by the number of edges in the matching. for example the size of matching of the given graph G is 2.
2) A vertex V which is incident with an edge in the set all said to be saturated.
3) In the given graph there is a matching M of size 2 which has saturated 4 vertices.



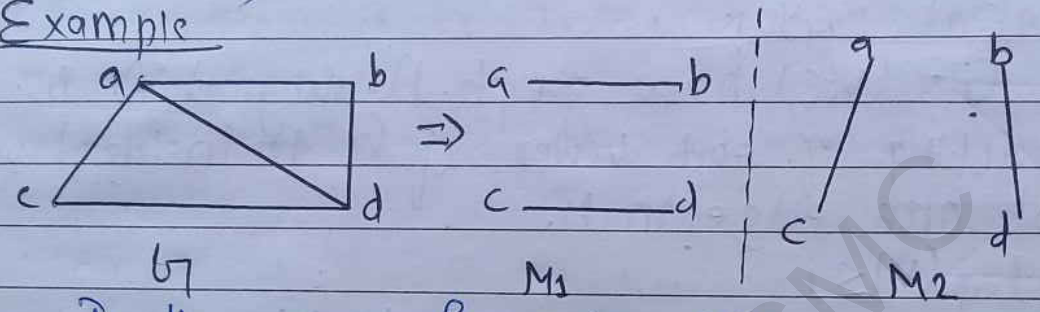
प्रश्न नं. उत्तर

6.2 Perfect Matching

A matching M in G is called a perfect matching if M saturates all vertices of G i.e. if and only if every vertex of G is incident to exactly one edge of the matching set.

i.e. $\deg(v) = 1$

Example

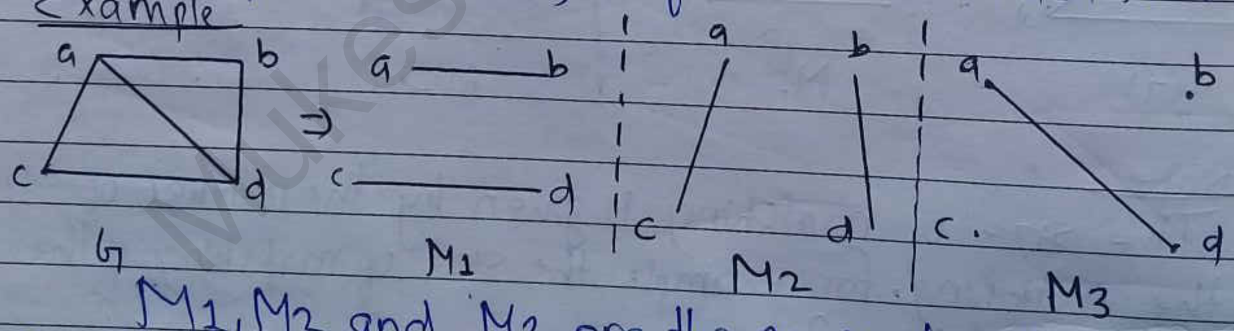


In the given figure M_1 & M_2 are examples of perfect matching which has saturated all 4 vertices of G.

6.3 Maximal Matching

A matching M of graph G is said to be maximal if no other edges of G can be added to it.

Example



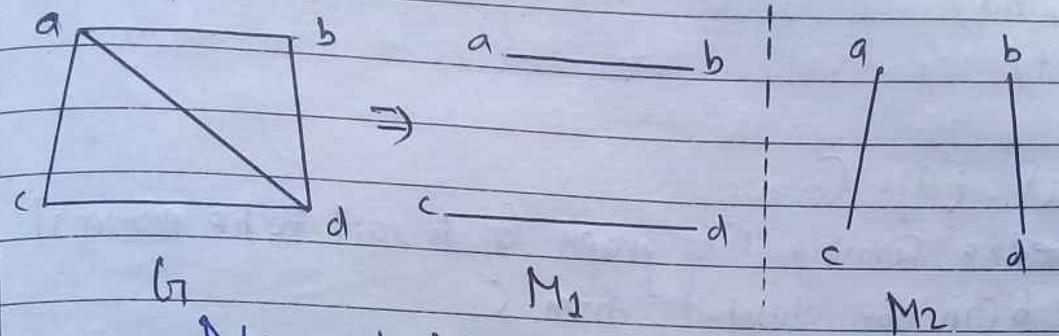
M_1, M_2 and M_3 are the maximal matching of G.

6.4 Maximum Matching

It is also known as largest maximal matching. Matching matching is defined as the maximal matching with maximum number of edges.

The number of maximum edges in the maximum matching is called its matching number.

Example



M_1 and M_2 are maximum matching of G. We can provide maximum two edges to maintain matching-ness. Hence, we have matching number of 2.

Note

Every perfect matching is a maximum matching but not every maximum matching is a perfect matching.

6.5 Covering

A graph covering 'C' of a graph 'G' is a sub-graph of G which contains either all the vertices or the edges corresponding to some other graph.

A sub-graph which contains all the vertices is called a line/edge covering.

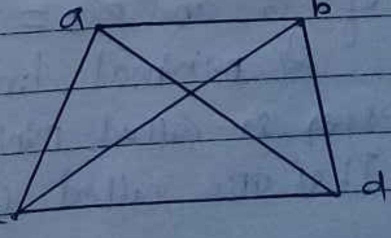
A sub-graph which contains all the edges is called a vertex covering.

1. Edge Covering

A set of edges which covers all the vertices of a graph G is called line covering or edge covering of G.

Edge covering of a graph G with n vertices has at least $n/2$ edges.

Every vertex of G is incident with at least one edge in C i.e $d_G(v) \geq 1$



Example Here, $G = \{a, b, c, d\}$

Set of coverings are:

$C_1 = \{ab, cd\}$

$C_2 = \{ad, bc\}$

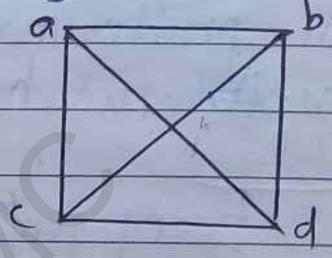
$C_3 = \{ab, bc, bd\}$
$C_4 = \{ab, bc, cd\}$

a) Minimal Edge Covering

An edge covering C of graph G is said to be minimal if no edge can be deleted from C.

- No minimal edge covering contains a cycle.

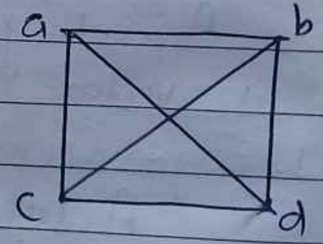
Example



From the graph, edge coverings are:
$C_1 = \{ab, cd\}$
$C_2 = \{ad, bc\}$
$C_3 = \{ab, bc, bd\}$
$C_4 = \{ab, bc, cd\}$
$C_5 = \{ab, ad, cd\}$

Here, C_1, C_2 and C_3 are minimal edge coverings but C_4 and C_5 are not because we delete bc & ad from C_4 & C_5 respectively.

b) Minimum Edge Covering



From the given graph, edge coverings are:
$C_1 = \{ab, bc\}$
$C_2 = \{ad, bc\}$
$C_3 = \{ab, bc, bd\}$
$C_4 = \{ab, bc, cd\}$
$C_5 = \{ab, ad, cd\}$

Here, C_1 and C_2 are the minimum edge coverings of G and $\alpha_1 = 2$.

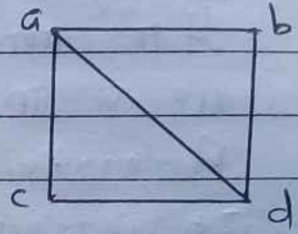
A minimal line covering with minimum number of edges is called minimum line/edge covering of graph G. It is also called smallest minimal line covering.

2. Vertex Covering

A set of vertices which covers all the nodes/edges of a graph G, is called a vertex Covering.

Example

From the graph vertex coverings are :



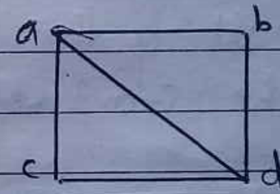
$C_1 = \{a, d\}$
$C_2 = \{a, b, c\}$
$C_3 = \{b, c, d\}$
$C_4 = \{a, d, b\}$

a) Minimal Vertex Covering

A vertex covering C of graph G is said to be minimal if no vertex can be deleted from C.

Example

From the graph, vertex coverings are:



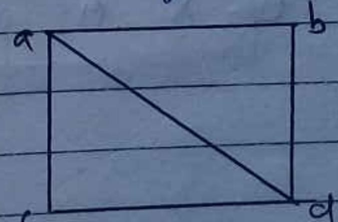
$C_1 = \{a, d\}$
$C_2 = \{a, b, c\}$ Here, C1, C2 and C3 are minimal
$C_3 = \{b, c, d\}$ vertex covering, but C4 is not because b can
$C_4 = \{a, d, b\}$ be deleted.

~~Her~~

b) Minimum Vertex Covering

A minimum vertex Covering is called when minimum number of vertices are covered in a graph G. It is also called smallest minimal vertex covering.

The number of vertices in a minimum vertex covering in a graph G is called the vertex covering number of G and it is denoted by α_2.



Example : Vertex coverings are:

$C_1 = \{a, d\}$
$C_2 = \{b, c, d\}$ Here, C1 is minimum vertex
$C_3 = \{b, c, d\}$ Covering of G, as it has only two vertices.
$C_4 = \{a, d, b\}$ $\therefore \alpha_2 = 2$

6.6 Matching in Bipartite Graph

Matching in a bipartite graph is a set of edges for which every vertex belongs to exactly one of the edges.

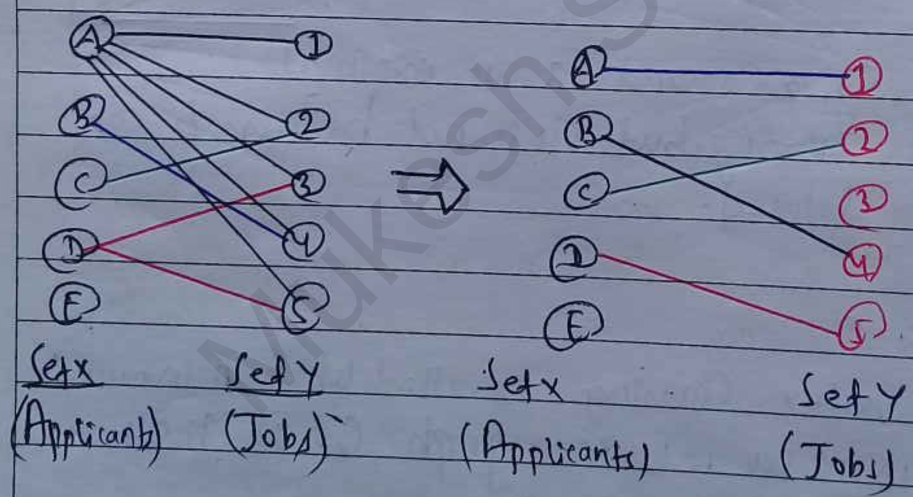
It is arranged in such a way that no two edges share a single vertex.

A maximum matching is a matching of maximum size (maximum number of edges).

There can be more than one maximum matching for a given Bipartite graph.

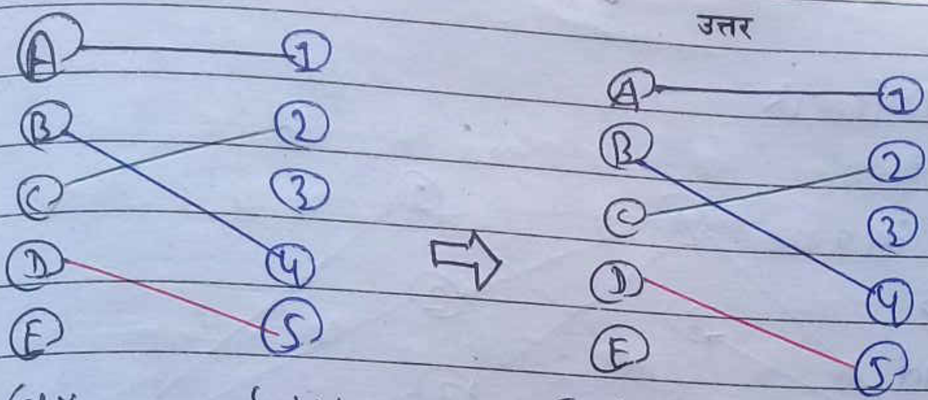
Example of general matching in bipartite graph

A matching is one-to-one pairing of some or all of the elements of one set X, with the element of the second set Y.



Example of Complete Matching in bipartite graph

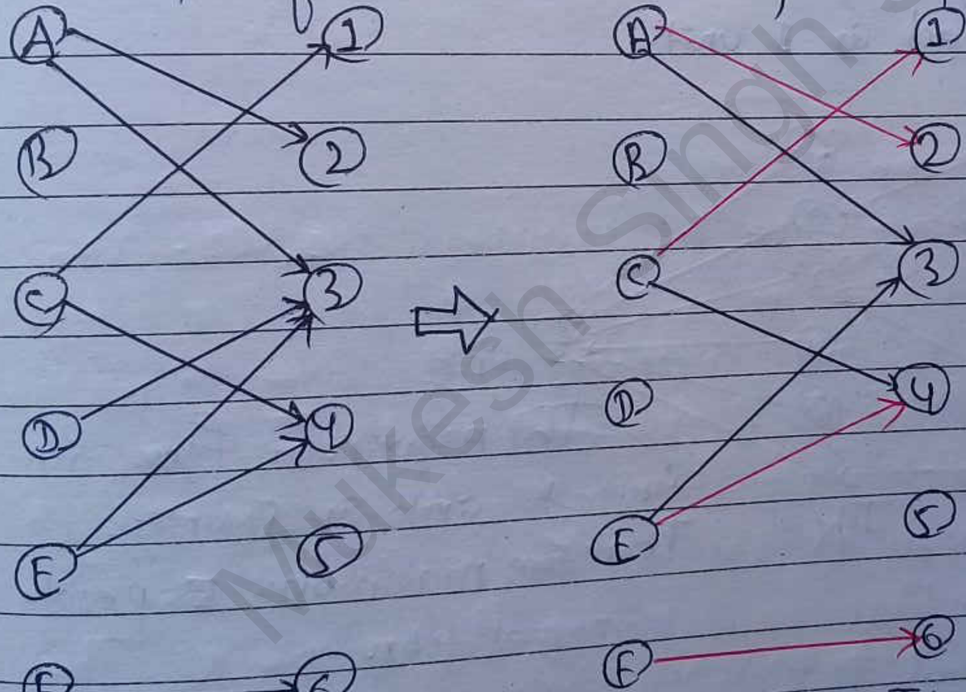
A complete matching is when every member of X is paired with one member of Y.



Set X (Applicants) Set Y (Jobs)
Incomplete Matching

Set X (Applicants) Set Y (Jobs)
Complete matching

Example of Maximum matching in bipartite graph:



So, maximum 5 people can get jobs

Applicants (X) Jobs (Y)

Applicants (X) Jobs (Y)
Maximum matching

Maximum bipartite matching using max flow problem:

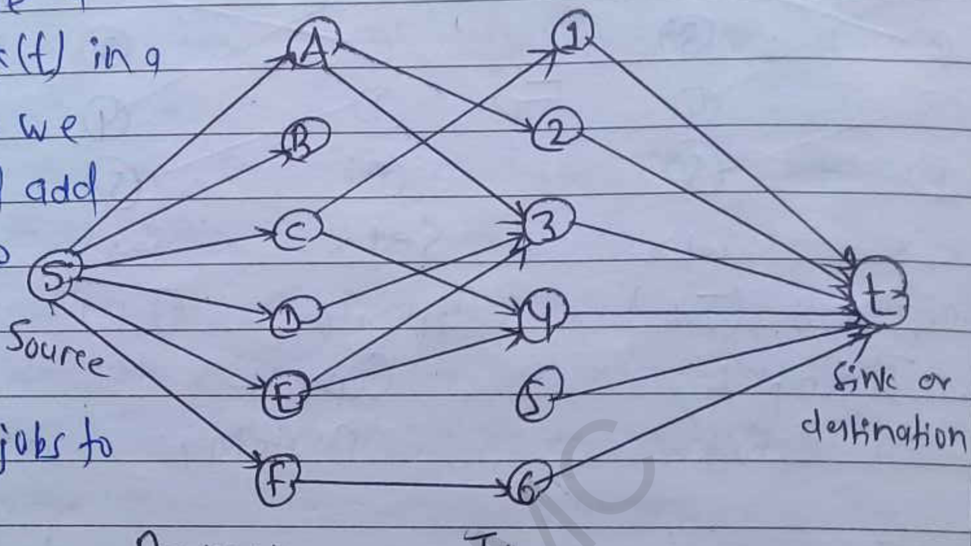
Maximum bipartite matching (MBP) problem can be solved by converting it into a flow network.

प्रश्न नं. उत्तर

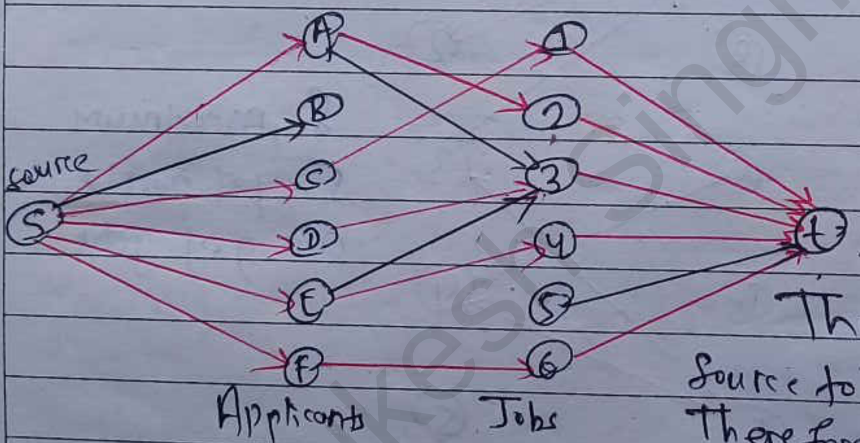
1) Build a flow network

There must be a source (s) and sink (t) in a flow network. So we add a source and add edges from source to all applicants.

Similarly, add edges from all jobs to sink.



The capacity of every edges is marked as 1 unit.



The maximum flow from source to sink is 5 units. Therefore maximum 5 people can get jobs.

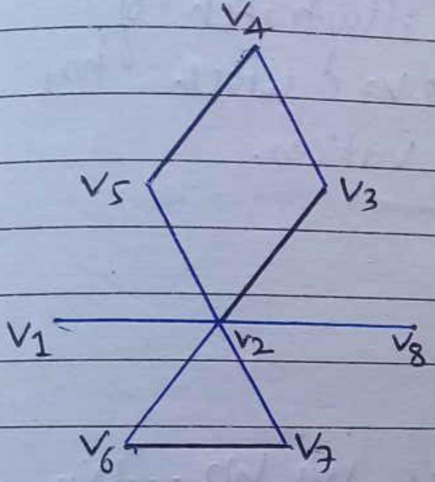
2) Find the maximum flow

We use Ford-Fulkerson Algorithm to find the maximum flow in the flow network built in step 1. The maximum flow is actually the MBP we are looking for.

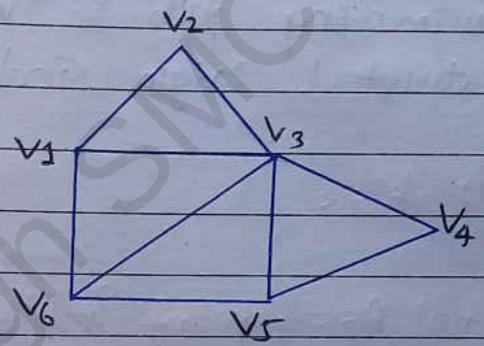
6.7 ## M-alternating path

Let $G = \langle V, E \rangle$ be a graph and M its matching. A path P in G is said to be an M-alternating path if its edges are alternatively in E−M and M.

For example, a path $P = \langle v_5, v_2, v_3, v_4, v_5 \rangle$ is an M-alternating path in the given figure (a), where P has travelled through the edges of matching and non-matching parts.



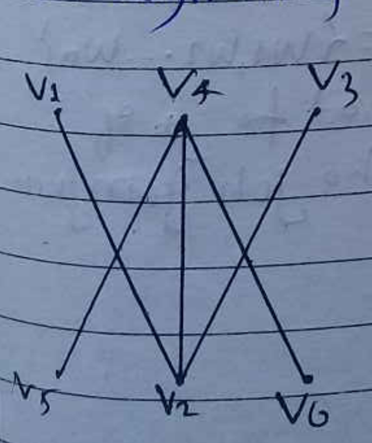
(a)



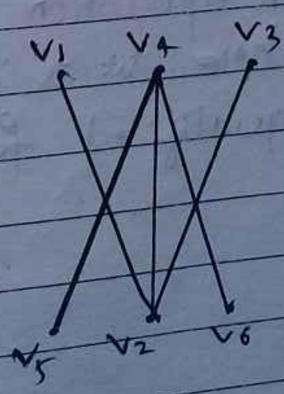
(b)

An M-~~altering~~ alternating path is defined as an ~~M-altering~~ M-alternating path whose origin and terminals are both M-unsaturated. In the above figure (b), the path: $P = \langle v_2, v_1, v_3, v_6, v_5, v_4 \rangle$ is an M-augmenting path where v_2 & v_4 are M-unsaturated.

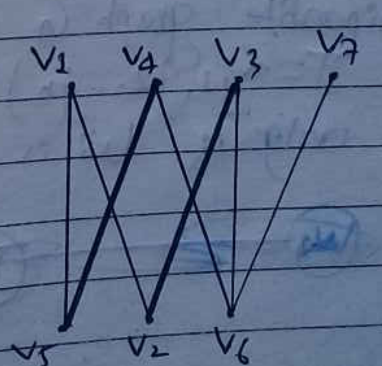
The other examples of M-~~al~~ alternating and M-augmenting path are given below in figure (b) and c.



(a)



(b)



(c)

The figure (a) is about a matching M in a bipartite graph G where G has been bipartitioned into two graphs $G_1 = \{v_1, v_2, v_3\}$ and $G_2 = \{v_4, v_5, v_6\}$. Here, matching in M with size 2.

The figure (b) is about ~~the~~ to show M-alternating path $P = \{v_5, v_4, v_2, v_3\}$ where it has passed through the edges of matching M and E-M edges.

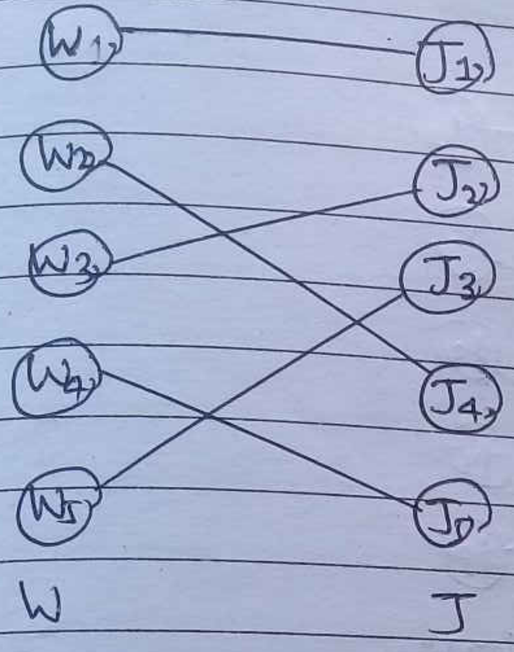
Similarly, the figure (c) is for illustration of M-augmenting path $P = \{v_1, v_5, v_2, v_3, v_6, v_7\}$ which has unsaturated origin and terminus vertices.

Applications

6.8 Personal Assignment Problem

Let there are 'n' workers $w_1, w_2, \ldots w_n$ and m jobs $j^1, j^2 \ldots j^m$ available in any company. The speciality of all workers is that each worker is qualified for at least one of the jobs. Then a question is raised like; is it possible to assign one job for each worker as per his/her qualification. This is the problem known by the name of "The personal assignment problem".

In order to solve this problem, we need to construct a bipartite graph G with bipartition $W = \{w_1, w_2 \ldots w_n\}$ and $J = \{j_1, j_2 \ldots j_n\}$ where, w_i is joined to j_k if and only if w_i is qualified for the job ~~jk~~ j_{km}.

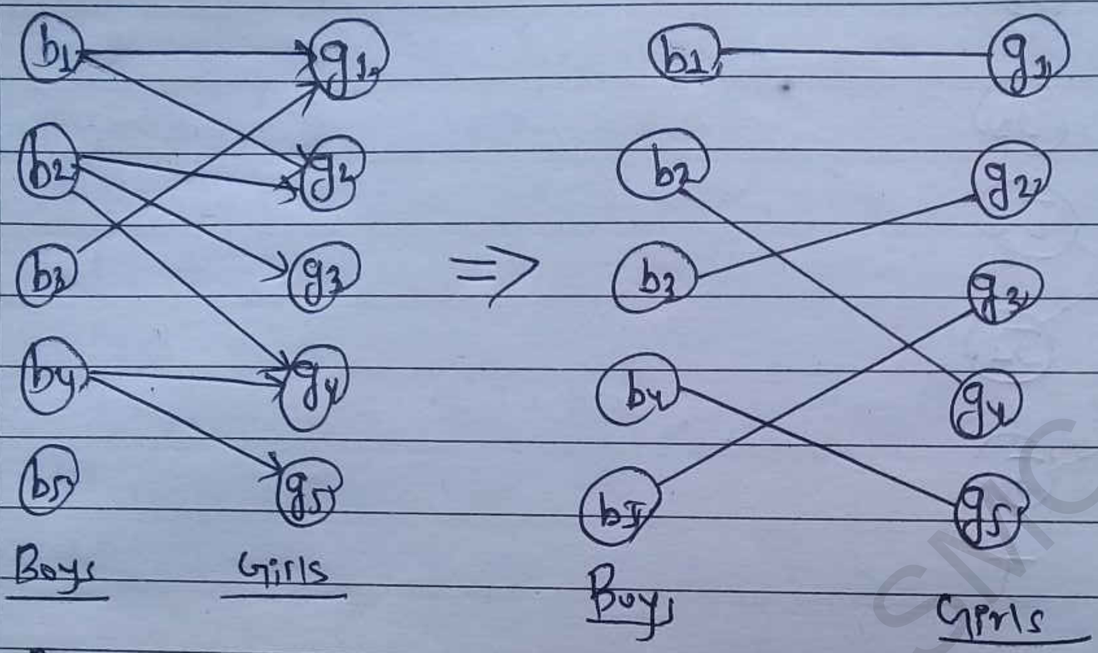


6.9 The Marriage Problem

We assume that there are n boys and m girls in a city for the marriage purpose. Let $B = \{b_1, b_2 \ldots b_n\}$ be a set of n boys and $G = \{g_1, g_2 \ldots g_n\}$ be a set of m girls. Suppose each boy ~~marries~~ has one or more girl friends, the problem is that under what condition we can arrange their marriage in such a way that each boy marries one of his girl friends. Such problem is called, "The Marriage Problem."

For the graph-theoretical formulation of this problem, we let G be the bipartite graph with the bi-partition $\{B, G'\}$ such that b_i is joined to g_j if and only if g_j is a girlfriend of b_i. The marriage is equivalent to finding the conditions under which G has a matching that saturates every vertex of G.

Example figure



Befor Matching After Matching