③

Chapter

Page No.:

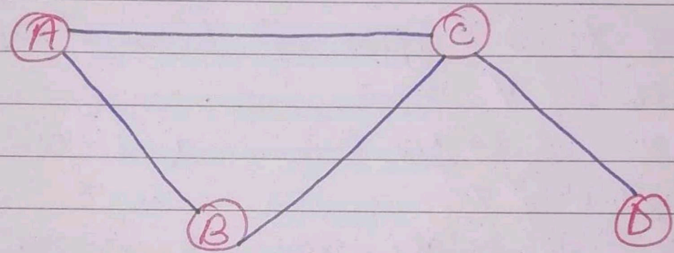
GRAPH & TREE

A graph is a set of points, called nodes or vertices which are interconnected by the set of lines called paths or edges.

$$G = (V, E)$$

V = non-empty set of vertices
E = set of edges.

eg:



Let us consider, a graph is $G = (V, E)$
vertices $V = \{a, b, c, d\}$
Edges $E = \{\{a, b\}, \{a, c\}, \{b, c\}, \{c, d\}\}$

Reachability or Reachable :-

If 'a' & 'b' are any two nodes of simple graph and there exist a path from node 'a' to 'b' so that 'b' is reachable from node 'a'.

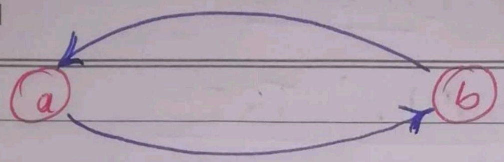
~~It~~ For eg:-

1) $a \longrightarrow b$

Node 'b' is reachable from node 'a'

2) $a \longleftarrow b$

Node 'a' is reachable from node 'b'

3) 

Node 'a' and 'b' both are reachable from each other.

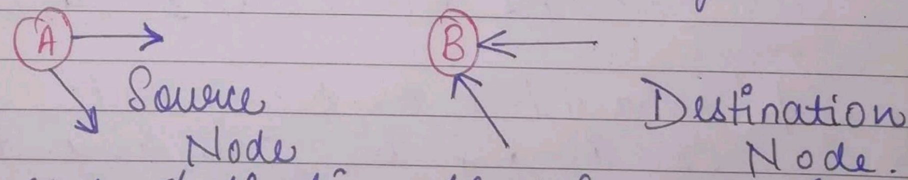
Degree of a Vertex / Node :-

The degree of a vertex V of a graph G denoted as degree (V) is the number of edges connected with the vertex V

There are two types of the degree.

1) In-degree of V
   Number of edges coming into V.
2) Out-degree of V
   Number of edges going from V

⇒ A node V is called source if it has only out-degree



⇒ A node V is called destination if it has only in-degree

Even & odd Vertex. →

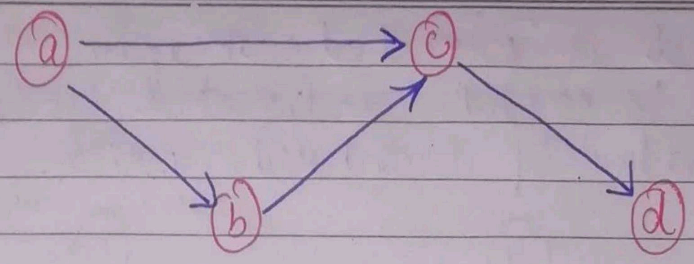
If the degree of a vertex is even, then the vertex is called an even vertex

If the degree of a vertex is odd, then the vertex is called an odd vertex.

Degree of a Graph →

The degree of a graph is the largest vertex degree of that graph.

For eg:-



Vertex	Indegree	Outdegree	Degree	Type
a	0	2	2	Even
b	1	1	2	Even
c	2	1	3	Odd
d	1	0	1	Odd

Degree of a Graph = 3 (highest degree)

~~for~~ Sum of degree of all the vertices $= 2+2+3+1$
$= 8$

Node 'a' is <u>source</u> node (because indegree is 0 only having outdegree 2

Node 'b' is <u>destination</u> node (because outdegree is 0 & having only indegree)

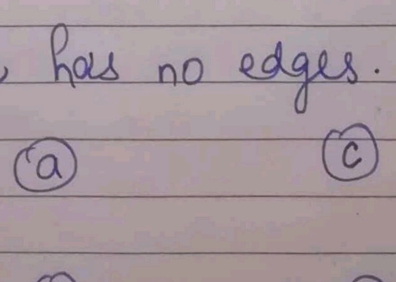
Types of Graph →

(1) Null graph

A null graph has no edges.

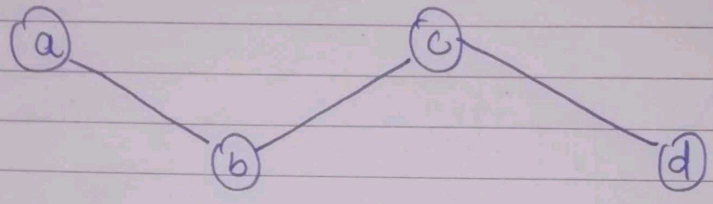
eg:-

A graph with four vertices (a, b, c, d) but no edge

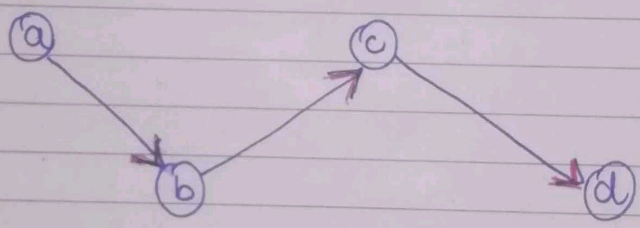


(2) Undirected & Directed Graph

A graph is called undirected graph if the edge set is made of unordered vertex.

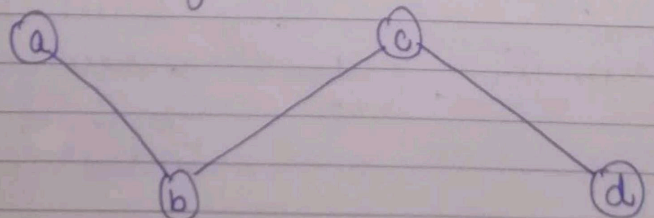


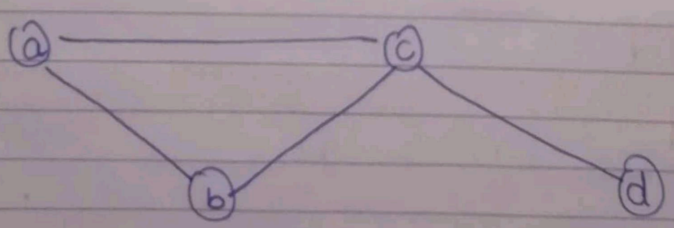
A graph is called directed graph if the edge set is made of ordered vertex pair.



(3) Simple Graph

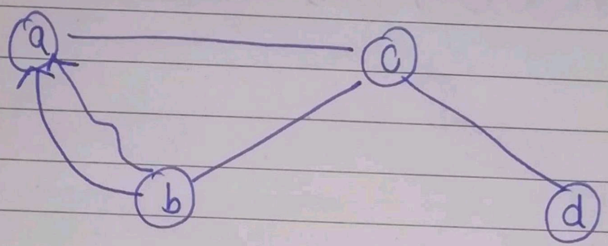
A graph is called simple graph if the graph is undirected and does not contain any loop or multiple edges.

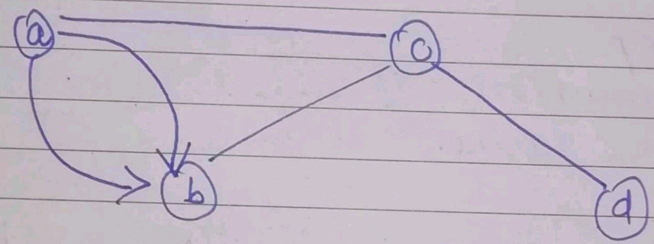




(4) MultiGraph

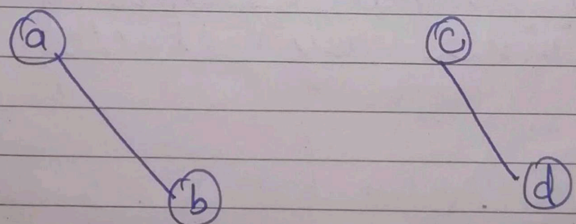
Graph having at least one loop or multiple edges





(5) Disconnected Graph

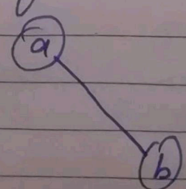
A graph is disconnected if at least two vertices of the graph are not connected by a path. If a graph G is disconnected, then maximal connected sub-graph of G is called a connected component of the graph G

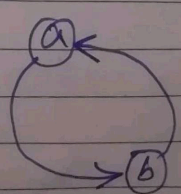


(6) Connected Graph

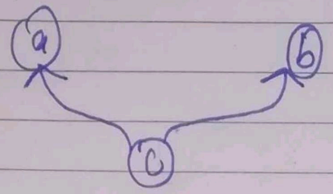
1) A graph is said to be strongly connected if for any pair of nodes of the graph both the nodes are reachable from each other

eg:-



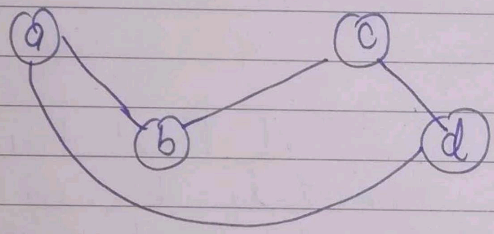


2) A directed graph is said to weekly connected if for any pair of nodes of the graph both the nodes are not reachable from each other.



(7) Regular Graph

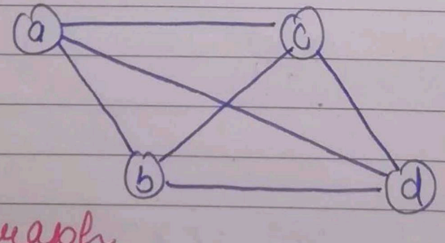
A graph is regular if all the vertices of the graph have the same degree.



Vertex	Degree
a	2
b	2
c	2
d	2

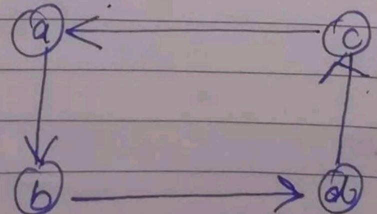
(8) Complete graph

A graph is called complete graph if every two vertices are joined by exactly one edge.



(9) Cycle graph

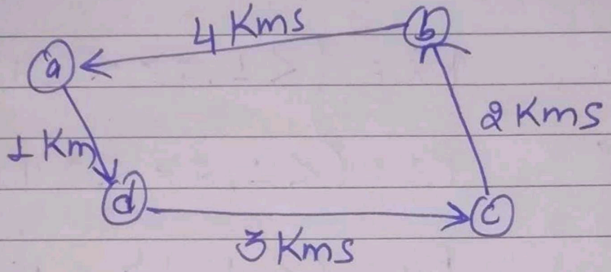
A graph consists of a single cycle it is called cycle graph.



if we start with 'a' then ended at a

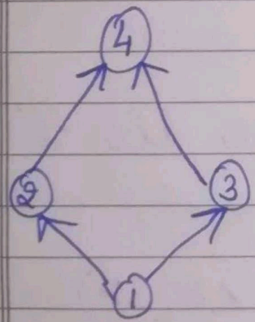
(10) **Weighted graph**

A graph is termed as weighted graph if all the edge in it are labeled with some values/ weight.



(11) **Isomorphic Graph**

when two graph having same characterstics. it is called isomorphic Graph.

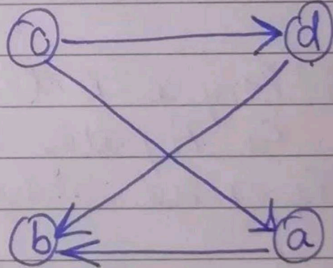


1 → c (2-out-degree)

4 → b (2 out-degree)

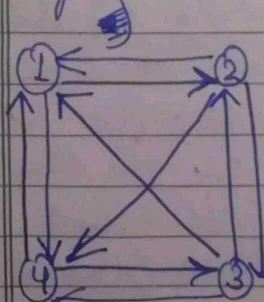
2 → a (1-indegree 1-outdegree)

3 → d (1 in degree 1-out degree)



Path → An open walk in which no vertex appear more than one is called path.

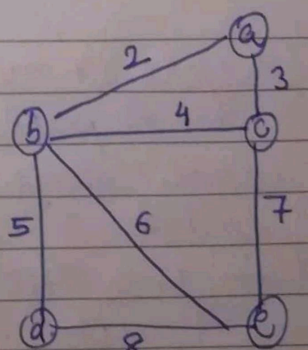
Eg:1) a 2 b 5 d 8 e



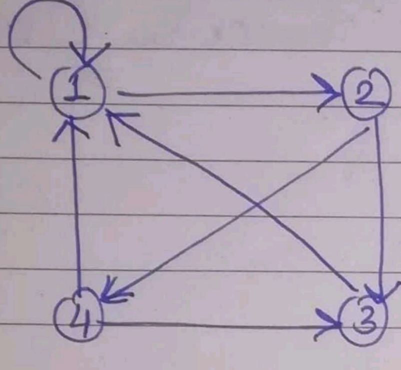
P1 = ~~<(1,~~
(<1,2><2,3>)

P2 = (<1,4> <4,3>)

P3 = (<1,2> <2,4> <4,3>)



Path → path originating in node 1 and ending in node 3 are:



P1 = (<1,2> <2,3>)

P2 = (<1,2> <2,4> <4,3>)

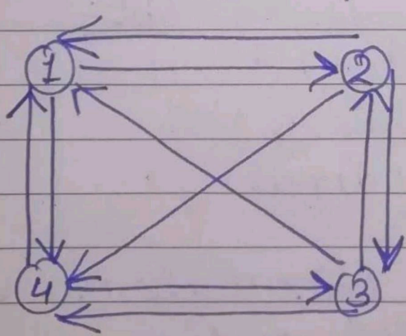
P3 = (<1,2> <2,4> <4,1> <1,2> <2,3>)

P4 = (<1,1> <1,2> <2,4> <4,3>)

Two types –
1. Simple path
2. Elementary path

1. Simple path
↳ A path in which the edges are all distinct is called a simple path.

eg:



P_1 = (<1,2> <2,3>)

P2 = (<1,4> <4,3>)

P3 = (<1,2> <2,4> <4,3>)

Cycle or circuit →
A path which originates and ends in the same node is called a cycle or circuit. A cycle is called simple if its path is simple i.e. no edge in the cycle appears more than once in the path.
A cycle is called ~~an~~ elementary if it does not traverse through any node more than once.

Simple & elementary cycle

$P1 = (\langle 1,2\rangle \langle 2,3\rangle \langle 3,1\rangle)$

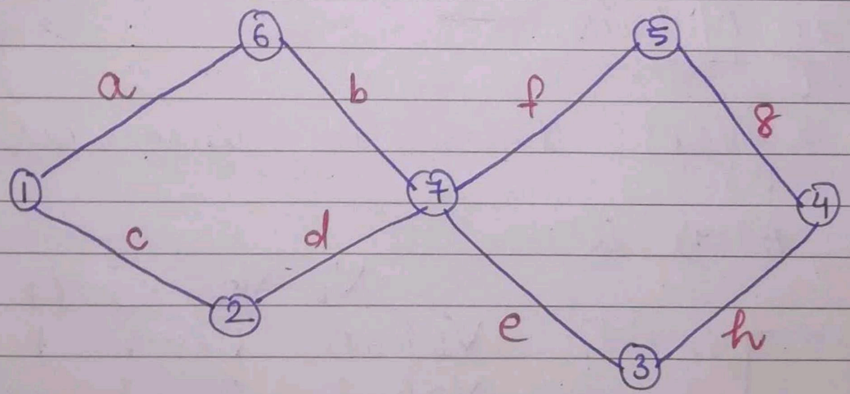
$P2 = (\langle 1,4\rangle \langle 4,3\rangle \langle 3,1\rangle)$.

Not Simple and not elementary Cycle

$P3 = (\langle 1,2\rangle \langle 2,4\rangle \langle 4,1\rangle \langle 1,2\rangle \langle 2,3\rangle \langle 3,1\rangle)$

Walk → Repeatation is allowed in walk.
→ Repeatation of edge
→ Repeatation of vertices.

closed   open



Eg: 1) 1 a 6 b 7 e 3 h 4 g 5 f 7 b 6 a 1

2) 1 a 6 b 7 d 2 c 1

3) 5 8 4 h 3 e 7 d 2 c 1

4) 6 b 7 d 2 c 1 a 6

Trail → Repeatation of edges is not allowed in trail.

closed / open

Repeatation of vertices is allowed.

Eg:- 7 f 5 g 4 h 3 e 7 b 6

closed trail is also called circuit.

Eg: 1 6 7 5 4 3 7 2 1 (closed trail)

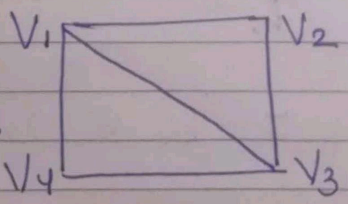
~~Isomorphism's Graph.~~

Matrix Representation of Graphs →

(a) **Adjacency Matrix.** →

If G is a graph, then its adjacency matrix. A_G or $A(G)$ is

(Undirected Graph)



$$\begin{array}{c|cccc} & V_1 & V_2 & V_3 & V_4 \\ \hline V_1 & 0 & 1 & 1 & 1 \\ V_2 & 1 & 0 & 1 & 1 \\ V_3 & 1 & 1 & 0 & 1 \\ V_4 & 1 & 0 & 1 & 0 \end{array}$$

Degree of Vertice V_1 = 3 (3 edges incident on vertex V_1)

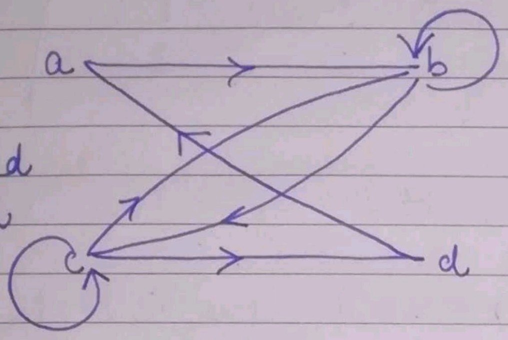
Degree of Vertices V_2 = 3

Degree of Vertices V_3 = 3

Degree of Vertices V_4 = 2

* Adjacent Matrix satisfying the symmetric condition so called symmetric matrix.

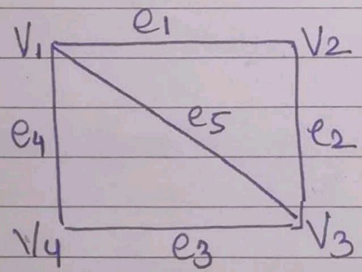
Directed Graph



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

Degree of vertex a = 1
Degree of vertex b = 2
Degree of vertex c = 3
Degree of vertex d = 1

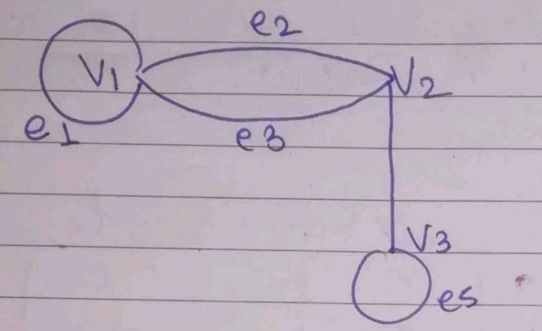
(b) Incidence Matrix representation. →



$G = (V, E)$ undirected Graph

I_G or $I(G)$

$$\begin{array}{c|ccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ \hline V_1 & 1 & 0 & 0 & 1 & 1 \\ V_2 & 1 & 1 & 0 & 0 & 0 \\ V_3 & 0 & 1 & 1 & 0 & 1 \\ V_4 & 0 & 0 & 1 & 1 & 0 \end{array}$$

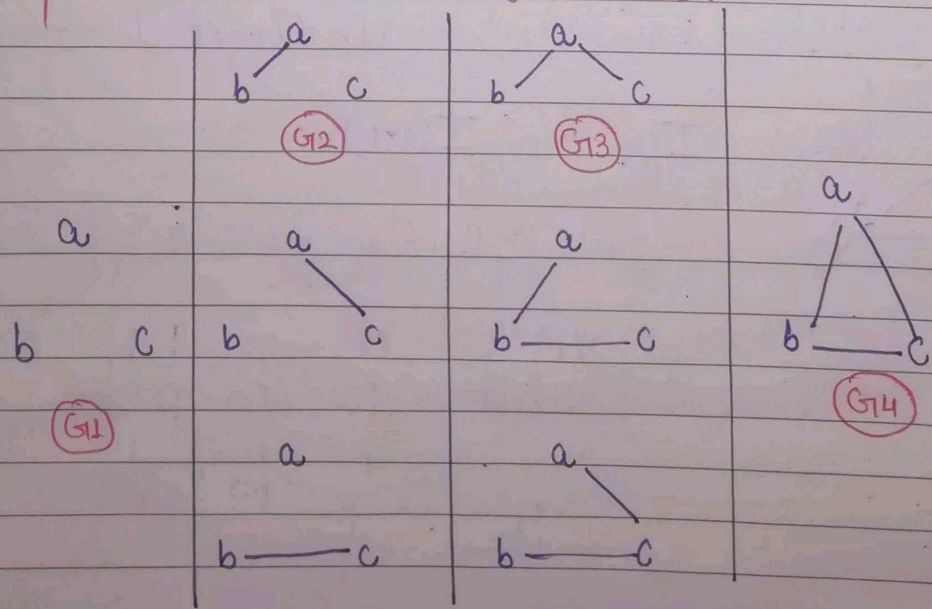


$$\begin{array}{c|ccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ \hline V_1 & 1 & 1 & 1 & 0 & 0 \\ V_2 & 0 & 1 & 1 & 1 & 0 \\ V_3 & 0 & 0 & 0 & 1 & 1 \end{array}$$

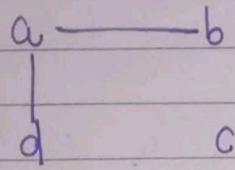
1) Every Row of incidence matric contains exactly two 1's.
2) A row with all 0's represent an isolated vertex
3) A row with single one represent an pendant vertex.
4) The total no. of 1s in i^{th} row represent the degree of the vertex.

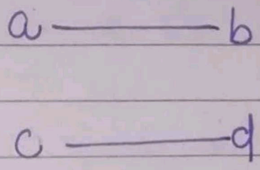
ISOMORPHISM'S OF GRAPHS →

Q1 How many simple non-isomorphic graph are possible with 3 vertices.

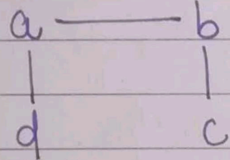


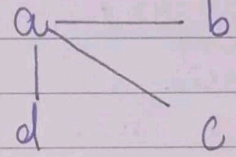
Q2. How many simple non-isomorphic graph are possible with 4 vertexs and 2 edges?

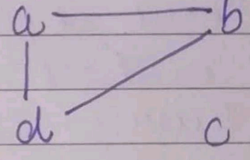




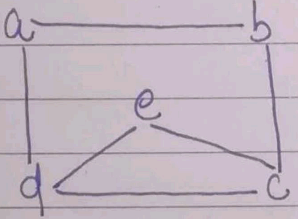
Q3. How many simple non-isomorphic graph are possible with 4 vertices and 3 edges?

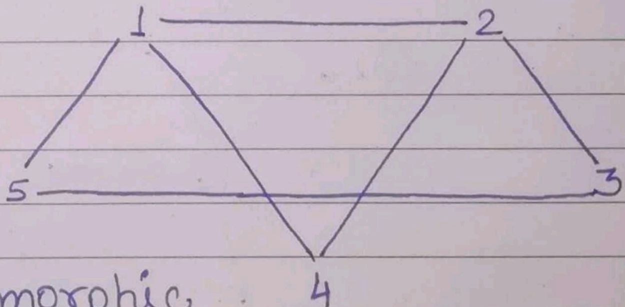




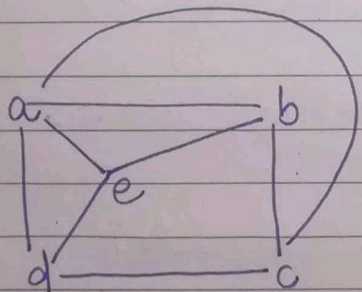


Q4 Identify Isomorphic Graph →

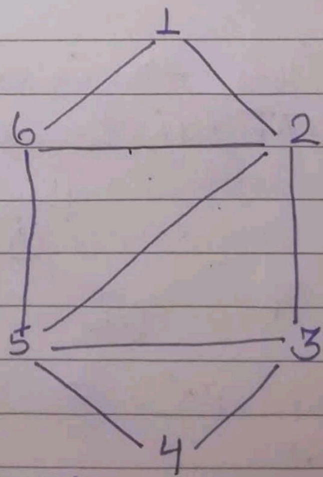




Both graph are isomorphic



No. of vertices = 5



No. of vertices = 6

Both graph are not isomorphic

TREES — Trees is a connected graph without any ~~connected~~ circuit

OR

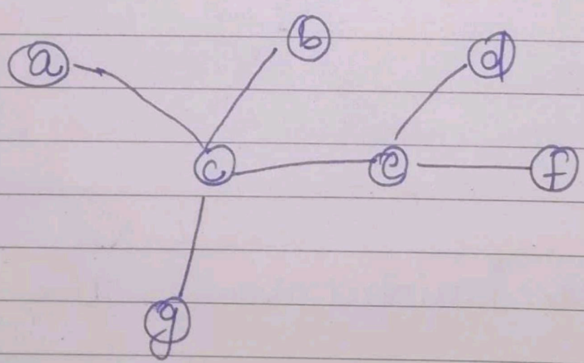
Tree is a simple graph.

OR

Tree is a connected acyclic graph
have no cycle

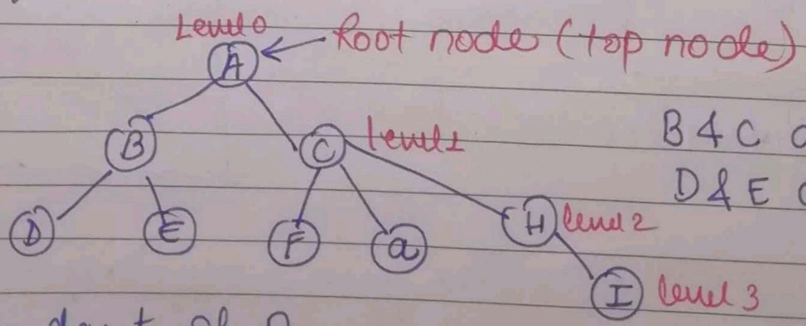
Properties →

1) There is only one path b/w every pair of vertices in a tree.
2) every edge in a tree is a bridge.
3) A tree with n vertices then it have (n-1) edge
4) Any connected graph with (n) vertices & (n-1) edges is also a tree.



Rooted Tree → is a tree in which some node is distinguish as a root.

A = root
B = parent of D & E
(node highes than parent) ← A = Ancestor of D & E
D & E = Descendant of A



B & C are sibling
D & E are sibling

leaf Node → do not have any child (D, E, F, G, I)
Interior Node → B, C, H
Height → length of longest path b/w root to any leaf node
Height = 3 (given graph)

Spanning Tree →

1) spanning subgraph that is a tree.
2) It composed of all vertices and some edges of G, $\underbrace{\text{some}}_{(V-1)}$

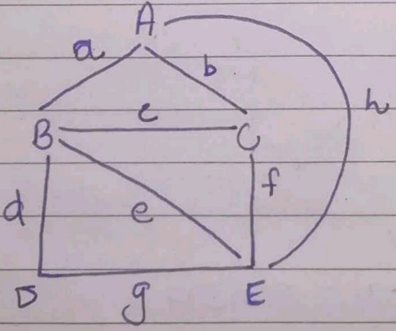
Branch — An edge in a spanning tree

chord — An edge of G that is not given in spanning tree.

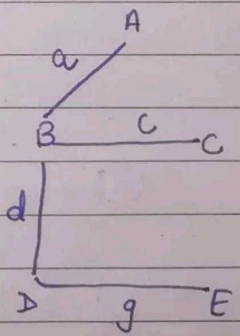
3) All the vertices are reachable

Simple Graph →

(1)

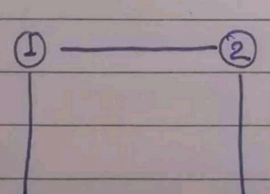


Spanning tree →

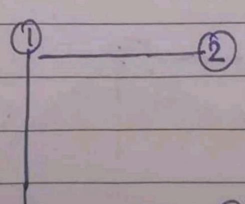


Branch = {a, c, d, g}
chord = {h, f, ~~g~~, b, e}

(2)



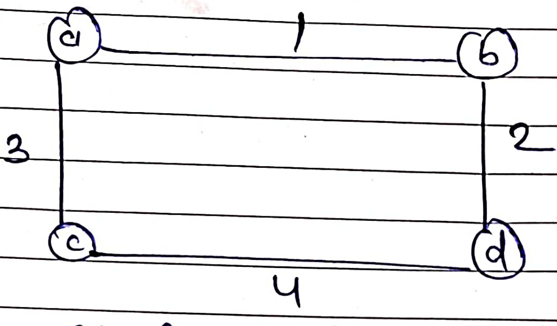
G



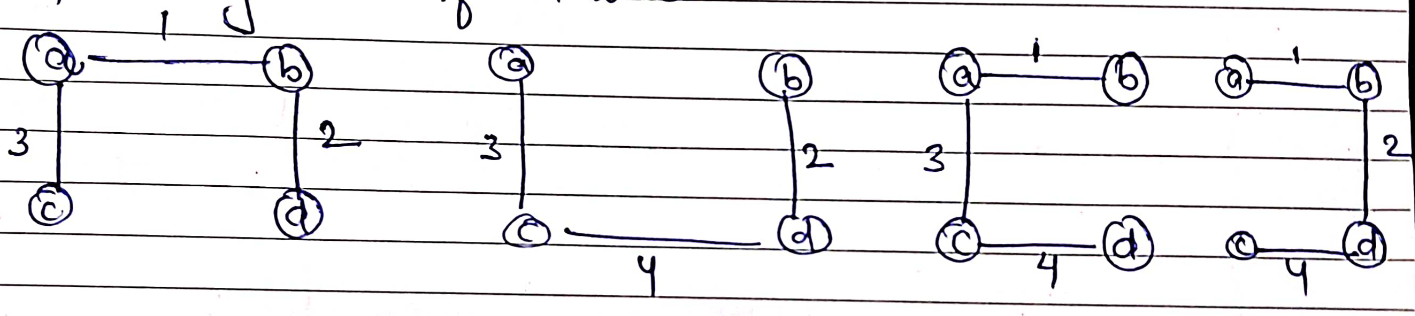
ST

Minimum Spanning Tree

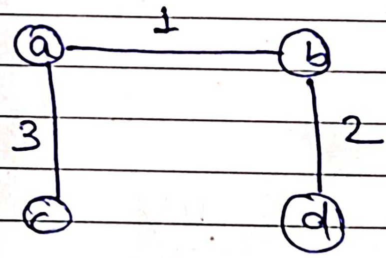
Let G be a weighted graph. A minimal spanning tree of G is a spanning tree of G with minimum weight



Spanning tree of G are



Minimal Spanning tree is



$1+3+2=6$

Minimum Spanning Tree

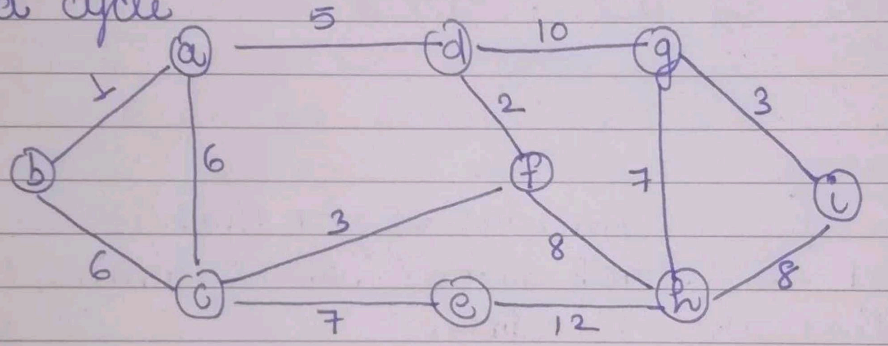
Prim's Algorithm | Kruskal Algo.

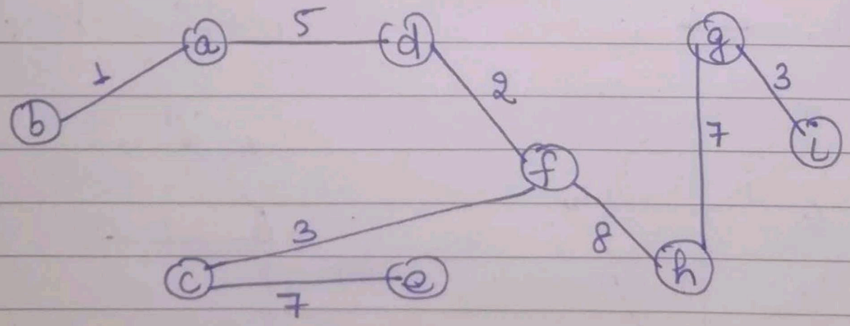
(1) PRIM'S ALGORITHM

To find minimum cost spanning tree.

→ Initial start with any of the vertices.

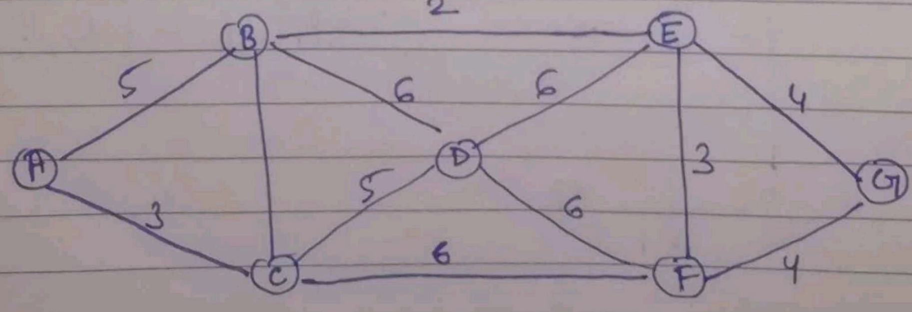
→ Avoid cycle



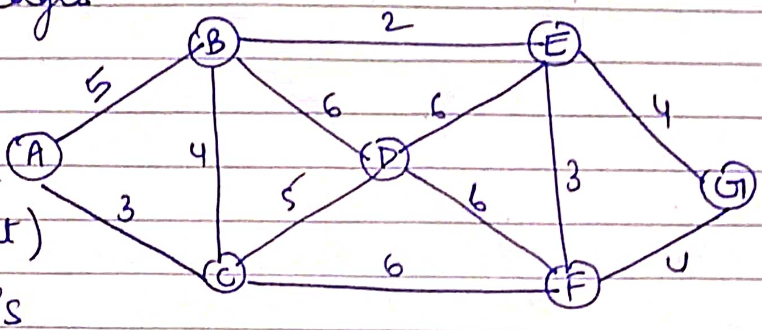


$= 1+5+2+3+7+8+7+3 = 36$

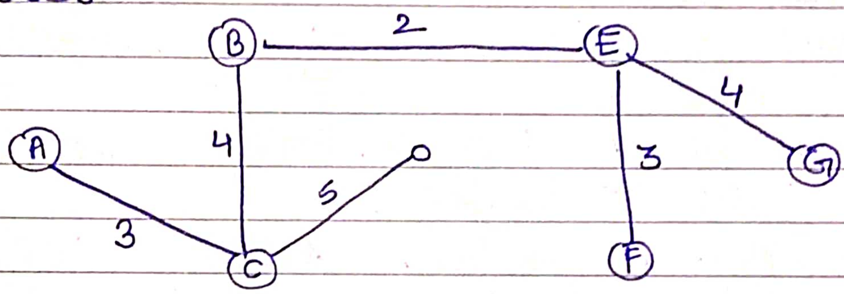
(2) Kruskal Algorithm



1) Take all vertices
2) (n-1) edges
3) Take shortest path (shortest weight)



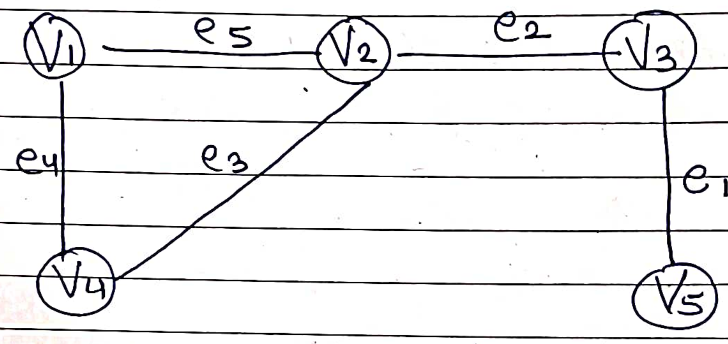
4) cycle is not allowed



$$3+4+5+2+3+4 = 21 \text{ (MC)}$$

Euler Path

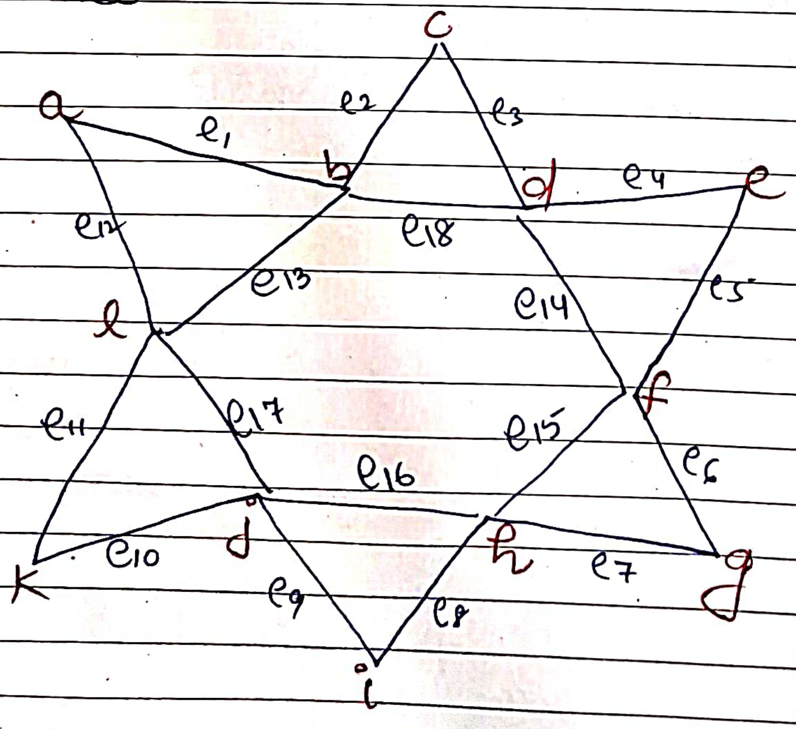
A path in a graph is said to be an euler path if it traverses each edges in the graph once and only once.



Euler path V5 e1 V3 e2 V2 e3 V4 e4 V1 e5

Euler Circuit

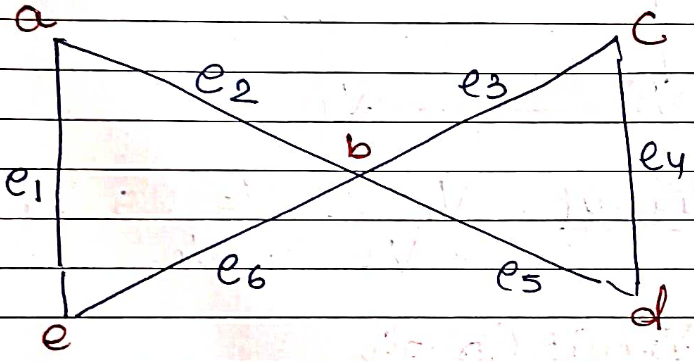
A circuit in a graph is said to be euler circuit if it traverses each edges in the graph once & only once



Euler Circuit $l\ e_{11}\ K\ e_{10}\ j\ e_9\ i\ e_8\ h\ e_7\ g\ e_6\ f$
$e_5\ e\ e_4\ d\ e_3\ c\ e_2\ b\ e_1\ a\ e_{12}$
$l\ e_{17}\ e_{16}\ e_{15}\ e_{14}\ e_{18}\ e_{13}$

Euler Graph

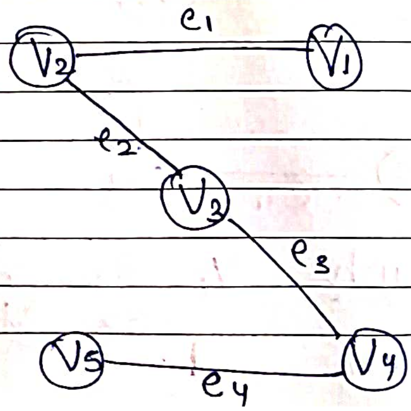
A connected graph which contain an Euler circuit is called euler ~~p~~ graph.



Euler Graph $b\ e_3\ c\ e_4\ d\ e_5\ b\ e_2\ a\ e_1\ e\ e_6$

Hamiltonian path

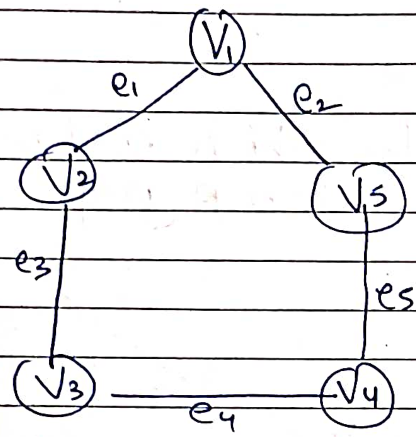
A path which contain every vertex of a graph G exactly once is called HP.



HP → $V_1\ e_1\ V_2\ e_2\ V_3$
$e_3\ V_4\ e_4\ V_5$

Hamiltonian Circuit

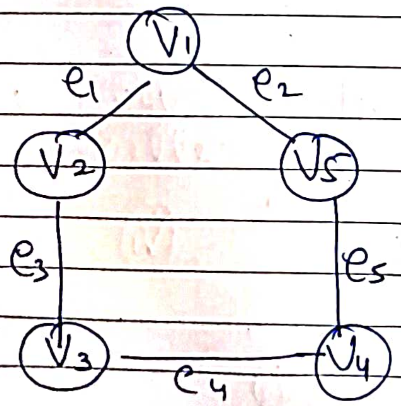
A circuit that passes through each of the vertices in a group G exactly one except the starting vertex & end vertex is called HC



Hamiltonian Circuit $V_1\ e_2\ V_5\ e_5\ V_4\ e_4\ V_3\ e_3\ V_2\ e_1\ V_1$

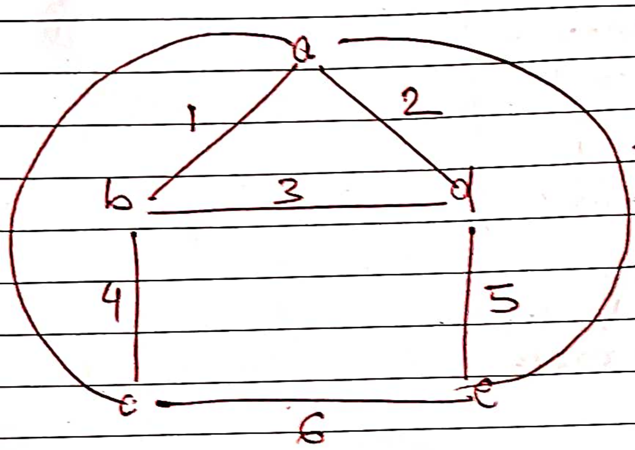
Hamiltonian ~~Circuit~~ Graph

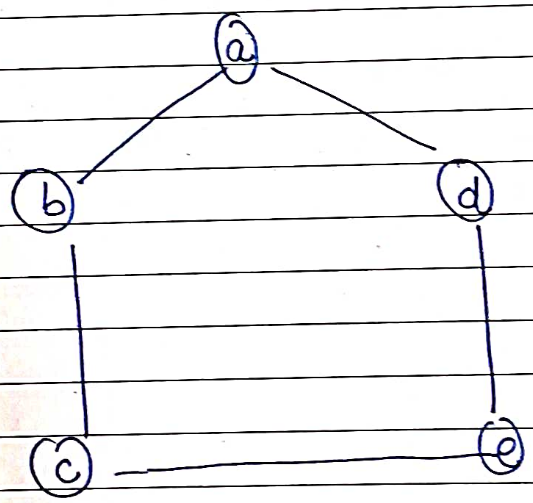
A connected graph which contain Hamiltonian circuit is called Hamiltonian Graph



Hamiltonian Graph $V_1\ e_2\ V_5\ e_5\ V_4\ e_4\ V_3\ e_3\ V_2\ e_1\ V_1$

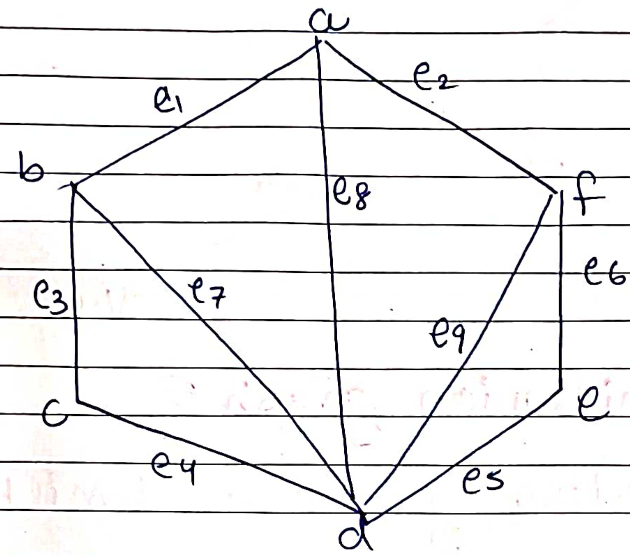
Q1. Determine a minimum Hamiltonian circuit for the graph given below



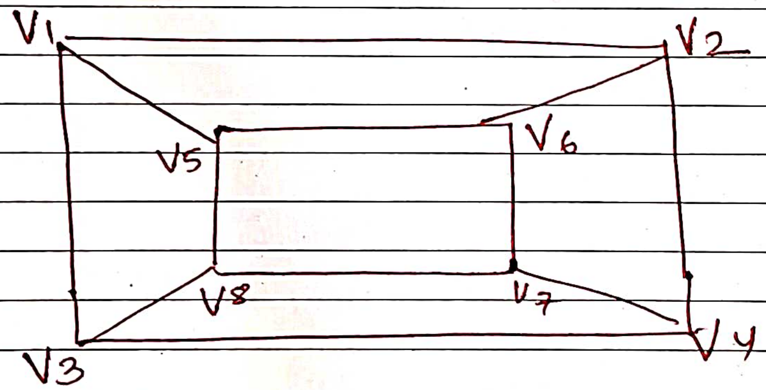


Hamiltonian Circuit should be closed and cover all the vertex of the graph.

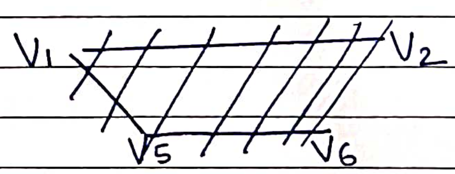
Q2. Draw a graph with 6 vertices containing a Hamiltonian circuit but not Euler circuit.



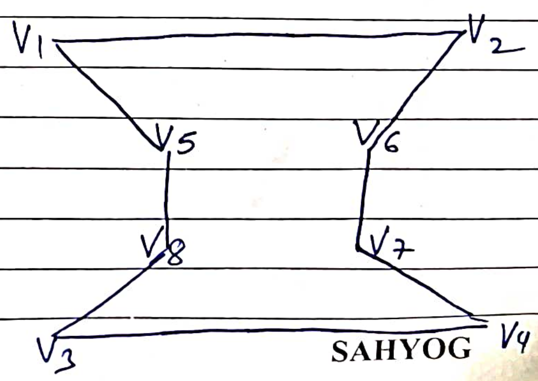
Q3 Discuss the graph shown in below.

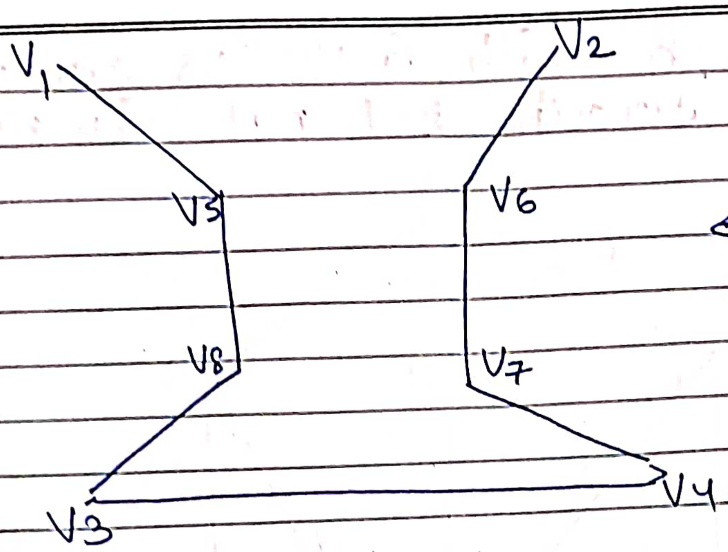


(i) Draw Hamiltonion circuit and Hamiltonion path.



This is Hamiltonian circuit →





← This is Hamiltonian path

(ii) Is it Hamiltonion graph?

Yes the given graph is Hamiltonion graph.

(iii) Is it Euler graph?

No. the given graph is not Euler graph.