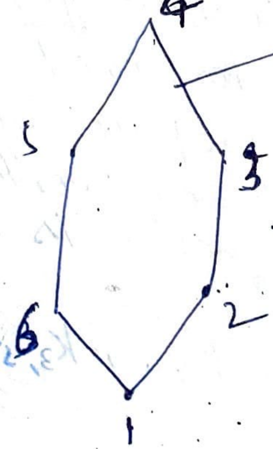
Bipartite:– A simple graph G is called bipartite if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2).

* When this condition holds, we call the pair (V_1, V_2) a bipartition of the vertex set V of G.

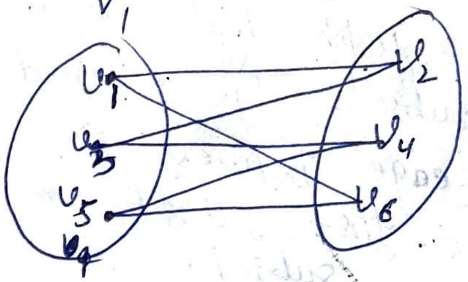
Ex:– (1)

C_6



→ C_6

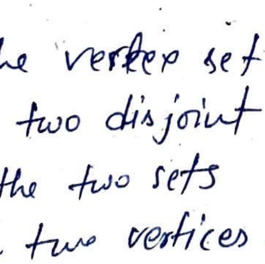
break into 2 vertices



$V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$

where
$V_1 = \{v_1, v_3, v_5\}$
$V_2 = \{v_2, v_4, v_6\}$.

(2) K_3



→ we can't find a partition of ~~disjoint~~ vertex V which has two disjoint vertices V_1 & V_2

because

1 connected to 2

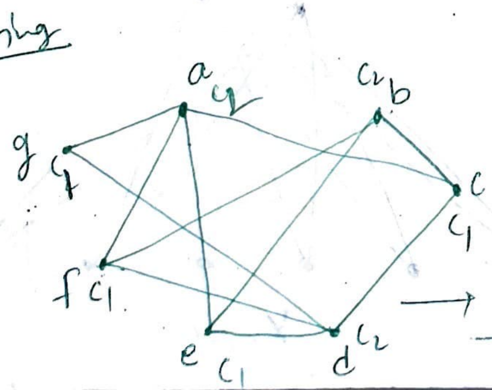
1 is also connected to 3

2 ——— 1

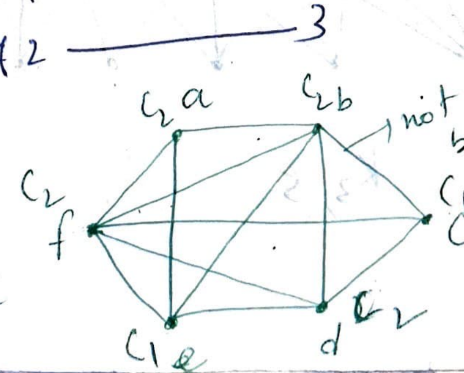
~~4~~ 2 ——— 3

if we divide the vertex set of K_3 into two disjoint sets, one of the two sets must contain two vertices.

Coloring



→ bipartite



→ not a bipartite

check

Thm: A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.

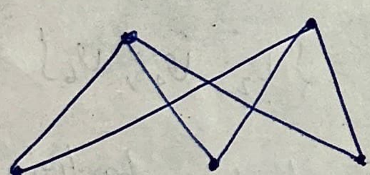
c_1 or c_2 $\triangle$ → not bipartite

c_1 c_2

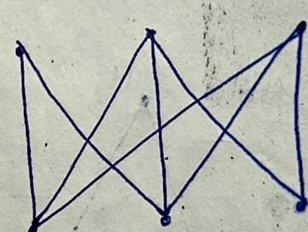
(This theorem provides a useful criterion for determining whether a graph is bipartite)

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

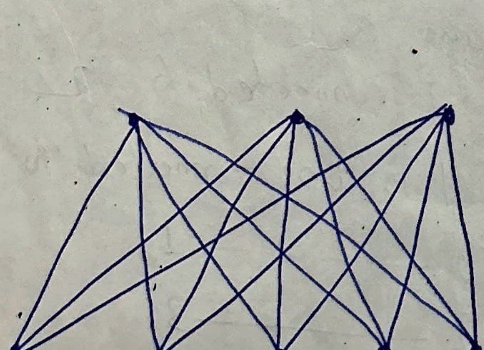
The complete bipartite graphs $K_{2,3}$, $K_{3,3}$, $K_{3,5}$ and $K_{2,6}$ are displayed in fig 9 —



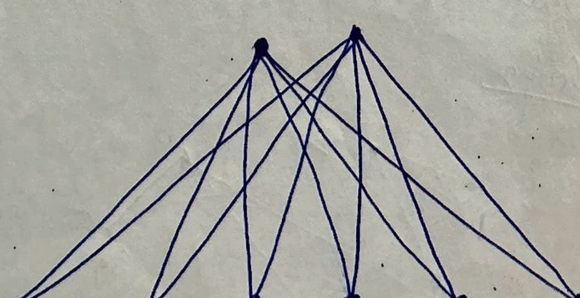
$K_{2,3}$



$K_{3,3}$



$K_{3,5}$



$K_{2,6}$

n-cubes → $n \to$ length

then no. of vertices of n-cubes $= 2^n$

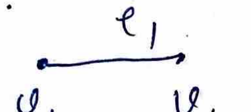
and ~~edges~~ $= n \cdot 2^{n-1}$

Note: n-cubes is always n-regular.

Ex: Let $n = 1$ then find 1-cube

no. of vertices $= 2^1 = 2$

no. of edges $= n \cdot 2^{n-1} = 1 \cdot 2^{1-1} = 1$

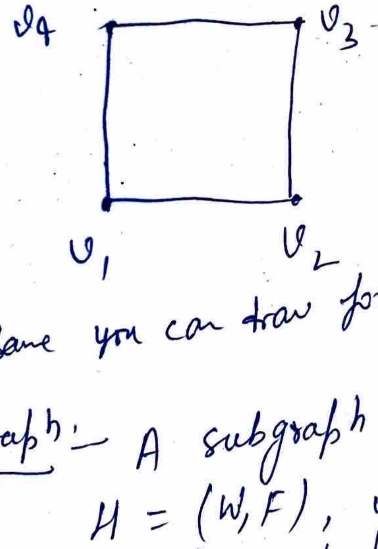


No need

let $n = 2$ then find 2-cube

no. of vertices $= 2^2 = 4$

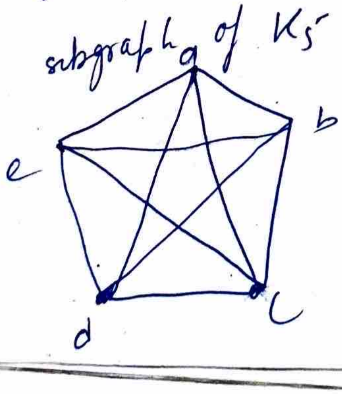
no. of edges $= n \cdot 2^{n-1} = 2 \cdot 2^{2-1} = 2 \cdot 2 = 4$

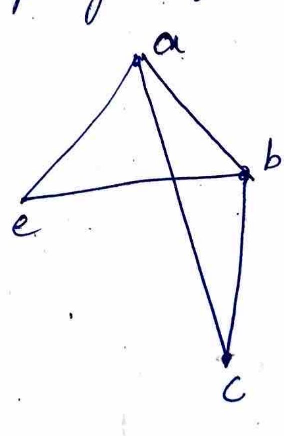


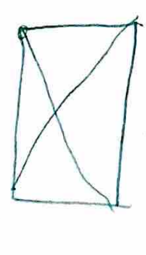
Same you can draw for $n = 3$, $n = 4$.

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

Ex: subgraph of K_5



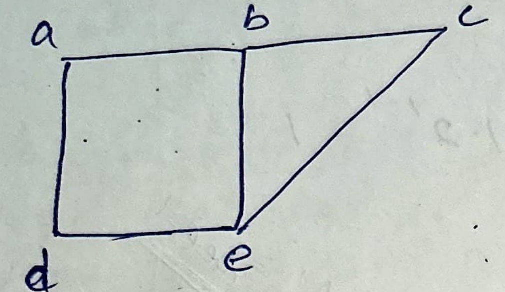


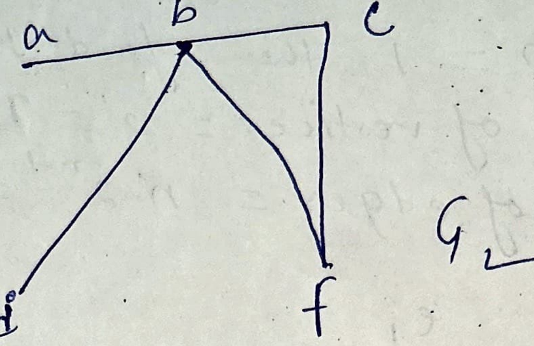


Union of simple graph

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

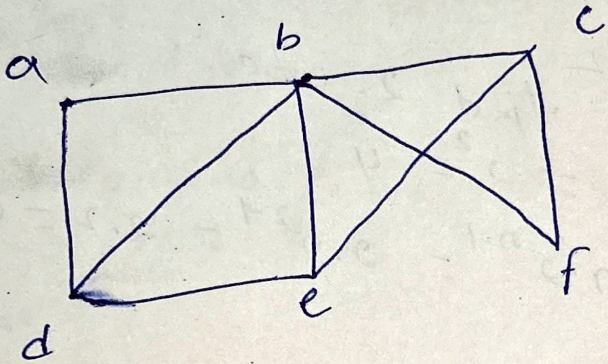
Ex:





G_2

G_1



$G_1 \cup G_2$

V_r

{a.b c d e f}

Adjacency Matrices:-

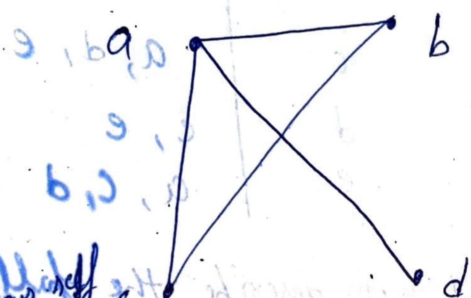
Let $G = (V, E)$ is a simple graph with $|V| = n$ and the vertices of G are listed arbitrarily as $v_1, v_2, \ldots, v_n$.

[The adjacency matrix A (or A_G) of G with respect to this listing of the vertices, is the $n \times n$ zero-one matrix]

A matrix $A = [a_{ij}]$ is called adjacency matrix if

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

Ex:- 2 Use an adjacency matrix to represent the following graph



vertices are $\begin{matrix} v_1 & v_2 & v_3 & v_4 \\ a & b & c & d \end{matrix}$

$|V| = 4$

$\Rightarrow [A]_{4\times 4}$

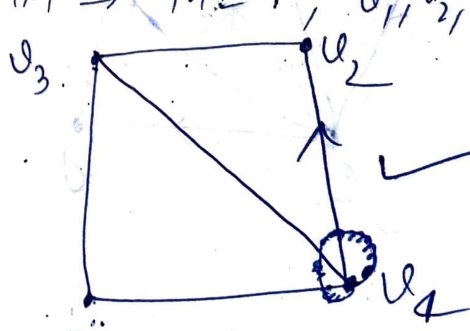
means self loop

bcz UD

$$A_G = \begin{array}{c c} & \begin{matrix} a & b & c & d \end{matrix} \\ \begin{matrix} a \\ b \\ c \\ d \end{matrix} & \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

(2) Draw a graph with the adjacency matrix

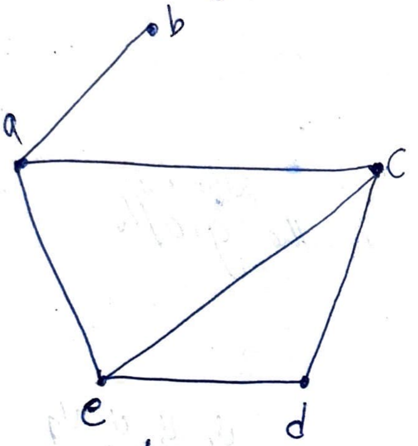
$$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \longrightarrow 4\times 4 \rightarrow |V| = 4, \; v_1, v_2, v_3, v_4$$



Representing Graphs:

To represent a graph with no multiple edges is to use adjacency lists, which specify the vertices that are adjacent to each vertex of the graph.

Ex:- ① Use adjacency lists to describe the following simple graph:

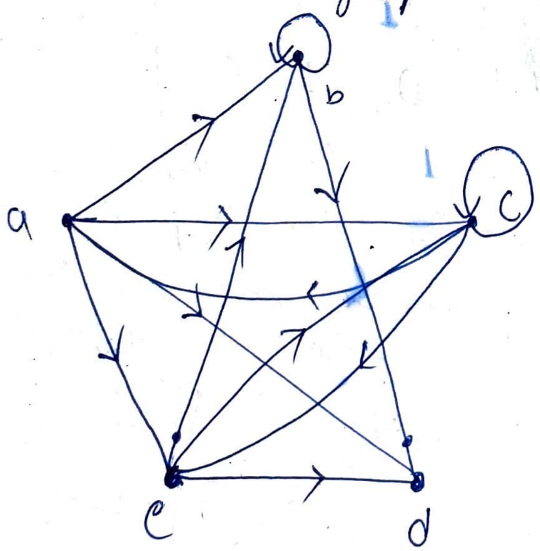


simple graph

An adjacency list for a simple graph

Vertex	Adjacent vertices
a	b, c, e (bcz. it is UD)
b	a
c	a, d, e
d	c, e
e	a, c, d

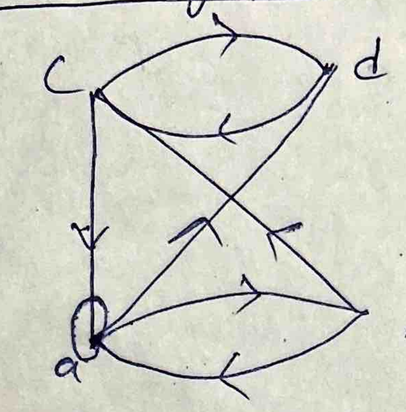
② Use adjacency lists to describe the following directed graph.



An adjacency list for a directed graph

Initial Vertex	Terminal vertices
a	b, c, d, e
b	b, d
c	c, e, a
d	
e	d, c, b

Directed multigraph



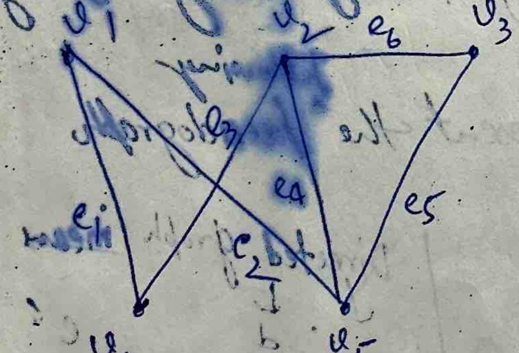
$$\begin{array}{c|cccc} & a & b & c & d \\ \hline a & 1 & 1 & 0 & 1 \\ b & 1 & 0 & 1 & 0 \\ c & 1 & 0 & 0 & 1 \\ d & 0 & 0 & 1 & 0 \end{array}$$

Incidence Matrices

This is another method to represent graphs. Let $G = (V, E)$ be an undirected graph. Suppose that $v_1, v_2, \ldots v_n$ are the vertices and $e_1, e_2, \ldots e_m$ are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix $M = [m_{ij}]$, where

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

Ex:- Represent the following graph with an incidence matrix



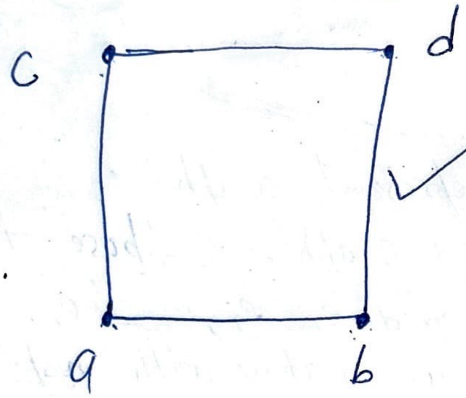
An undirected graph.

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

(3) Draw a graph with the adjacency matrix

$$\begin{array}{c|cccc} & u_1 & u_2 & u_3 & u_4 \\ \hline u_1 & & & & \\ u_2 & & & & \\ u_3 & & & & \\ u_4 & & & & \end{array}\quad \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

a b c d

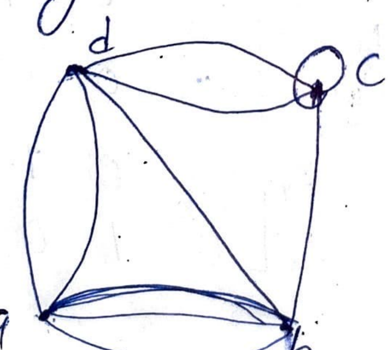


Note:- (1) The adjacency matrix of a simple graph is symmetric, i.e $a_{ij} = a_{ji}$ .. (bcz it is UD ~~prin~~ i.e if aRb then bRa)

(2) Each entry a_{ii}, $i = 1, 2, \ldots n$ is 0, because a simple gph has no loops.

How to represent a non-simple graph by using adjacency matrix:-

Ex 2:- Use an adjacency matrix to represent the following pseudograph

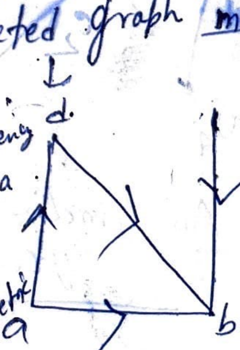


total no. of edges

$$A_G = \begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}_{4\times 4}$$

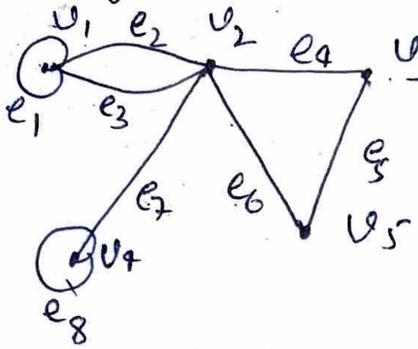
Directed graph means

The adjacency matrix of a DG does not have to be symmetric



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

1. Represent the pseudograph shown in the following fig. Using an incidence matrix.



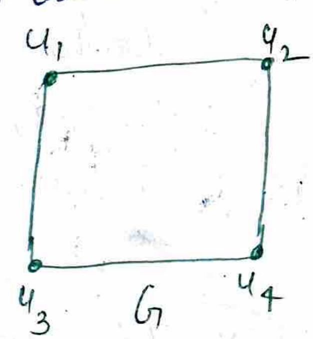
$$\begin{array}{c|cccccccc} & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \\ \hline v_1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ v_3 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ v_4 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ v_5 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{array}$$

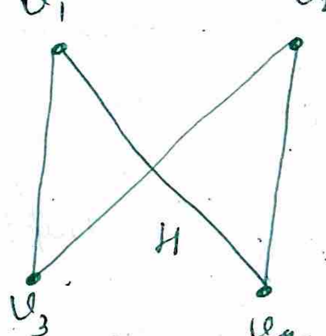
$f: A \to B$

[onto if for every element of B $\exists$ $a \in A$ such that $f(b) = a$].

Isomorphism of Graphs :- The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there is a one-to one and onto f^n f from V_1 to V_2 with the property that a and b are adjacent in G_1 iff $f(a)$ and $f(b)$ are adjacent in G_2, for all a and b in V_1. such a f^n f is called an isomorphism.

Q2 Show the following graphs $G = (V, E)$ and $H = (W, F)$ are isomorphic.





Soln: $\Rightarrow$ from this.

The f^n f with $f(u_1) = v_1$, $f(u_2) = v_4$, $f(u_3) = v_3$, and $f(u_4) = v_2$ is a one-to-one correspondence bet^n V & W.

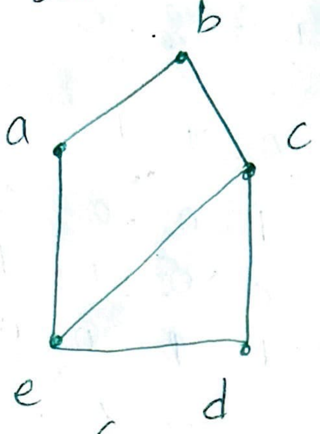
The adjacent vertices in G are u_1 and u_2, u_1 and u_3, u_2 and u_4 and u_3 and u_4, and each of the pairs $f(u_1) = v_1$ and $f(u_2) = v_4$, $f(u_1) = v_1$ and $f(u_3) = v_3$, $f(u_2) = v_4$ and $f(u_4) = v_2$, and $f(u_3) = v_3$ and $f(u_4) = v_2$ are adjacent in H.

one-one

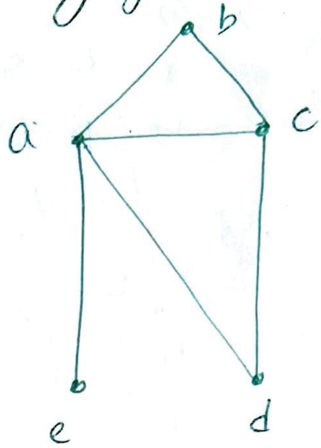
whenever $f(x) = f(y)$ then $x = y$ $\forall$ $x, y \in A$

$f: A \to B$)

(2) Show that the following graphs ~~are~~ is not isomorphic



G



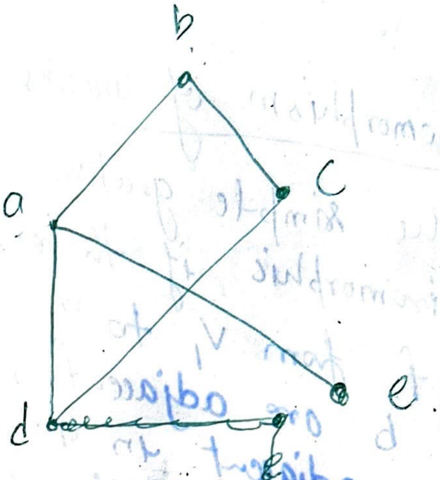
(1) $f(a) = a$ and $f(a) = d$

if $f(a) = f(b)$ whenever $a = b \; \forall a, b \in G$

if $a \neq b$ then $f(a) \neq f(b)$

Here $a \neq d$ but $f(a) = f(a)$

$\Rightarrow$ not one-one.



Both G and H have five vertices and six edges.
However, H has vertex of degree one, namely, e, whereas G has no vertices of degree one. It follows that G and H are not isomorphic.

* The number of vertices, the number of edges, and the number of vertices of each degree are all invariants under isomorphism.
If any of these quantities differ in two simple graphs, these graphs cannot be isomorphic.

(2) Try to solve book's excercise.

Table for Dijkstra's Algorithm

Note:- if two vertex are not directly connected by an edge then represent it by ∞.

Source	b	c	d	e	Z
$\{a\}$	$2\{a\}$	$3\{a\}$	∞	∞	∞
$\{a, b\}$	—	$3\{a\}$	$7\{a, b\}$	$4\{a, b\}$	∞
$\{a, b, c\}$	—	—	$7\{a, b\}$	$4\{a, b\}$	∞
$\{a, b, c, e\}$	—	—	$5\{a, b, e\}$	—	$8\{a, b, e\}$
$\{a, b, c, e, d\}$	—	—	—	—	$7\{a, b, e, d\}$

Shortest path from a to Z is $a - b - e - d$

~~and length~~ $= 2 + 2 + 1 + 2 = 7$

Theorem:- Dijkstra's algorithm finds the length of a shortest path between two vertices in a connected simple undirected weighted graph.

Path for DG

Defn:- Let n be a nonnegative integer and G a directed graph. A path of length n from u to v in G is a sequence of edges $e_1, e_2, \ldots, e_n$ of G such that e_1 is associated with (x_0, x_1), e_2 is associated with (x_1, x_2) and so on, with e_n associated with (x_{n-1}, x_n), where $x_0 = u$ and $x_n = v$. When there are no multiple edges in the DG, this path is denoted by its vertex sequence $x_0, x_1, x_2, \ldots x_n$.

no need

* A path of length greater than zero that begins and ends at the same vertex is called a circuit or cycle.
* A path or circuit is called simple if it does not contain the same edge more than once.

First explain what is a walk

Walk → An alternating sequence of verties and edges of a graph.

where ① no edge appears more than once
② vertex may appears more than once

and a walk

path (open walk) → starting & ending point are different

closed path (closed walk) → starting & ending points are same a – b – a – b – d — 1 → four edge becz length will be four.

a – b – a – c – d — 2

a – b – d – b – d — 3

a – b – d – c – d — 4

a – c – a – b – d — 5

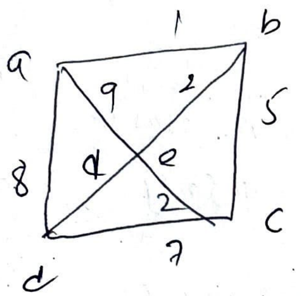
a – c – a – c – d — 6

a – c – d – b – d — 7

a – c – d – c – d — 8

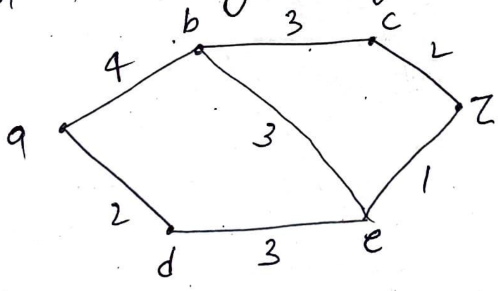
Weighted Graphs: – Graphs that have a number assigned to each edge are called weighted graphs.

Ex:-



Shortest path Problem :

Ex:- 2 What is the length of a shortest path betⁿ a and Z in the following weighted graph.



We want to go a to Z

first either we will start from initial vertex a then we have two way to go. first is b and second is Z. but the shortest path is a to d becz distance is 2 i.e d is the closed vertex to a. a to b distance is 4

Now go d to e so length $(a, d, e) = 5$. ~~while to~~

~~So~~ No we will check a to b length is 4 so either we can go b to e and then e to z then length $(a, b, e, z) = 8$

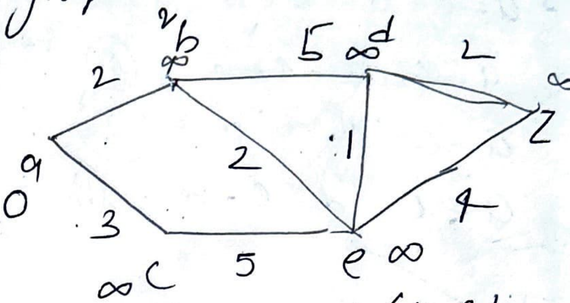
another way $(a, d, e, z) = 6$ & another way $(a, b, c, z) = 9$

⇒ shortest path is (a, b, c, z) with length 6:-

<u>Find Shortest path by using Dijkstra's algorithm :-</u>

Qus 2 Find shortest path from a to z using Dijkstra's algorithm of the following graph

and if.
$d(u) + c(u,v) < d(v)$
then
$d(v) = d(u) + c(u,v)$



initial Vertex is a so $d(a) = 0$ and cost of (u, v) i.e $c(u, u) = \infty$ if v is not connected directly to u.

<u>Now</u>

Source	b	c	d	e	z
{a}	2{a}	3{a}	∞	∞	∞

(bcz other are not directly connected by a).

Now min^m distance length is 2 so we will stop here [and find next shortest distance length which is 3{a}]

{a, b}	—	3{a}	7{a, b}	4{a, b}	∞
{a, b, c}	—	—	7{a, b}	4{a, b}	∞

check $d(a, b, c) + c(a, b, c, d) < d(d)$

$7 < \infty$

⇒ 7 as it

{a, b, c, e}	—	—	5{a, b, e}	—	8(a, b)

bcz 5 < 7

{a, b, c, e, d}	—	—	—	—	7(a, b, ed)

7 < 8

4 to d
a-b-d
2+5 = 7
7 < ∞

4 < ∞   Shortest path ⇒ a-b-e-d-z

2+2+1+2 = 7

u = a,
v = b

if $d(a) + c(a,b) < d(b)$
then $d(b) = d(u) + c(u,v)$ → $0 + 2 < 2$

$d(v) =$ $\boxed{d(a) + c(u,v) < d(v)}$

open walk mean → my starting and ending points are different

closed walk means → my starting and ending points are same

open walk is called path

⇒ no ~~verte~~ edge appears more than once (bcz its a walk)

⇒ no vertex appers more than once.

⇒ No. of edges in a path is called length of path

$v_1 - e_1 - v_2 - e_2 - v_3 - e_3 - v_4$

no. of edge = 3 = legth of path.

Closed path is called circuit → $v_1 - e_1 - v_2 - e_2 - v_3 - e_3 - v_4 - e_4 - v_1$
there → no vertex appears more than once except initial & final vertex.

And path or circuit are simple becuas they do not contain same edge more than once

(Q) 2nd path is possible

$v_2 - e_5 - v_5 - e_6 - v_3 - e_2 - v_2$ → ~~cl~~s circuit

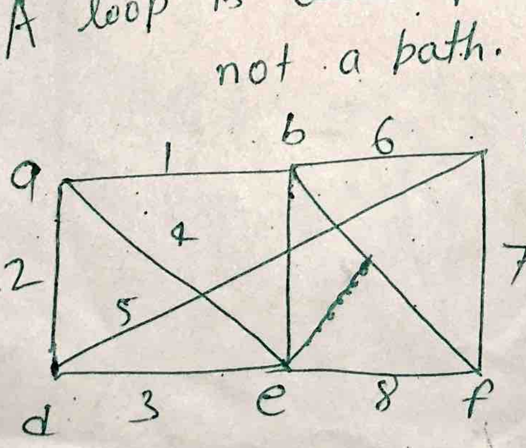
$v_2 - e_5 - v_5 - e_6 - v_3$ → path

$v_2 - e_2 - v_3 - e_3 - v_4$ → path

$v_2 - e_2 - v_3 - e_3 - v_4 - e_7 - v_2$ → closed circuit.

A loop is ~~closed path~~ $v_1 - e_1 - v_1$ not a path.

(2)

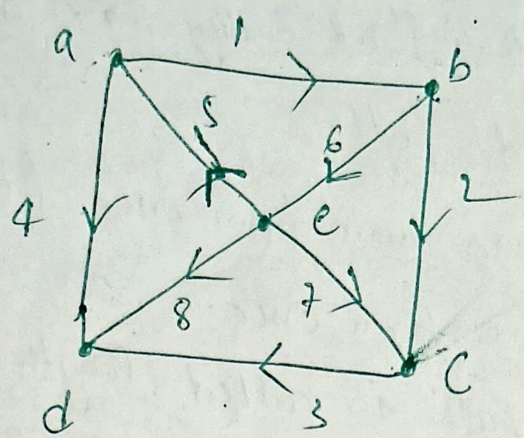


b.c f e b
↓
circuit

→ a-2-d-5-c-7-f-8-e
→ is a simple path
no. of edge = 4 = length

→ ~~bcze~~ d e c a → is not a path bcz {e, c} is not an edge.

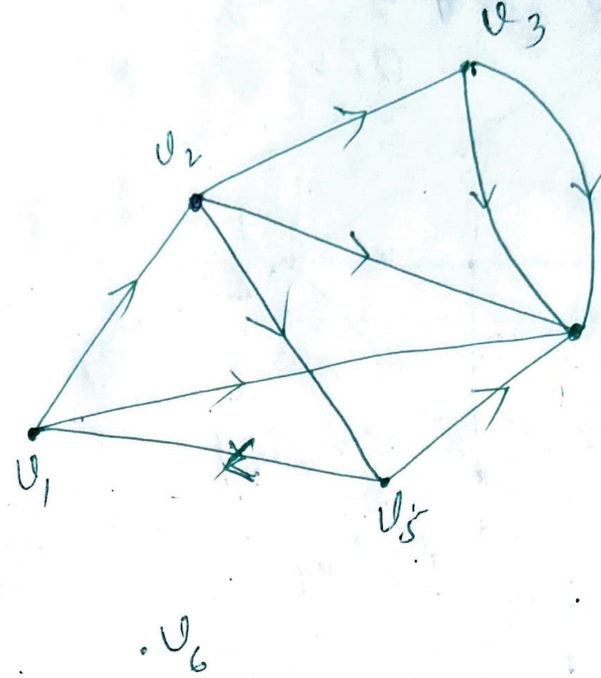
path for DG



if i will go a-1-b-2-c-3-d → path
length = 3

a-1-b-6-e-5-a → closed

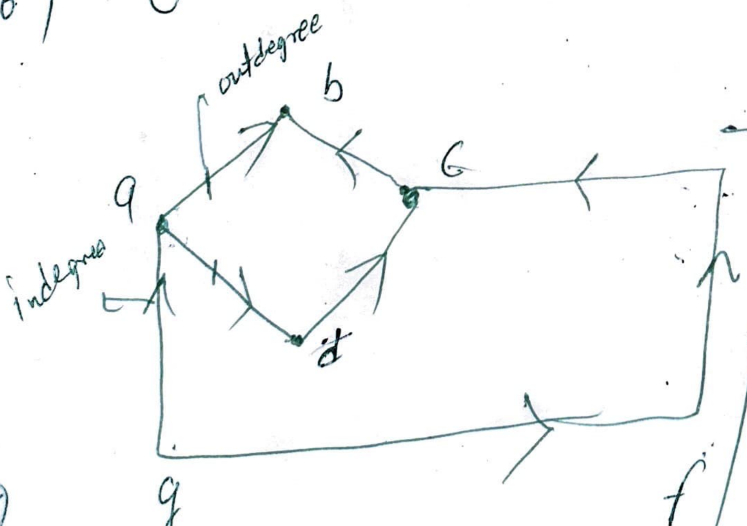
Example of in-degree and out degree of a vertex



out degree means going outwards →

in degree means coming toward ←

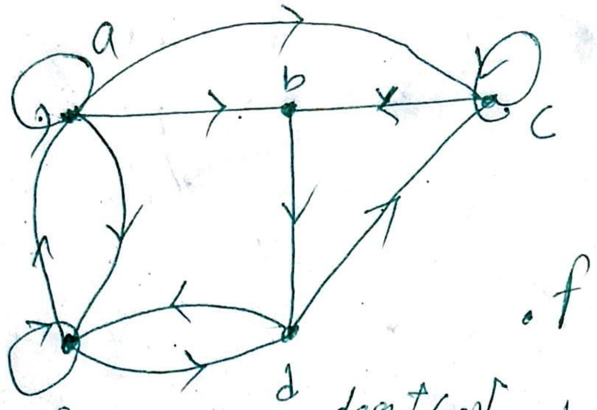
In degree of vertices		Out degree of vertices
v_1	1	2
v_2	1 (Coming to v_2 (which is v_1))	3 → going outwards
v_3	1	2
v_4	5	0
v_5	1	2
v_6	0	0



	in degre	out degre
a	1	2
b	2	0
c	2	1
d	1	1
e	1	1
f	1	1
g	0	2

Thm 2: Let $G = (V, E)$ be a graph with directed edges. Then

$$\sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = |E| = \text{no. of vertices}$$



	$\deg^-(v)$ In degree	$\deg^+(v)$ out degree	$\deg(v)$ Total degree
a	2	4	6
b	2	1	3
c	3	2	5
d	2	2	4
e	3	3	6
f	0	0	
	12	12	12

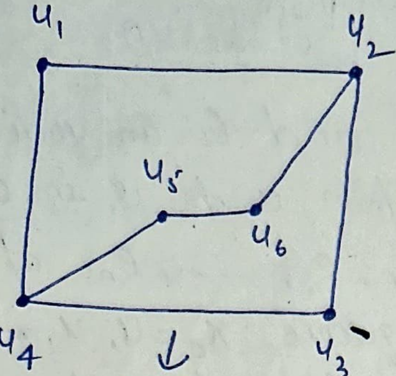
$|E|$ = no. of edges $= 12$

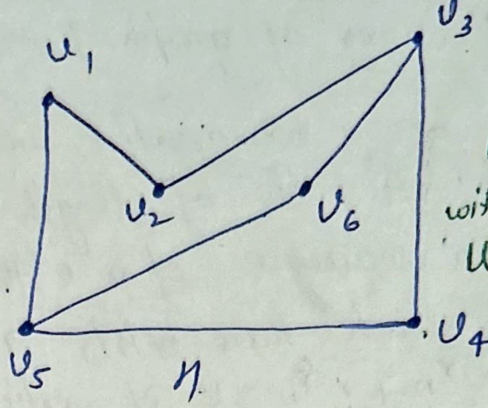
$\sum_{v \in V} \deg^-(v) = 12$

Verify theorem

$\sum_{v \in V} \deg^+(v) = 12$

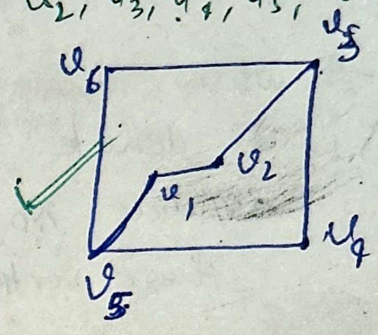
2^{nd} Method check isomorphism by using adjacency matrix.





Here why $f(u_1) = v_6$ bcz $\deg(u_1) = 2$ with adjacent vertices u_2 & u_4 which has degree 3 only v_6 in H has these property

same reason for u_2, u_3, u_4, u_5, u_6



$$A_G = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{array} \begin{array}{c} \begin{array}{cccccc} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$$A_H = \begin{array}{c} \\ v_6 \\ v_3 \\ v_4 \\ v_5 \\ v_1 \\ v_2 \end{array} \begin{array}{c} \begin{array}{cccccc} v_6 & v_3 & v_4 & v_5 & v_1 & v_2 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

bcz $A_G = A_H$

Now

$H \cong G$

$\downarrow$

onto $f : A \to B$

for every $b \in B$ $\exists$ a ele $a \in A$ such that $f(a) = b$

Path: [A path is a sequence of edges that begins at a vertex of a graph and travels from vertex to vertex along edges of graph.] → Defn path for UD

Defn:- Let n be a non-negative integer and G an undirected graph. A path of length n from u to v in G is a sequence of n edges $e_1, e_2, ---, e_n$ of G for which there exists a sequence $x_0 = u, x_1, ---, x_{n-1}, x_n = v$ of vertices such that e_i has, for $i = 1, 2, ---, n$, the endpoints x_{i-1} and x_i.

no need

* When the graph is simple, we denote this path by its vertex sequence $x_0, x_1, --- x_n$ (because listing these vertices uniquely determines the path). (for example if e_1 then endpoints of edge e_1 is x_0 and x_1 $\overset{e_1}{x_0 — x_1}$)

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

* A path or circuit is simple if it does not contain the same edge more than once.

Before path first explain what is a walk. [going to v_1 vertex and end to v_2 then path

Walk means

① Alternating sequence of vertices and edges.

② No edge appears more than once

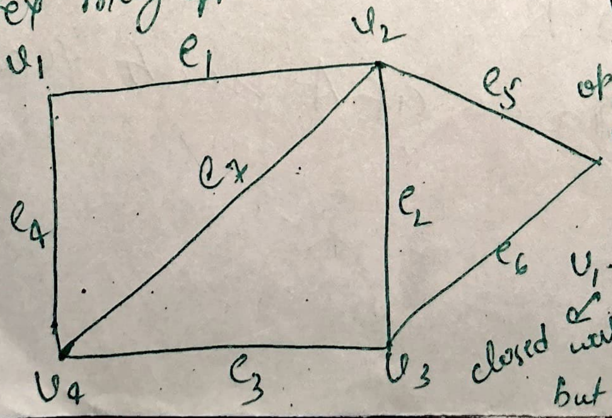
③ Vertex may appears more than once.

but if going to v_1 and end to v_1 circuit]

$v_1 - e_1 - v_2 - e_2 - v_3 - e_3 - v_4$

open walk ↓ no edge appear more than once → first walk

Ex:-



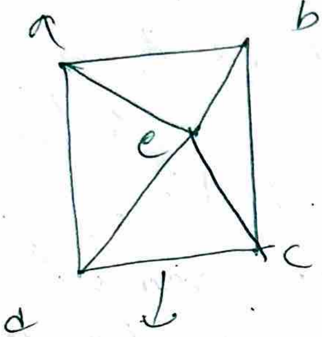
$v_1 - e_1 - v_2 - e_2 - v_3 - e_3 - v_4 - e_4 - v_1$

closed walk ↓ no edge appear more than once but vertex appear more than one

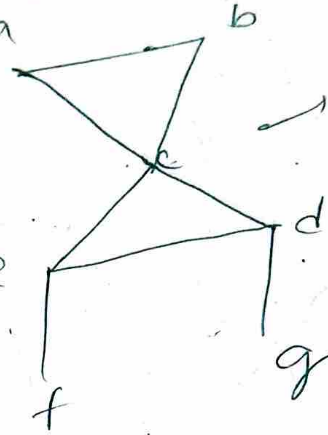
Connected or (path connected): An UG is called connected if there is a path betⁿ every pair of distinct vertices of the graph.

* An UG that is not connected is called disconnected.
* We disconnect a graph by removing vertices or edges or both.

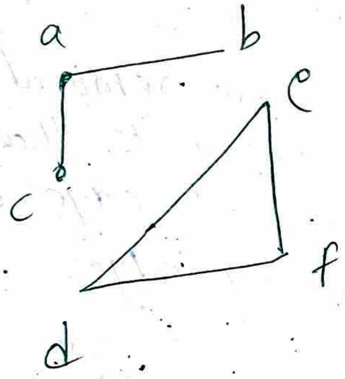
Ex:



connected graph



→ connected



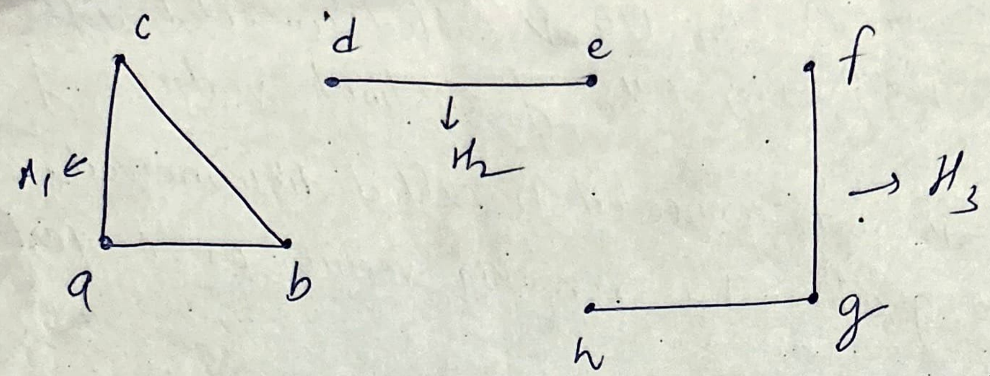
→ disconnected → there is no path betⁿ c to d, a to e or b to e, b to d

Theorem:- There is a simple path betⁿ every pair of distinct vertices of a connected undirected graph.

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

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

A disconnected graph which has connected components (or elements)



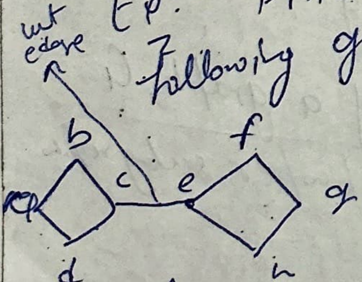
This is graph H where H_1, H_2 and H_3 are connected components of H (Here H_1, H_2 ad H_3 are three disjoint connected subgraphs of H)

Cut vertices (or articulation points):- The removal from a graph of a vertex and all incident edges produces a subgraph with more connected component. Such vertices are called cut vertices.

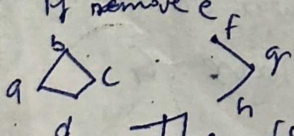
Cut edge or bidge:- An edge whose removal produces a graph with more connected components than in the original graph is called a cut edge or bridge.

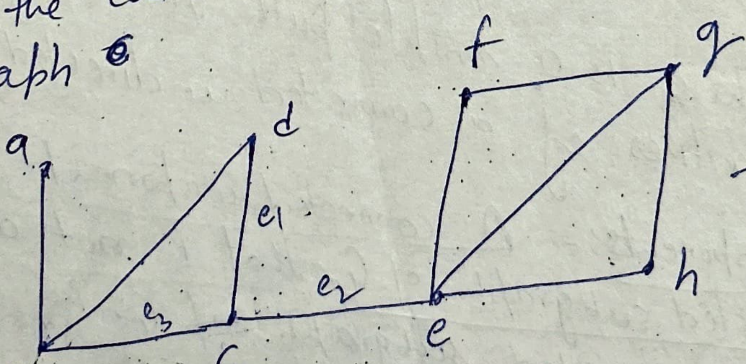
Ex:- Find the cut vertices and cud edges in the following graph

cut edge



cut vertex c, & e.
if remove e

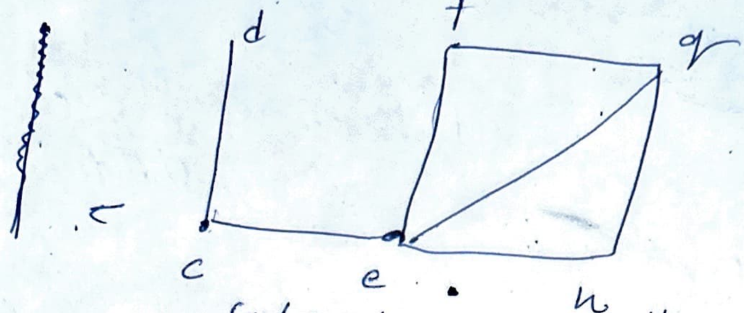




cud vertice
→ b, c, e
if I remove c then incident edge e_1, e_2 & e_3 automatically remove
⇒ discon
If I remove b

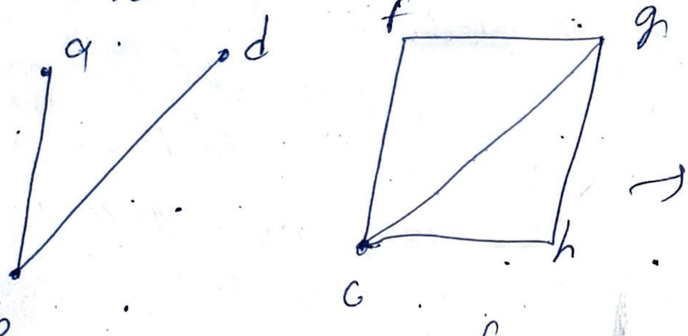
The cut vertices of G are b, c and e. The removal of one of these vertices (and its adjacent edges) disconnects the graph. The cut edges are $\{a, b\}$ and $\{c, e\}$. Removing either one of these edges diconnects G.

① Let I remove b & its incidnt edges then graph will be



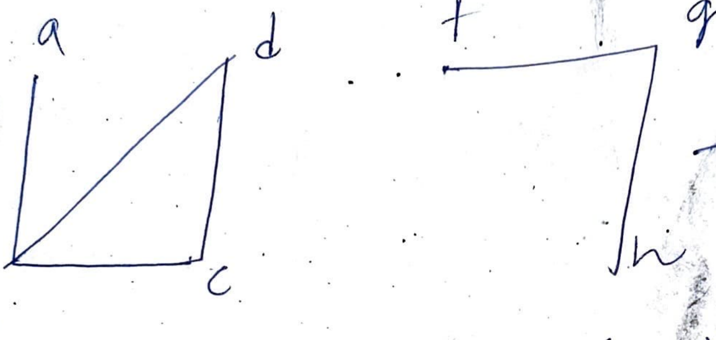
subgraph of G with connected componet

If i remove c then



→ subgraph of G

remove e

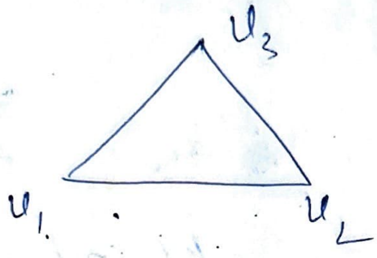


→ subgraph

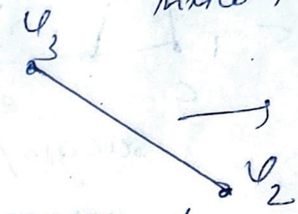
② Cut edge {c, e} & {a, b} (ask)

* Not all graphs have cut vertices

Eg K_3 ,



Let remove v_1 and its incident edge



→ & it is connected

Same reason for v_2 & v_3

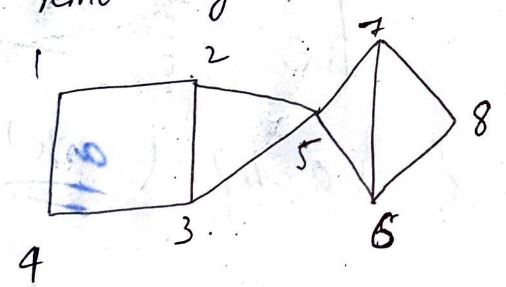
$K_n, n \geq 3$, has no cut vertices.

<u>Non seperable Graphs</u> :- Connected graphs without cut vertices are called non seperable graps. (i.e we can't find a disconneted graph).

<u>Cut-set</u> :- In a connected graph G, a cut-set is a set of edges whose removal from G leaves G disconnected, provided removal of no proper subset of these edges disconnects G.

{cut set collection of cut edge}.

Eg: 2



{(2,5), (3,5)} → cut set

{(1,2), (2,3), (3,5)} is also a cut set

→ The set {(1,2), (2,3), (3,5), (2,5)} is not a cut-set bcz a proper subset of this is a cut set.