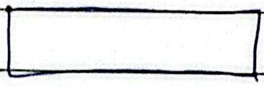
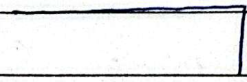
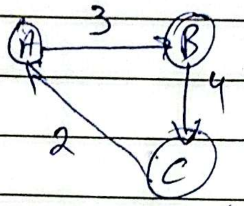
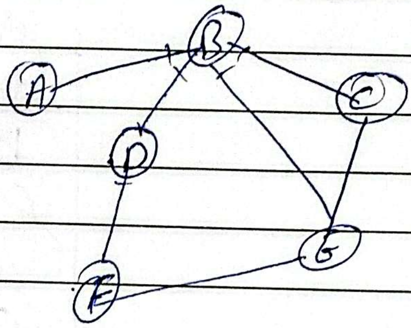
Date:

Edges  4 edges  2 edges.



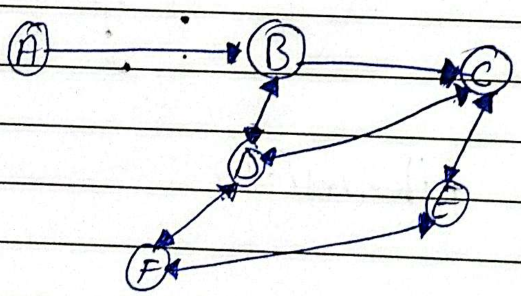
weighted.

→ directional ↔ bidirectional — un-directional.



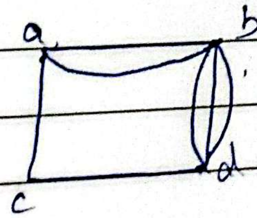
degree B = 4
degree D = 2

Directional Graph
In-degree → where data is recieved.
Out-degree → where data is givened



Out degree A = 1
B = 2
C = 2
D = 3
E = 2
F = 2

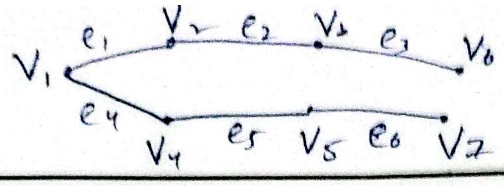
In degree A = 0
B = 2
C = 3
D = 3
E = 2
F = 2

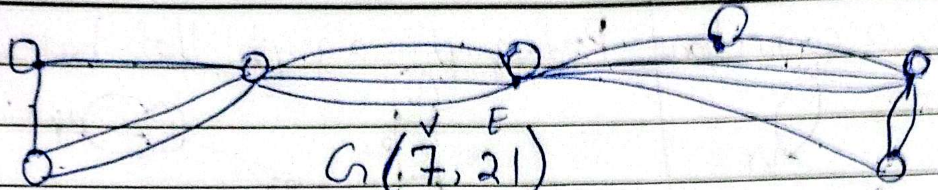


degree A = 3
C = 2

Degree B = 5
D = 4

Date:





V E
G(7, 21)

Paths & Circuits.

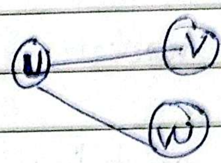
U & V & adjacent where vertices are connected through one degree.

Adjacents ⟶ vertices

Incidents ⟶ edges.

Pendent vertex degree (V) = 1

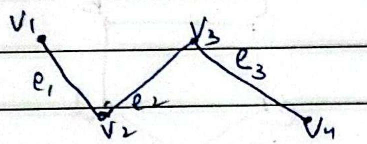
Isolated vertex degree (V) = 0



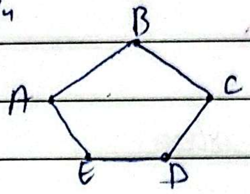
adjacent vertex
(U, V) (U, W)

Walk - A walk from a to b is a finite alternating sequence of adjacent vertices. and edges of G

$V_0 e_1, V_1 e_2 \cdots V_{n-1} e_n$



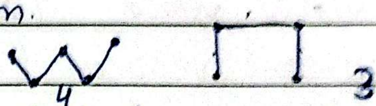
Closed walk



Loop ⟶ multiple graph.

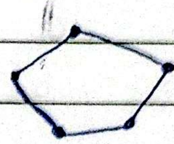
Circuit :- A circuit is a closed walk that does not contain a repeated edge, this is a circuit a walk of form.

Paths :- sequence of vertices connected by edge.



length of path ⟶ length of edges und

Cycle :- A path that starts & ends at same vertex

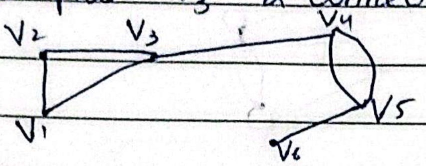


Types of Graphs :-

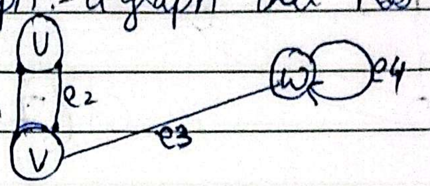
Simple , Multiple.

Date:

(a) is no path v_3 2 connections



Pseudograph :- a graph that has loop.



Theorem 1:-

$$2e = \sum_{v \in V} v$$

[* this is for undirected graph]

Total no of vertices = 2 x no of edges.

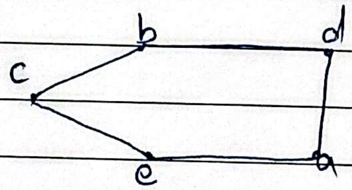
Theorem 2:-

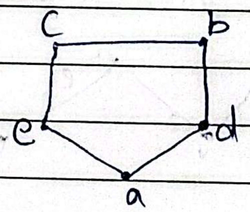
An undirected graph has even no of vertices with odd degree.

[* This is also for undirected graph]

* If the sum is odd then there would be no graph according to theorem 2 it should always be even.

(d) a c, b, d, a, e, c



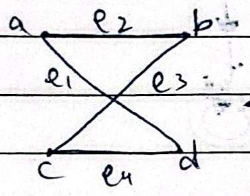


Adjacent vertices → nonconnected vertices
Incident edges →
Path , circuit , cycle
isolated vertices (degree 0)
pendent vertices (degree 1)
degree at loop = 2

(2) a = 6 , d = 5
e = 3 , b = 6
c = 6.

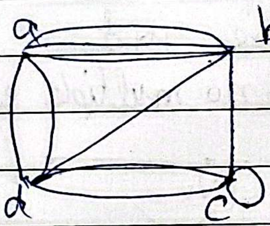
Date: Incident matrix

rows will have vertices
column will have edges.



	e_1	e_2	e_3	e_4
a	1	1	0	0
b	0	1	1	0
c	0	0	1	1
d	1	0	0	1

Adjancy list



Pseudograph or multiple graph

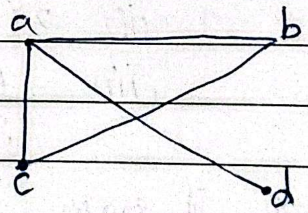
vertex	adjancy list
a	b, d
b	a, d, c
c	b, d, c
d	a, b, c

$$\begin{array}{c|cccc} & a & b & c & d \\ \hline a & 0 & 3 & 0 & 2 \\ b & 3 & 0 & 1 & 1 \\ c & 0 & 1 & 1 & 2 \\ d & 2 & 1 & 2 & 0 \end{array}$$

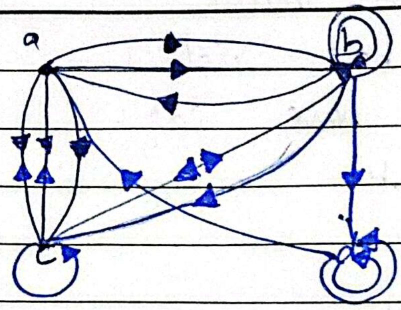
- Adjacent Matrix
  - ↳ simple
  - ↳ multiple
  - ↳ directed Graph.
- Adjaent List
- Incident Matrix

①

	a	b	c	d
a	0	1	1	1
b	1	0	1	0
c	1	1	0	0
d	1	0	0	0

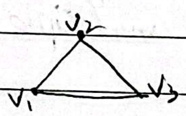


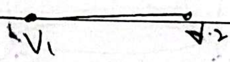
	a	b	c	d
a	0	2	3	0
b	1	2	2	1
c	2	1	1	0
d	1	0	0	2

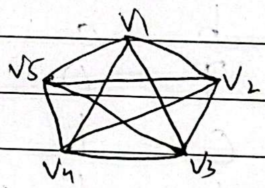


Complete Graph.

Starts & end at same point.
Its a simple path (~~too~~ no multiple edges)

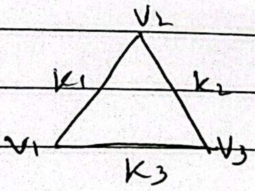






$\boxed{\text{Exercise}}$ For complete graph k_n find.

i- degree of each vertex $= n-1$
ii- total degree $= n(n-1)$
iii) no of edges $= \frac{n(n-1)}{2}$

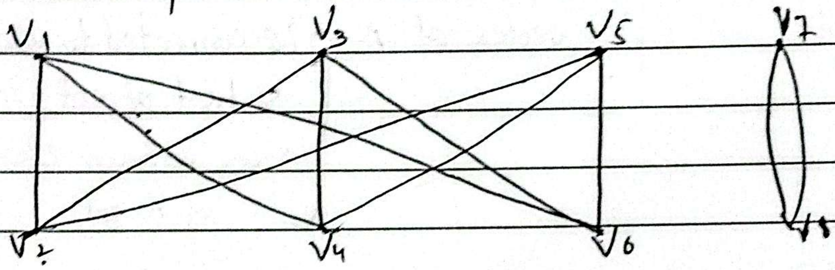


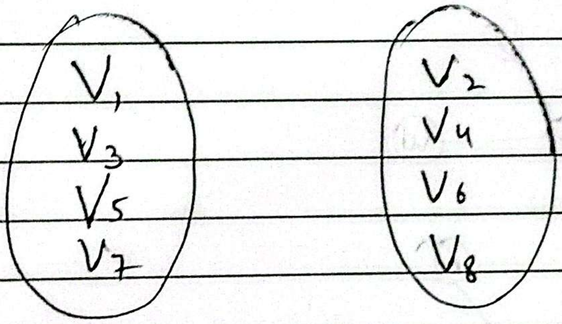
(i) 2
(ii) $3(3-1) = 6$
(iii) $\frac{6}{2} = 3$

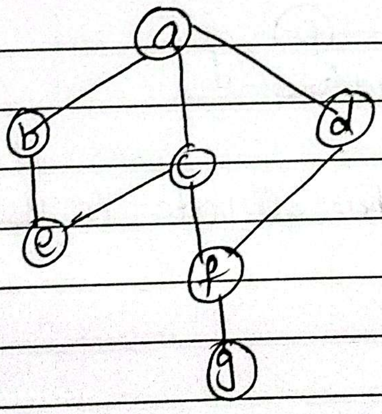
Graph is regular if every vertex has degree k.
Complete graph can be regular.
but not all complete graph are regular.

Date:

- Complete Graph
- Regular Graph
- Bipartite Graph.

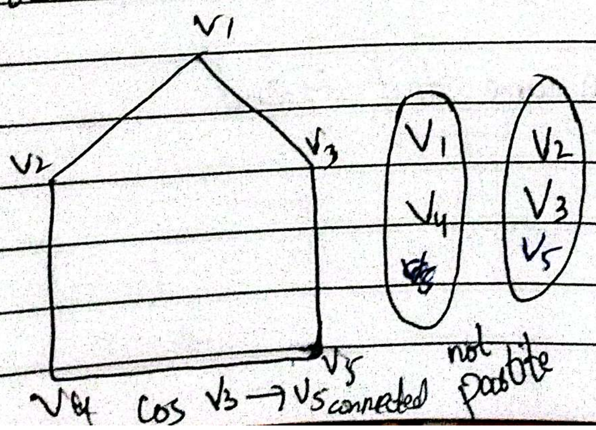




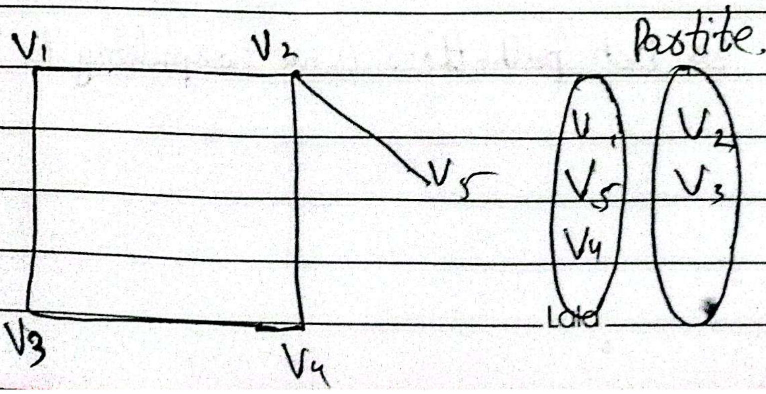


a	b
e	d
f	c
	g

Bipartite
real time ex: Uber, timetable, Register allocation, Map coloring, sudoku



cos V3 → V5 connected not partite



Date:

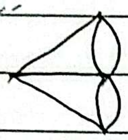
Complete graph:-(simple graph, every vertex is connected to each other)

Regular Graph ⇒ every vertex has same degree.

Bipartite Graph ⇒ Set is divided into two sets A & B. in which no vertex of A → is connected to vertex of B.

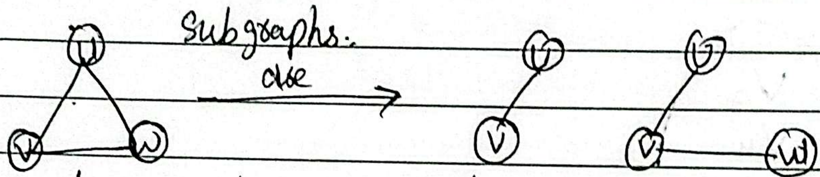
Euler Circuit:

has no

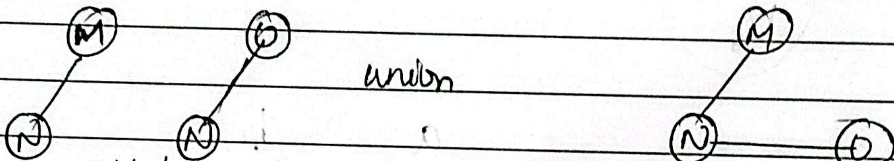


subset of set
→ some contain element of a set

Subgraphs.
are



can do union here too intersection too



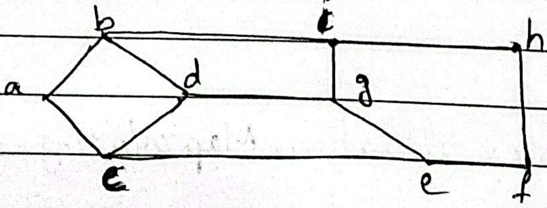
In weighted graph, edges have a Cost/value/amount.

Main circuit Vs Euler circuit.

Euler Theorem

here all vertex is used.

1) Every vertex has even
2) No edges is repeated
3) Every edge must be utilized.



This is euler circuit.

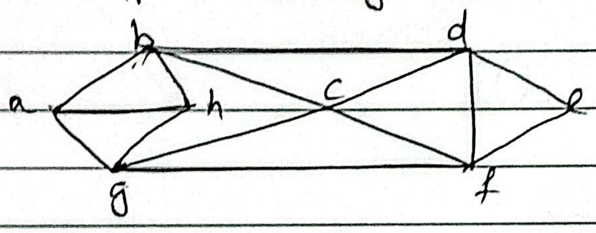
~~Note~~ Each path: there is no compulsory to have same start & end.

Lala

Date:

Hamiltonion Circuits:

a hamiltonion circuit for G is simple circuit that includes every vertex of G, that is hamiltonion for G is a sequence of adjacent vertices & distinct edges in which every vertex of G appears exactly one.

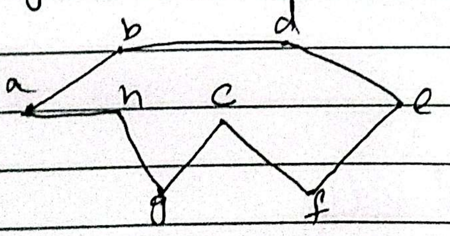


abdefcgha   bcdefghab

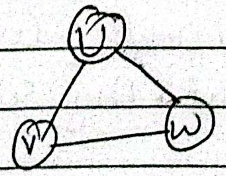
Proposition:

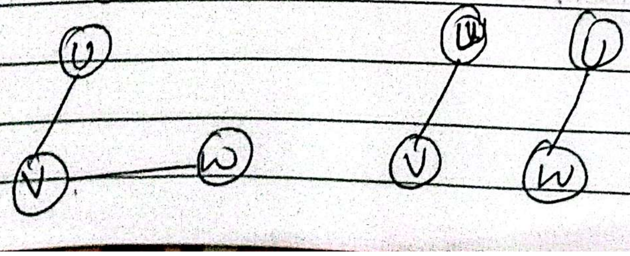
a graph G has a hamiltonion circuit then G has a subgraph with the following properties:

1) H has contain every vertex of G
2) H is connected
3) H has no of edges as vertices
4) Every vertex of H has degree 2



Every vertex must be utilized
No repetition of vertices



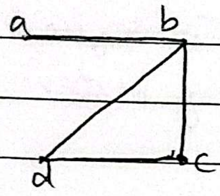


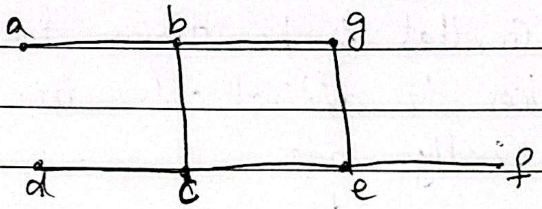
$H_1 = \{U, V\}$
$H_2 = \{(U,W)(V,W)\}$

Date:

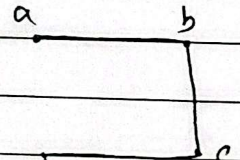
$H_1 \cap H_2$ = No Graph

$H_1 \cup H_2$ = {(u,v) (v,w) (v,w)}

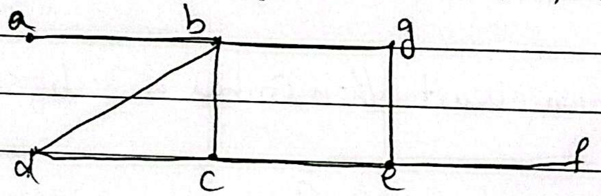




$G_1 \cap G_2$ = {(a,b) (b,c) (c,d)}



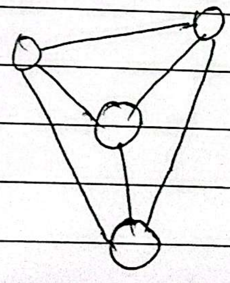
$(G_1 \cup G_2)$ = {(a,b) (b,g) (b,d) (d,c) (c,e) (e,f)}



Planer Graph

3rd, 2nd mostly planer

If edges are not crossing each other then planner graph.



4 vertices

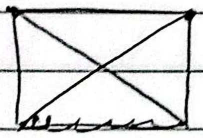
6 edges

its planer graph

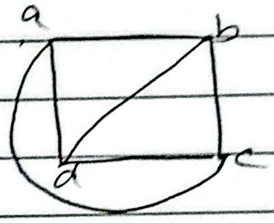
& no intersection of edges

A planer graph is graph that can be drawn in plane without crossing there no 2 edges intersect geometrically except at a vertex to which other are incident. Any such drawings is called a planner graph.

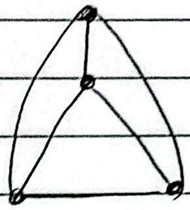
Date:



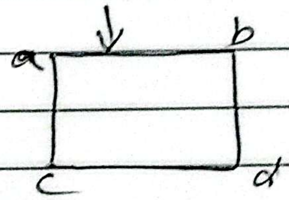
no a plane



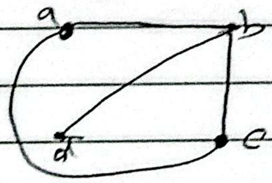
planes



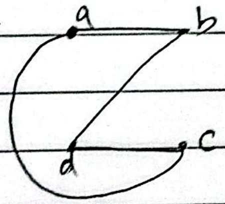
planes



Hamiltonian



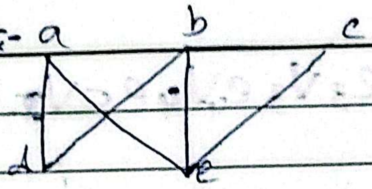
Euler



C+

Worst

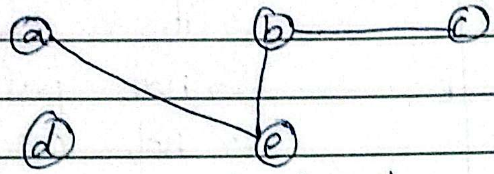
Date: **Assignment no 1:-**



Areeba
B-7083

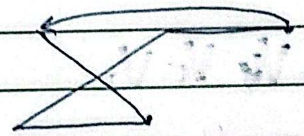
Q# 1:

1-(a) a, a, b, c, b



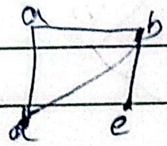
not simple   not circuit ..

b) a, e, a, d, b, c, a



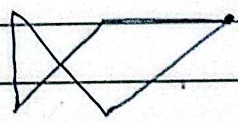
no circuit
not simple path

c) c, b, a, d, b, e



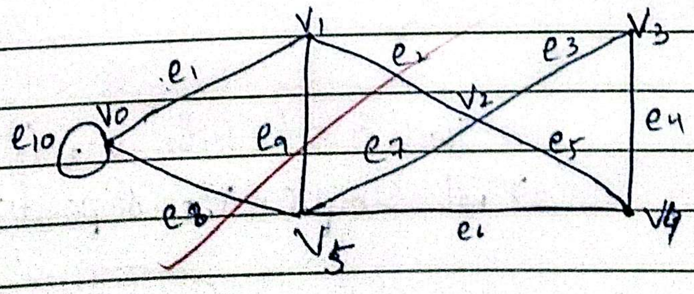
not simple path
no circuit

d) c, b, d, a, e, c



not simple path
circuit
as it start & end on c

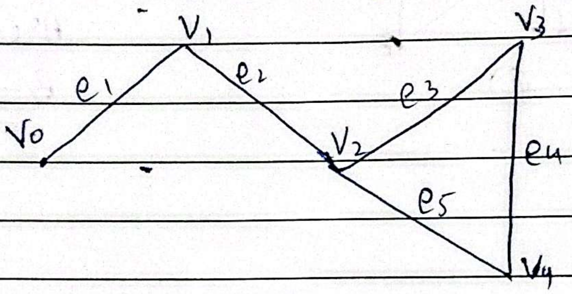
Q# 3:



Lala

Date:

(a) $V_1\ e2\ V_2\ e_3\ V_3\ e_4 V_4\ e_5\ V_2\ e_2\ V_1 e_1 V_0$

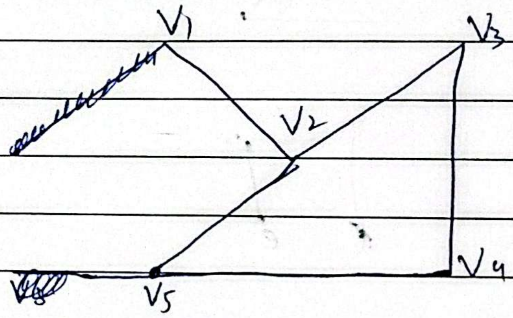


→ its a walk

→ not path (repeated vertices)

→ not circuit (not closed)

2- $V_1 V_2 V_3 V_4 V_5 V_2$

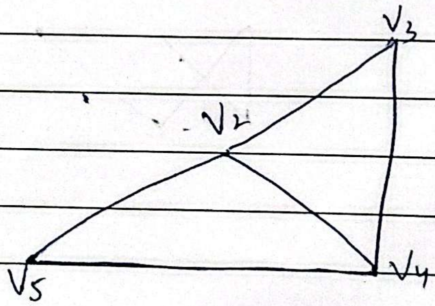


→ its a walk

→ not simple path (V_2 repeat)

→ not circuit

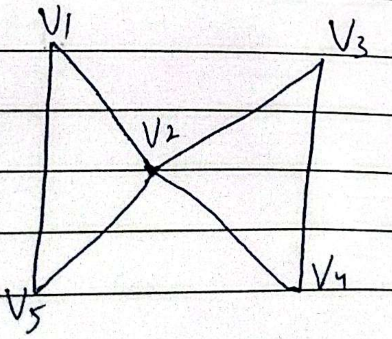
3- $V_4 V_2 V_3 V_4 V_5 V_2 V_4$



→ a closed walk circuit (start & end at V_4)

→ not simple V_2 repeated vertices

4) $V_2 V_1 V_5\ V_2\ V_3\ V_4\ V_2$



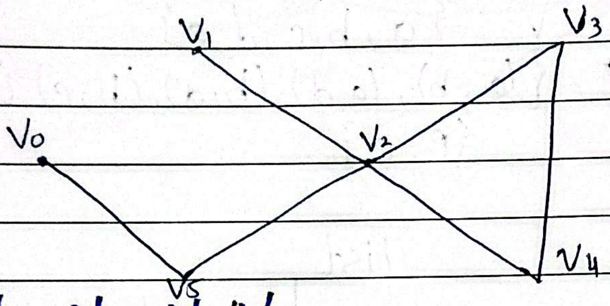
→ Start & end with V_2 (a closed walk)

→ not simple V_2 repeats

Lala

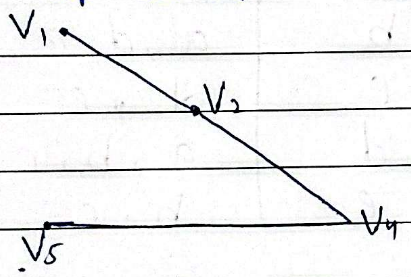
Date:

5) $V_0\ V_5\ V_2\ V_3\ V_4\ V_2\ V_1$



→ walk
→ not circuit (not closed)
→ not simple path (Repeat vertices)

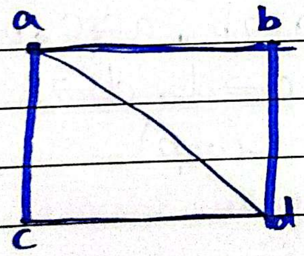
6- $V_5\ V_4\ V_2\ V_1$



→ Its walk
→ Its simple
→ not closed.

Q2: Adjancy matrix & list. Interpret & Justify.

1-



$V = \{a, b, c, d\}$
$E = \{(a,b), (b,d), (d,c), (c,a), (a,d)\}$

Matrix

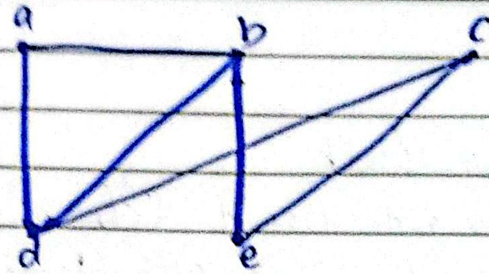
	a	b	c	d
a	0	1	1	1
b	1	0	0	1
c	1	0	0	1
d	1	1	1	0

Adjancy list.

	(Vertex)	A.V
a:	a	{b, c, d}
b:	b	{a, d}
c:	c	{a, d}
d:	d	{a, b, c}

Undirected

Date:

2. 

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

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

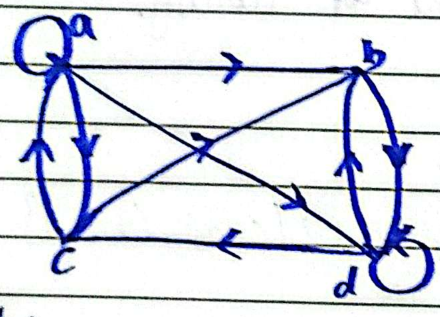
Matrix

	a	b	c	d	e
a	0	1	0	1	0
b	1	0	0	1	1
c	0	0	0	1	1
d	1	1	1	0	0
e	0	1	1	0	0

List

a	b, d
b	a, d, e
c	d, e
d	a, b, c
e	b, c

Undirected Graph

3. 

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

E: $a \to a$ (loop), $a \to b$, $a \to c$, $b \to a$
$b \to c$, $b \to d$, $c \to a$, $c \to d$, $d \to b$
$d \to d$ (loop)

Matrix

	a	b	c	d
a	1	1	1	1
b	0	0	0	1
c	1	1	0	0
d	0	1	1	1

List

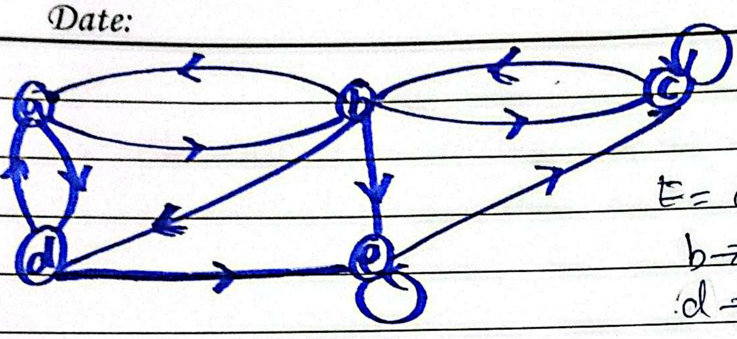
a	a, b, d, c
b	d
c	a, b
d	b, c, d

directed Graph

Lala

Date:

4.



V= a,b,c,d,e

E= a→b, d→d, a→e, b→c

b→e , c→c(loop) d→a

d→e , e→d, e—e(loop)

Matrix

	a	b	c	d	e
a	0	1	0	1	0
b	1	0	1	1	1
c	0	1	1	0	0
d	1	0	0	0	1
e	0	0	1	0	1

List

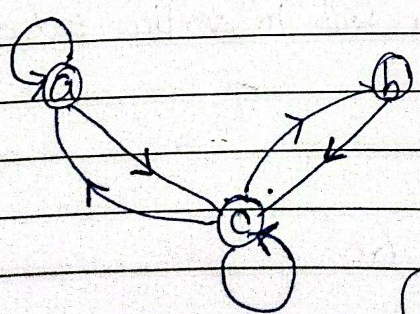
a	b, d
b	a, c, d, e
c	b, c
d	a, e
e	c, e.

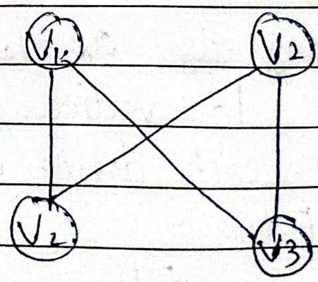
directed Graph

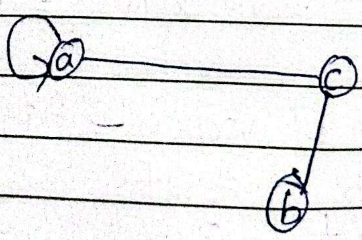
Q# 2:

1- Draw Graph by given adjancy list.

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



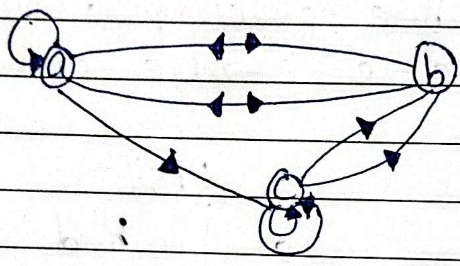




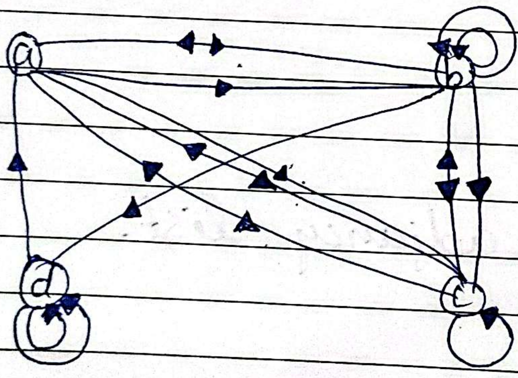
Lala

Date:

23-
$$\begin{array}{c|ccc} & a & b & c \\ \hline a & 1 & 2 & 1 \\ b & 2 & 0 & 0 \\ c & 0 & 2 & 2 \end{array}$$



24-
$$\begin{array}{c|cccc} & a & b & c & d \\ \hline a & 0 & 2 & 3 & 0 \\ b & 1 & 2 & 2 & 1 \\ c & 2 & 1 & 1 & 0 \\ d & 1 & 0 & 0 & 2 \end{array}$$



2- How many edges ~~are~~ there in a graph with 10 vertices each of degree 6?

10 vertices

each degree of six

Edges

Edges $= 10 \times 6 = 60$

Acc to theorem

degree of vertices $= 2t$

$60 = 2t$

$t = 30$

Date:

Q # 3

(i) Graph with four vertices of degrees 1, 2, 3 & 3

theorem :- Sum of degree must be even.

$1+2+3+3 = 9$

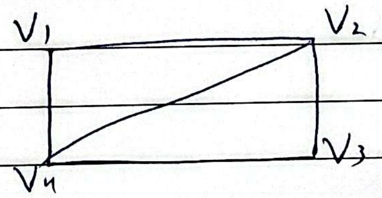
voilates theorem, so no graph

(ii) Graph with four vertices of degrees 1, 2, 3 & 4

theorem $1+2+3+4 = 10$

So even & edge count will be $= 5$

then let 4 vertices be V_1, V_2, V_3, V_4

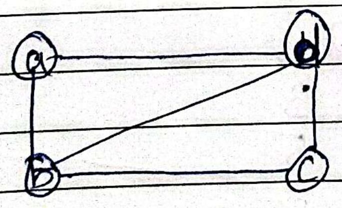


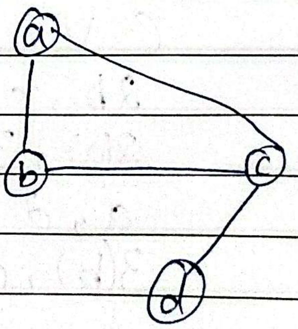
But in this way it wouldn't be a simple graph.

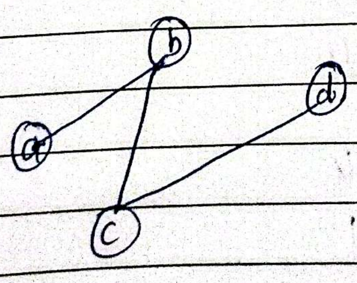
(iii) Simple Graph with 4 vertices of degree 1, 2, 3, 4

Impossible (degree 4 can't occur in simple 4-vertex graph)

5 edges





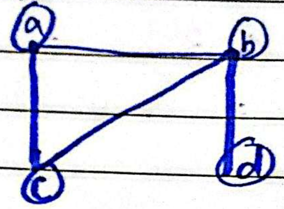


Lala

Date:

Q # 4,

3-

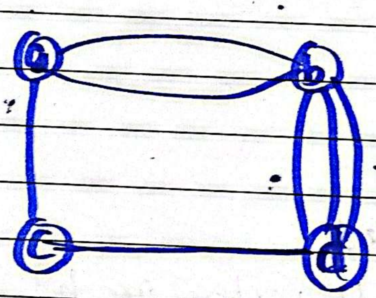


Vertex	connected vertices	degree
a	b, c	2
b	a, c, d	3
c	a, b,	2
d	b	1

Sum of degree = 2+3+2+1 = 8

2 x no of edges = 2 x 4 = 8

4-



Vertex	C. V	Degree
a	2b, c	3
b	2(a), 3(d)	5
c	a, d	2
d	3(b), c	4

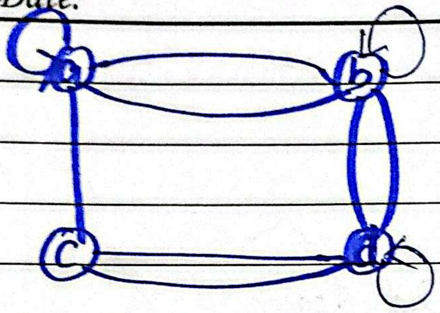
Sum of degree = 3+5+2+4 = 14

2 x no of edges = 2x7 = 14

Lala

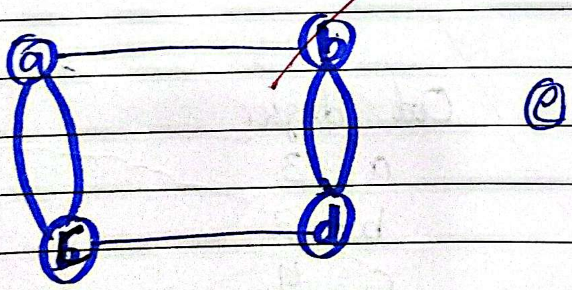
Date:

5



Vertex	Connected Vertex	Degree
a	a, 2(b), c.	4
b	2(a), b, 2(d)	5
c	a, 2(d)	3
d	2(b), 2(c), d	5

6)



Vertex	C.V	Degree
a	b, 2(c)	3
b	a, 2(d)	3
c	2(a), d	3
d	2(b), c	3
e		0

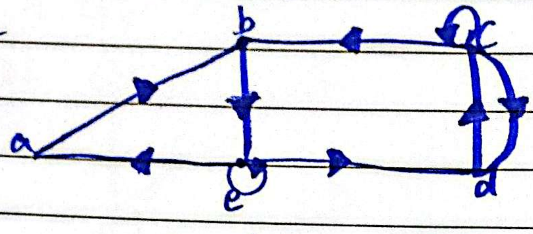
Sum = 3+3+3+3+0 = 12

2x no of edges = 6x2 = 12 correct.

Lala

Date:

7.



In degree

a = 1
b = 2
c = 2
d = 2
e = 2

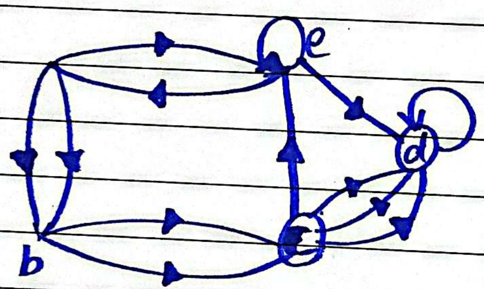
9

Out degree

a = 1
b = 1
c = 3
d = 1
e = 3

9

8.



In degree

a = 1
b = 2
c = 2
d = 5
e = 3

13

Out degree

a = 3
b = 2
c = 4
d = 1
e = 3

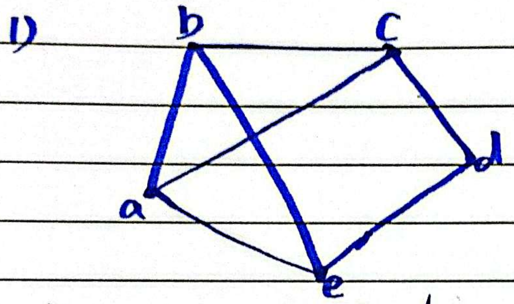
13

Best

Date: Assignment 2 Part 2:

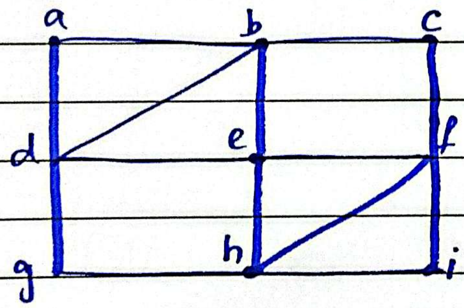
Q#2: Euler Circuit.

1)



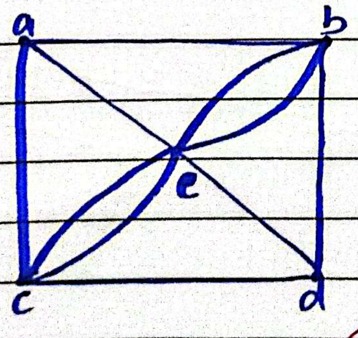
Here b got 3 degree so its not euler as its odd
All ~~vertices~~ vectors should be even.

2)



degree of all vectors are even
so its euler.

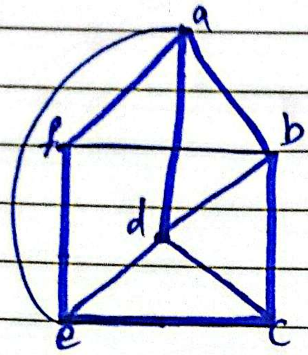
3)



Not all vertices got even degree so its not euler.

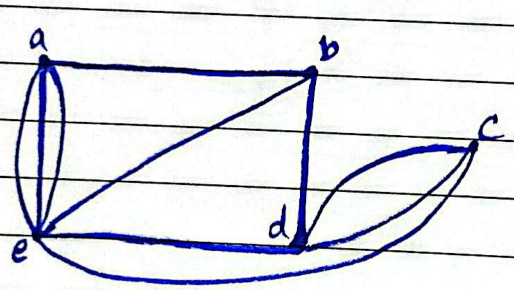
Date:

4-



C have odd degree so its not euler.

5-

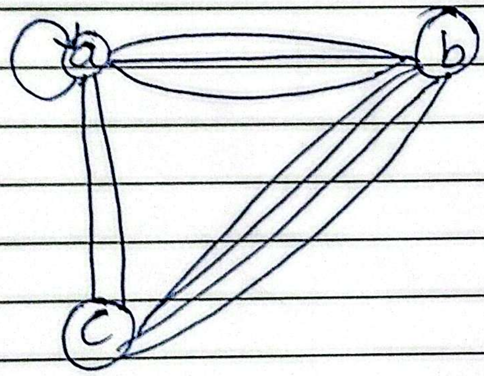


C have odd degree so its not euler.

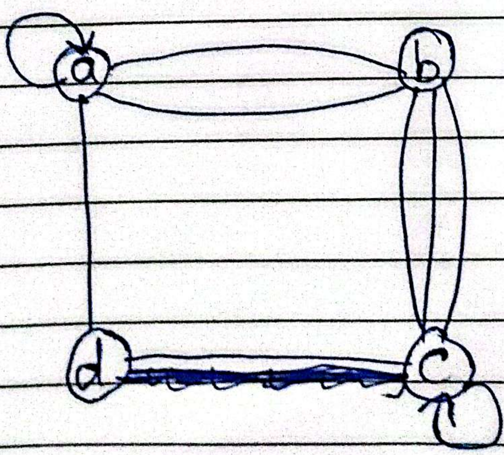
Date: Assignment no 2: 04/11/25

Q# 3 draw an undirected graph by given adjacent matrix.

16 -
$$\begin{bmatrix} 1 & 3 & 2 \\ 3 & 0 & 4 \\ 2 & 4 & 0 \end{bmatrix}$$



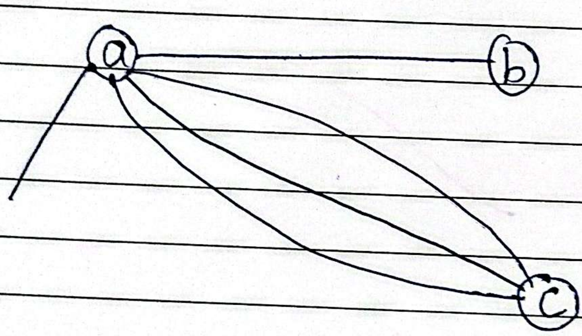
17 -
$$\begin{bmatrix} 1 & 2 & 0 & 1 \\ 2 & 0 & 3 & 0 \\ 0 & 3 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

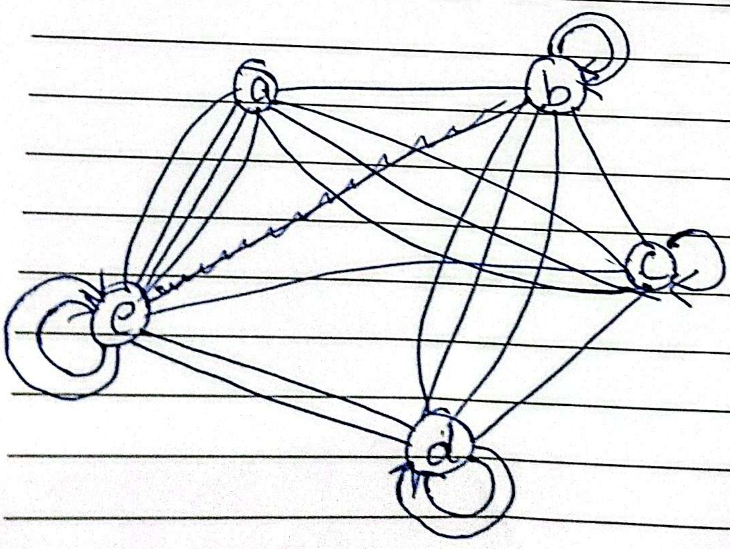


Date:

(12)

$$\begin{bmatrix} 0 & 1 & 3 & 0 & 4 \\ 1 & 2 & 1 & 3 & 0 \\ 3 & 1 & 1 & 0 & 1 \\ 0 & 3 & 0 & 0 & 2 \\ 4 & 0 & 1 & 2 & 3 \end{bmatrix}$$

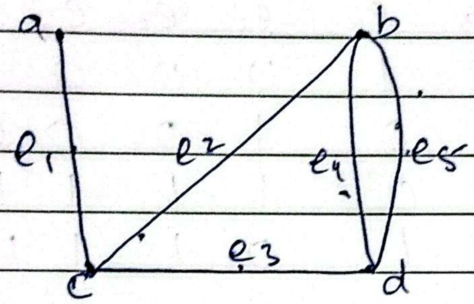




Date:

Q# Use incidence matrix to represent graph

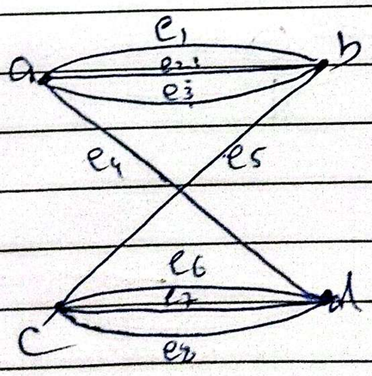
13



	e_1	e_2	e_3	e_4	e_5
a	1	0	0	0	0
b	0	1	0	1	1
c	1	1	1	0	0
d	0	0	1	1	1

	e_1	e_2	e_3	e_4	e_5
a	1	0	0	0	0
b	0	1	0	1	1
c	1	1	1	0	0
d	0	0	1	1	1

(14)

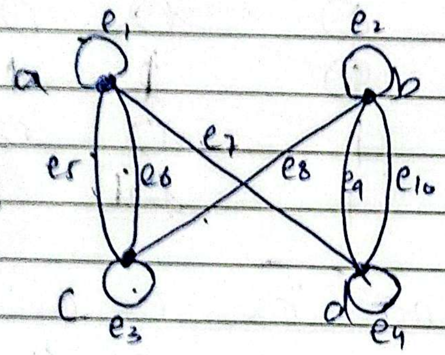


	V_1	V_2	V_3	V_4
V_1	1	3	0	1
V_2	3	0	1	0
V_3	0	1	0	3
V_4	1	0	3	0

Date:

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
a	1	1	1	1	0	0	0	0
b	1	1	1	0	1	0	0	0
c	0	0	0	0	1	1	1	1
d	0	0	0	1	0	1	1	1

15.



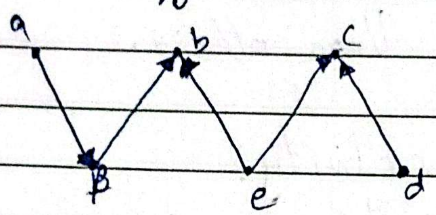
	V_1	V_2	V_3	V_4
V_1	1	0	2	1
V_2	0	1	1	2
V_3	2	1	1	0
V_4	1	2	0	1

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}
a	1	0	0	0	1	1	1	0	0	0
b	0	1	0	0	0	0	0	1	1	1
c	0	0	1	0	1	1	0	1	0	0
d	0	0	0	1	0	0	1	0	1	1

Date: Assignment no 3 Average

Q# 1: Strongly connected or not?

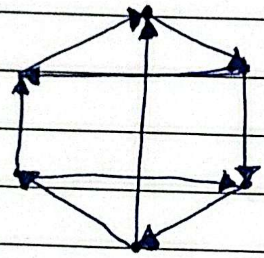
a)



weakly connected graph.

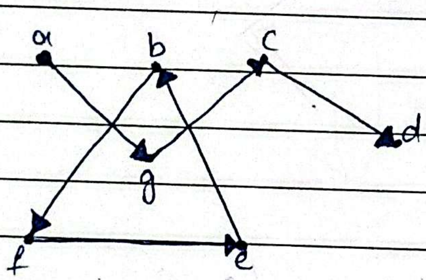
18/11/25

b)



strongly connected graph

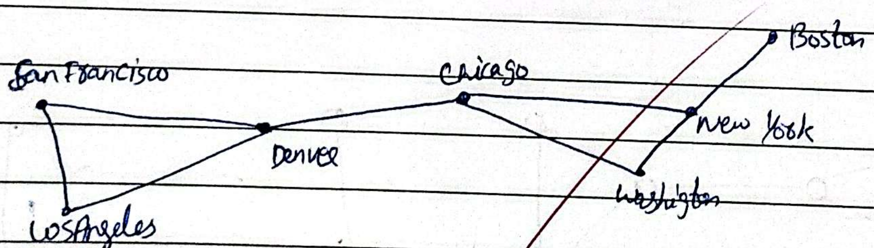
c)

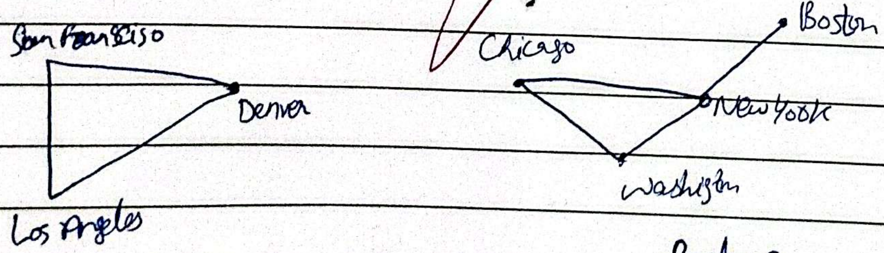


weakly connected.

Q#2:

a



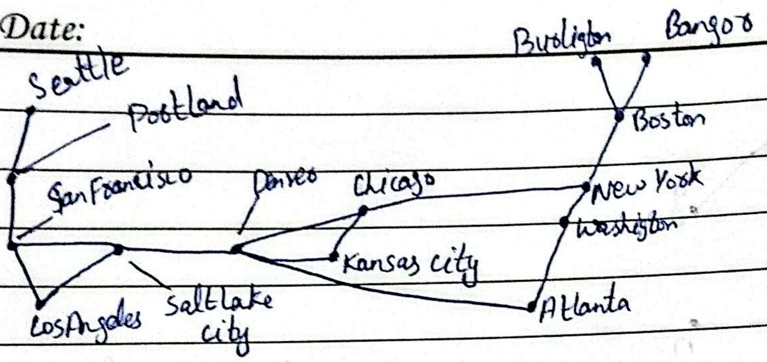


Denver - Chicago
New York - Boston.

Backup
Los angeles - Denver
Denver - Chicago

Date:

b)

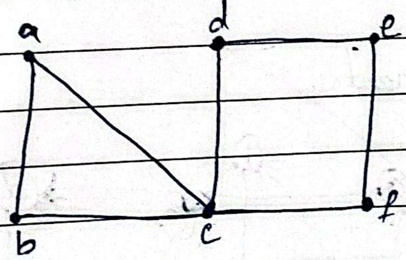


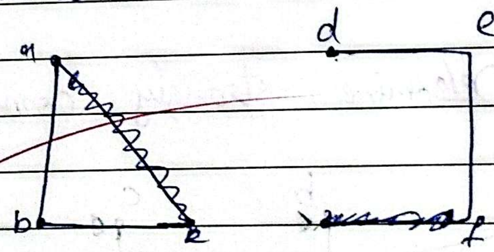
Backup
Chicago – Kansas city
Boston – Bangor
Seattle – Portland.

Seattle – Portland, Portland – San Francisco, Salt Lake city – Denver, New-York – Boston, Boston Burligton, Boston – Bangor.

Q#3:- Determine all the cut vertices

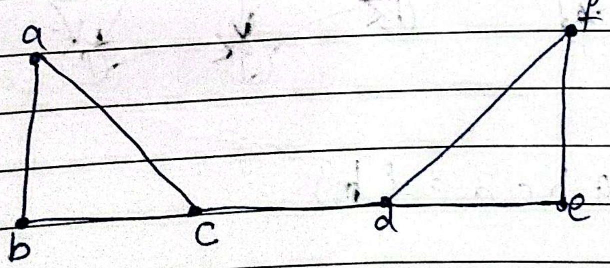
31.

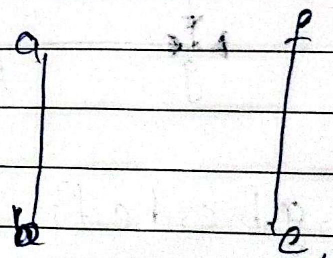




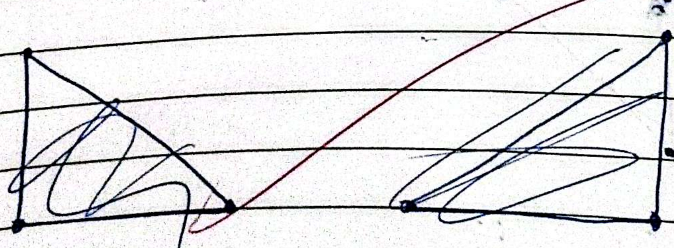
remove c

32.



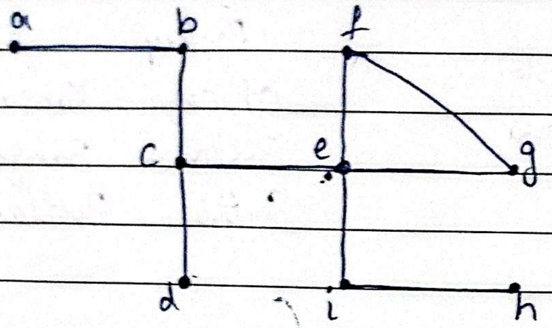


remove c, d

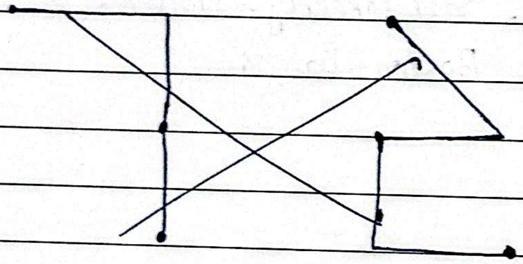


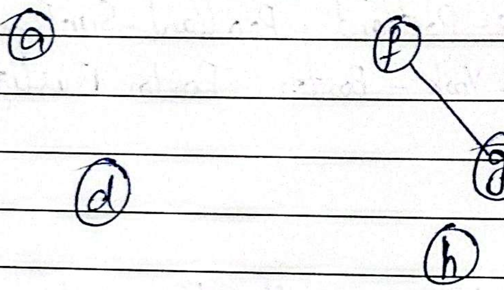
Date:

33-



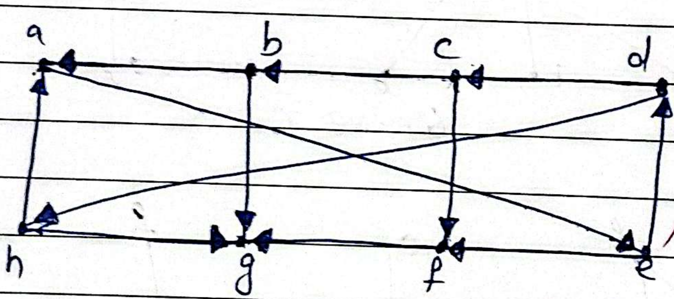
remove b, c, e, i

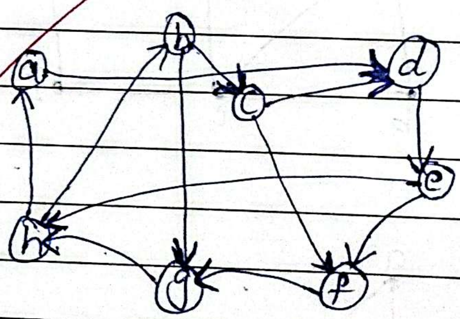




Qn 4:- Determine strongly connected components.

1-

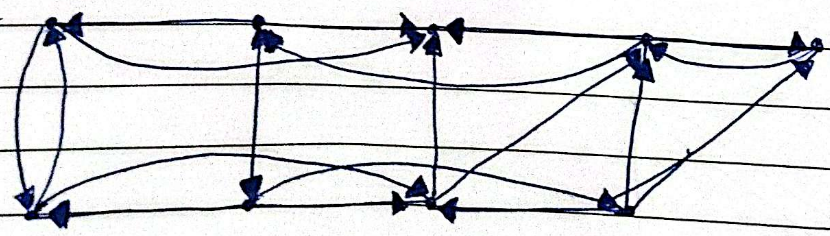




{a, h, g, b, c, d, e, f}

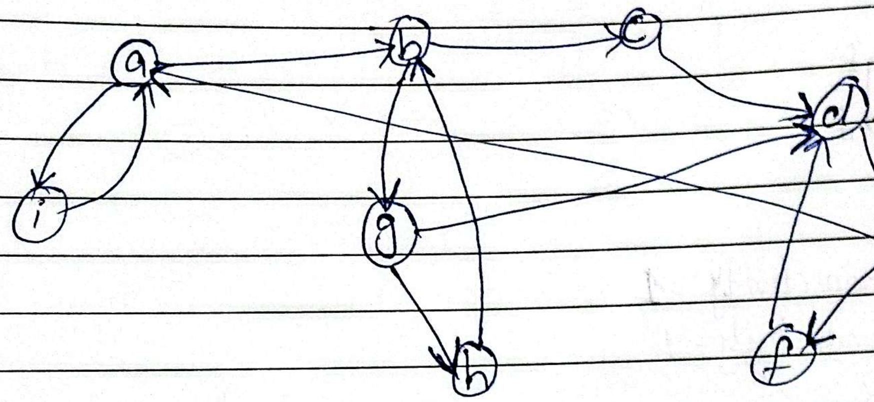
{a, b, c, d, e, f, h}

2-



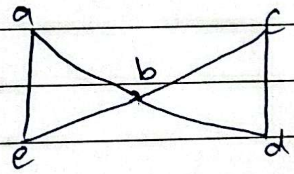
{a, i} {b, g, h} {c} {d, e, f}

Date:



Q#5:

(a)



$K(G)$ = vertex connectivity = 2

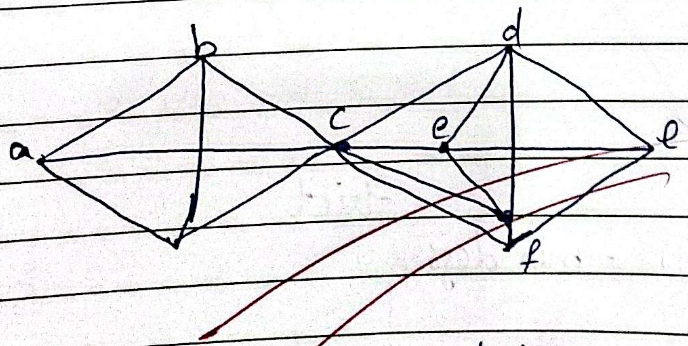
$\lambda(G)$ = Edge connectivity = 2

min deg(V) = Minimum degree = 2

Inequality (strict)

$K(G) = \lambda(G) = \min \deg(V)$ 2

(b)



$K(G)$ = vertex connectivity = 2

$\lambda(G)$ = Edge connectivity = 2

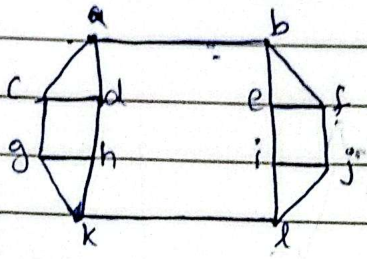
min deg(V) = 3

Inequalities

$K(G) = \lambda(G) < \min \deg(V)$

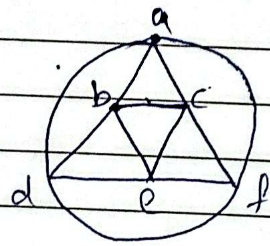
Date:

c)



$K(G)$ = vertex connectivity = 1
$\lambda(G)$ = Edge connectivity = 1
min degree = 2
Inequalities $K(G) = \lambda(G) < deg(v)$
So
$K(G) \le \lambda(G)$ is equality and
$\lambda(G) \le$ min $deg(v)$ is strict.

d)

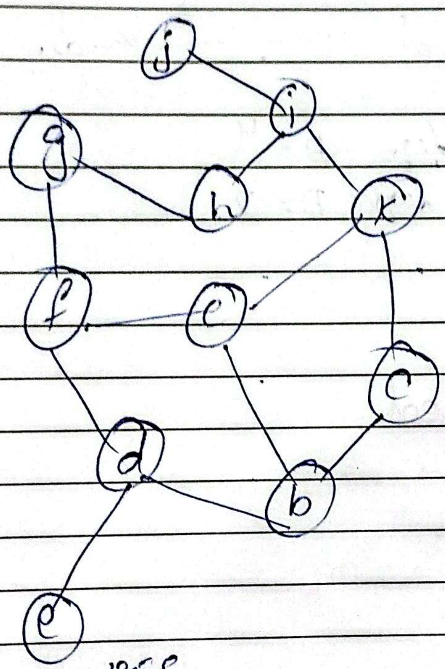


$K(G) = 4$
$\lambda(G) = 4$
min deg $v = 4$
strict
Inequalities $K(G) = \lambda(G) =$ min degree

Date: BFS DFS

Visit every node & explore it

BFS
level wise search



BFS

~~j~~	~~i~~	~~h~~	~~k~~	~~g~~	~~f~~	~~c~~	a	d	b

DFS

stack
LIFO

~~g~~
~~e~~
~~d~~
~~b~~
~~f~~
~~a~~
~~c~~
~~k~~
~~h~~
~~i~~
~~j~~

jikabdfghe

tala