Date

Chapter no. 10

GRAPHS:-

-> Problem to every concievable discipline can be solved using graphs.

A graph $G = (V, E)$ where V is a non-empty set of vertices (or nodes) and E, is a set of edges. Each edge has either one or two vertices associated with it called endpoints.

* An edge is said to connect its endpoints.

a graph with
- infinite vertices
- infinite edges

} is called as an infinite graph.

a graph with

finite vertices
finite edges

} is called a finite graph.

* 10.7 describe the formation of graphs where the edges cross one another.

UNDIRECTED GRAPHS.

SIMPLE GRAPHS.

A graph where each edge connects two vertices and where no two edges connect the same pairs of vertices are called as the simple graph.

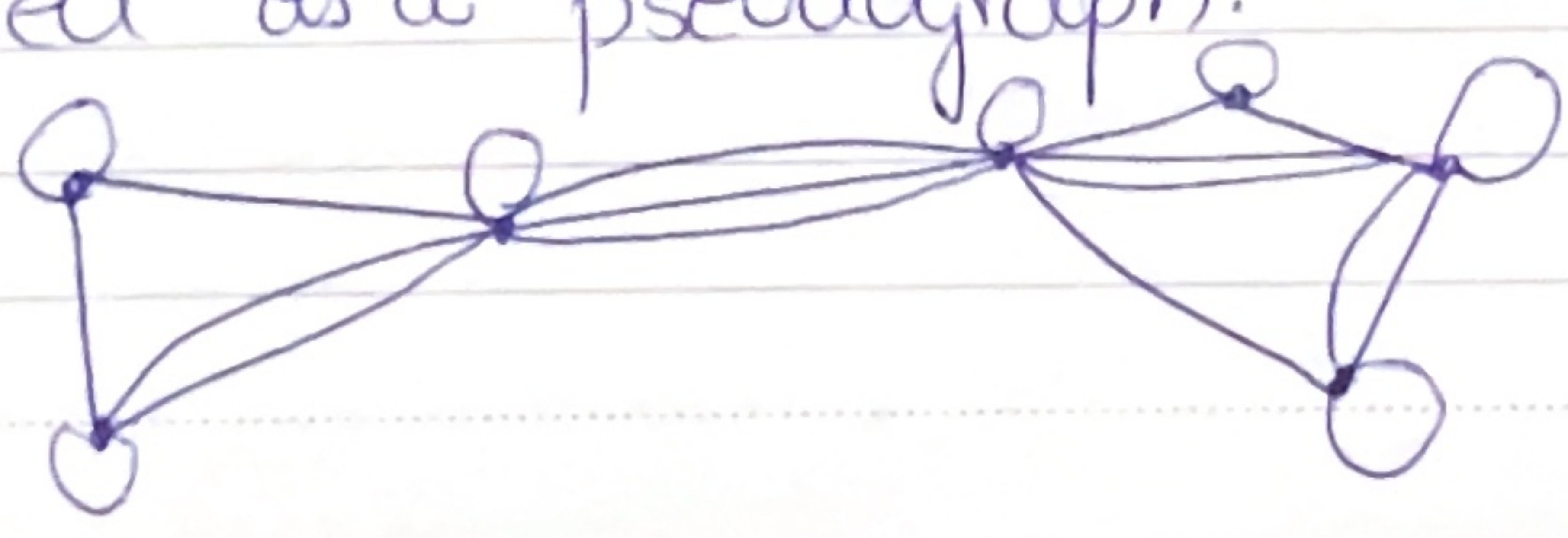
MULTIPLE GRAPHS.

A graph that has multiple edges connecting the same vertiEl are called as multigraphs.

- m different edge are associated to the same unordered pair of vertices.
- we can also say that $\{u, v\}$ is an edge of multiplicity of m.

PESUDO GRAPHS:-

A graph that may include loops and possibly multiple edges connecting the same pair of vertices or a vertex to itself is called as a pseudograph.



links that operate in only one direction.
↳ single duplex links.

DIRECTED GRAPHS :-

A directed graph or digraph (V, E) consists of a nonempty set of vertices V and a set of directed edges (or arcs). Each directed edge is associated with an ordered pair of vertices say (u, v)

$$(\underset{\text{start}}{u}, \underset{\text{end}}{v})$$

* we obtain a directed graph when we assign direction to every edge of an undirected graph.

the way the multiples edges connected the vertices of a graph e.g $\{u, v\}$ can be said to have a multiplicity of m.

m → according to the no. of multiple edge at $\{u, v\}$

⇒ here the direction of the graph is important i.e m denoted multiple $\{u, v\}$ edge with u ⇒ start
v ⇒ end.

Date

Type	edges	Multiple edge allowed	loops
Simple	undirected	NO	NO
Multi	undirected	YES	NO
Pseudo	undireted.	YES	YES.
Simple Directed	Directed	NO	NO
Multi Directed	Directed	YES	YES
MIXED	DIRECTED/ UNDIRECTED	YES	YES.

SOCIAL NETWORKS.

(i) ACQUINTANCESHIP GRAPH.

Graph that indicates whether two people have acquintance will one another.

Also known as the friendship graph.

Features

(i) connect two people an undirected edge. when know to eachother.

(ii) No multiple edges.

(iii) no loop (if the notion of self knowledge is absent).

(ii) INFLUENCE GRAPH.

Features.

(i) Directed graph.

(ii) No multiple edge.

(iii) No loops.

(iii) COLLABORATION GRAPHS.

(i) Simple graphs (undirected)

no multiple edges

no loops.

Date

COMMUNICATION NETWORKS.

Vertices = devices.
edges = communication links.

call graphs :-

(a) (Directed multigraphs).

telephone number = vertex.
telephone call = directed edge

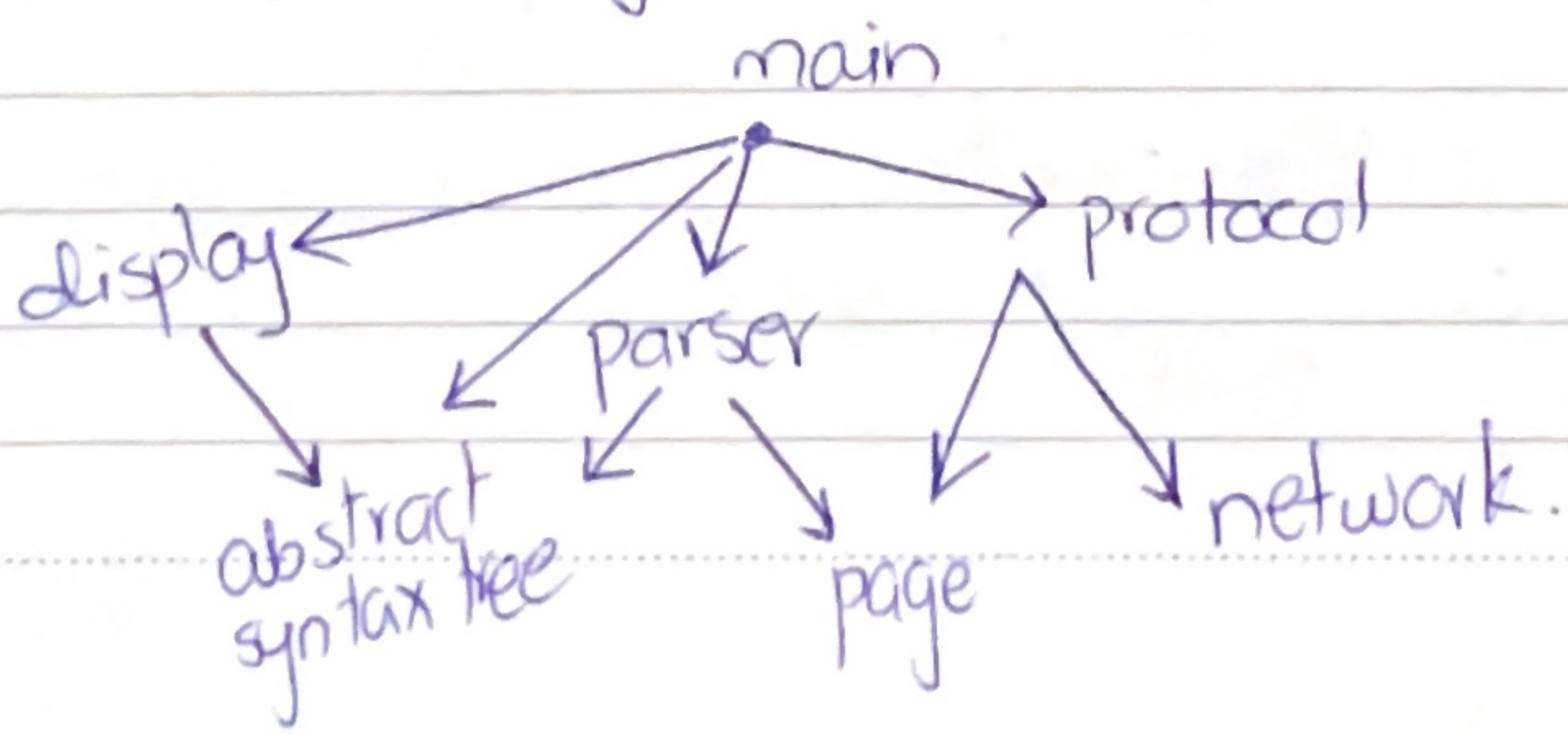
(b) when we donot consider who called whom so we use an undirected graph.

SOFTWARE DESIGN APPLICATIONS:-

MODULE DEPENDANCY GRAPH:-

helps Structure a program into different modules; each module is represented by a vertex. There is a directed edge from a module to a second module if the second module depends on the first. *

A program dependancy of a web browser is given as.



(important

Date

Revision

PRECEDANCE GRAPH.

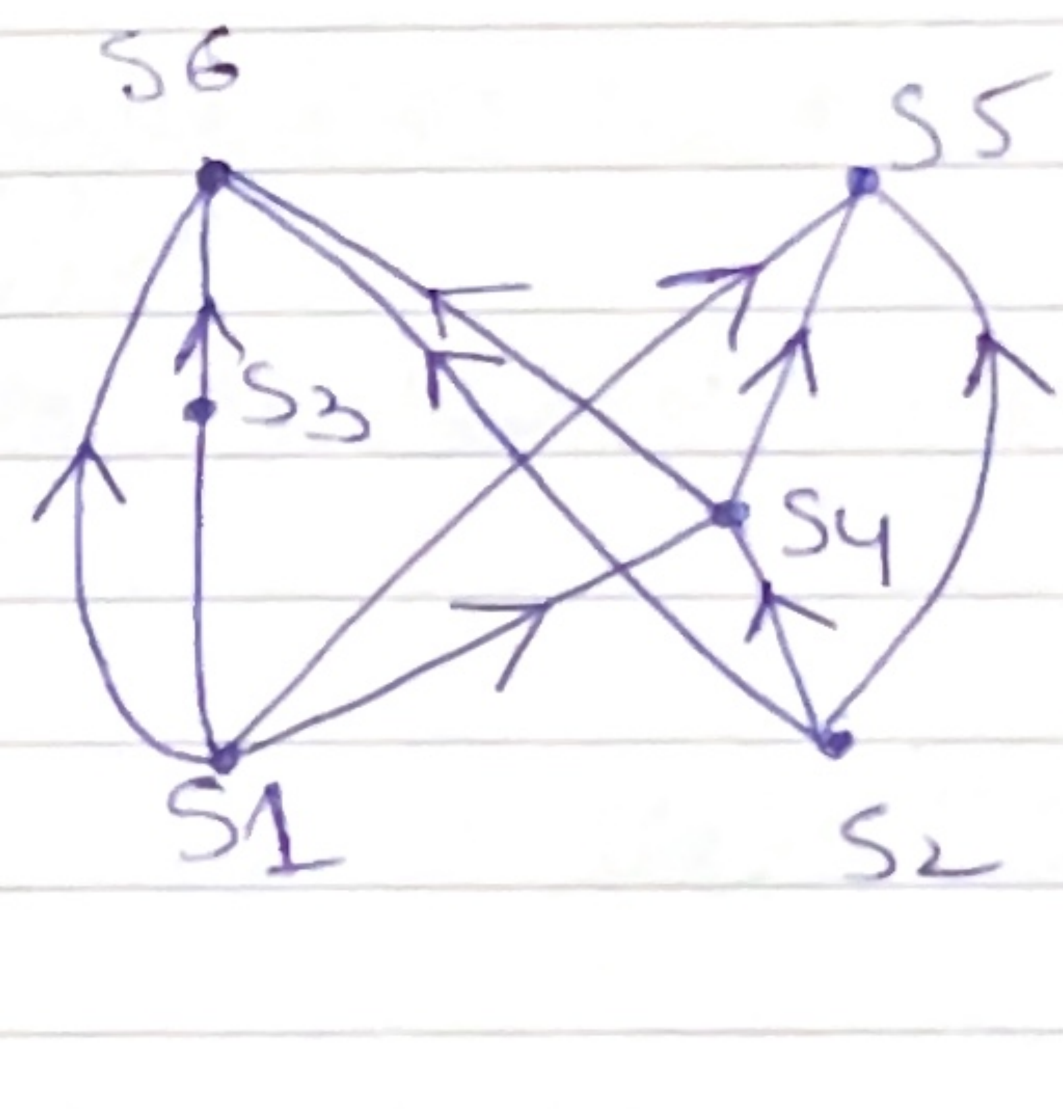
Computers work efficiently when they process statements concurrently however it is important that no statement is executed that depends upon the result of a previous execution of a certain other statement.

These dependance b/w graphs can be potrayed by directed graphs. Each statement is represented by a vertex and there is an edge from one statement to a second statement if the 2nd statement depends upon the first statement.*

This type of dependance analysis graph is called as precedancegraph.

$S1 \Rightarrow a := 0$
$S2 \Rightarrow b := 1$
$S3 \Rightarrow c := a+1$
$S4 \Rightarrow d := b+a$
$S5 : \Rightarrow e := d+1$
$S6 : \Rightarrow e := c+d$

S5 cannot be executed before S1, S2 and S4 are executed.



Date

NLU (natural language processing)

SEMANTIC NETWORKS:-

Vertices = works.
Undirected edge = connects two works with a semantic relationship.

A semantic relationship b/w two or more words is based on the meaning of the word.

10.2 :- "GRAPH TERMINOLOGIES"

Adjancey and incident. (UNDIRECTED GRAPH)

- Two vertices connected with eachother through an edge are adjacent to one another,
- Such edge b/w two adjacent (matrix^x) vertice are known as incident with the vertices.

Neighbourhood.

ones that make an edge with a vertex.

The set of all neighbours of a vertex v of $G = (V, E)$ is denoted as $N(v)$ and called as the neighbourhood of vertex v.

e.g

$$N(A) = \bigcup_{v \in A} N(v)$$

Neighbarhood of A

Date UNDIRECTED

* a loop in a vertex will thus include that vertex in its neighbours set too.

Degree of a Vertex.

(i) in an undirected graph, the no. of edges incident with it
(and)
(ii) a loop at a vertix (contributes twice to the degree)

forms the degree of a vertex 'v' denoted as. deg(v).

* a vertex of zero degree = isolated.
* an isolated vertex is thus not adjacent to any vertex.

* a vertex with degree 1 = pendant.
* a pendant vertex is thus adjacent to only one vertex,

THE HANDSHAKING THEOREM :-

The no. of edges of an undirected graph are related to the sum of degrees of the graph's vertices as.

$$2m = \sum_{v \in V} \deg(v)$$

where m is no. of edges.
v is the vertex from all the set of Vertice of graph (V)

Note that this applies even if multiple edges and loops are present.

Date

* The handshaking theorem thus also suggests that sum of the degrees of vertex(s) of an undirected graph is even.

Theorem no. 2

An undirected graph has even no. of vertices of odd degrees.

DIRECTED GRAPH :-

Adjancey:-

When (u,v) is an edge in a graph with directed edges, u is said to be adjacent to 'v' and 'v' is said to be adjacent to 'u'

The vertex 'u' is initally vertex and 'v' is the terminal / end vertex.

MCQ
* The initial and terminal vertex of a loop are same.

Degree of a vertex.

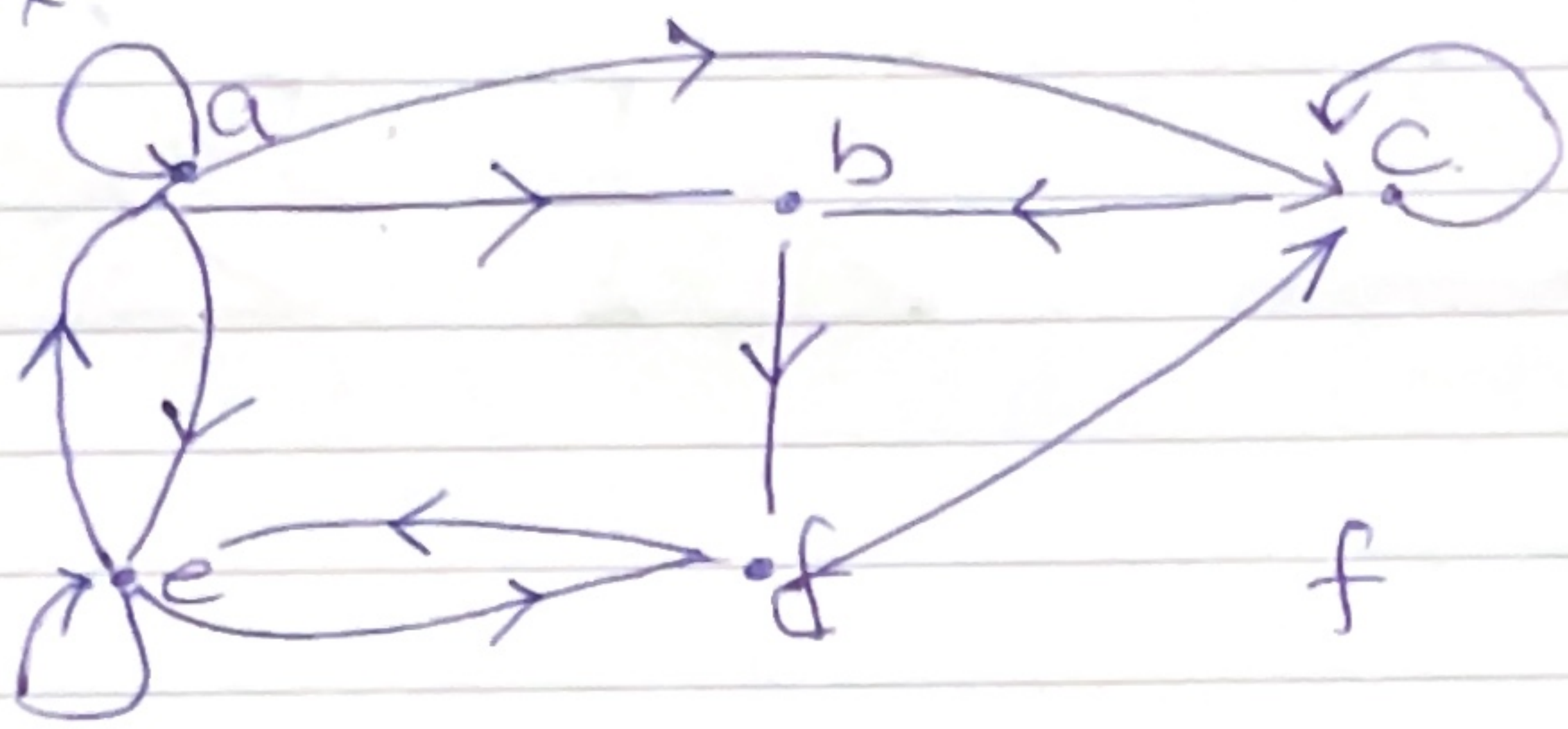
(i) In a graph with directed edges the indegree of a vertex 'v' is denoted as $deg^{-}(v)$ and it is the no. of edges with v as the terminal vertex.

Date

The out degree of a vertex v is denoted as $\deg^+(v)$ and it is the no. of edges with v as their initial vertex.

* Loop at a vertex in a directed graph contributes 1 in both in-degree and the out degree of that vertex.

Example
-x-



in-degrees

a: 2
b: 2
c: 3
d: 2
e: 3
f: 0 (isolated)

out degree.

a: 4
b: 1
c: 2
d: 2
e: 3
f: 0
Td: 2.7

Theorem no. 3 :-
(Handshaking theorem for directed graphs).

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

Date

10.2.3 Special Cases of Graphs.

Complete Graph.

A complete graph on 'n' vertices denoted by K_n is a simple graph that contains atleast one edge between each pair of vertices.

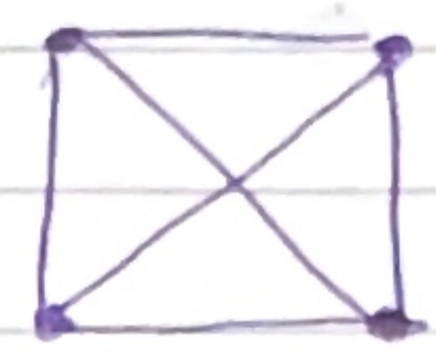
let $n = 1, 2, 3, 4, 5, 6$.

K_1

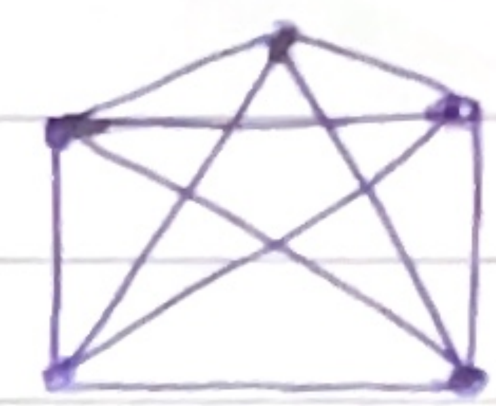
K_2

K_3

K_4



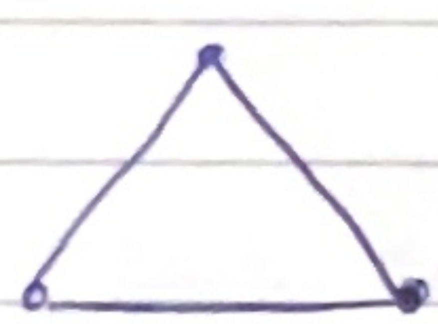
K_5



K_6

Cycle graphs.

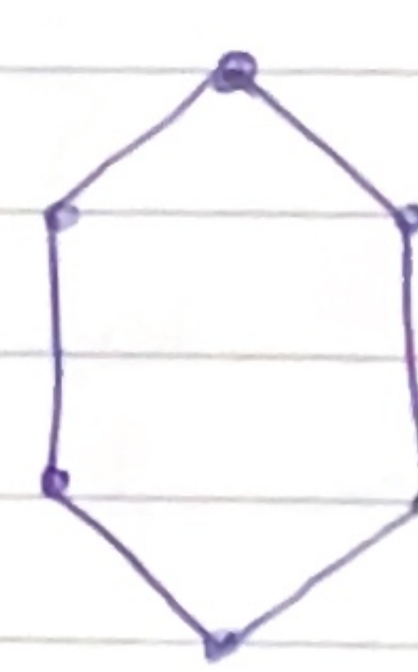
C_3



,

C_4

C_5



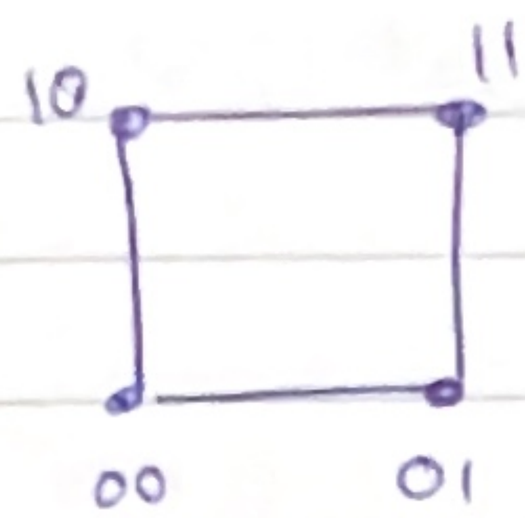
and so on....

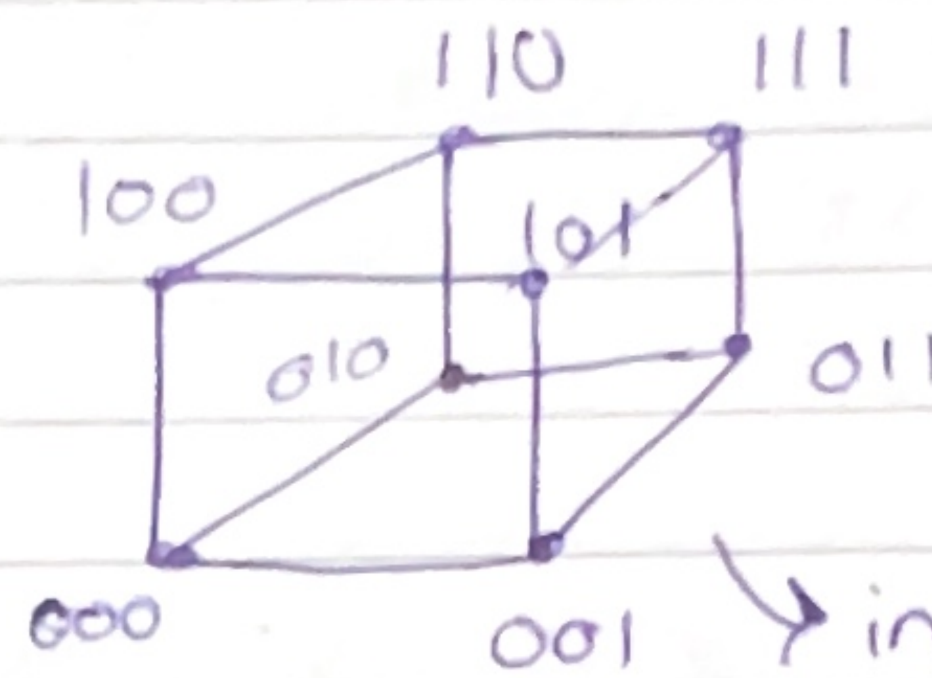
Cube graphs:-

also known as n-dimensional hypercube, denoted as Q_n is a graph that has vertices represented as 2^n bit strings of length n.
Two vertice in such graphs are only adjacent if they bit string difference of only one bit position.

Revision -12.

Date





→ important to draw.

10.2.4

Bipartite Graph.

A bipartite graph is called or formed from a simple graph if the vertex of that simple graph can be partioned into two disjoint sets i.e 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 connect either two vertices in V_1 or two vertices in V_2.)

at this condition we call the pair V_1 & V_2 the bipartion of V.

Theorem 4 (helps to check if a graph is Bipartite).

"A simple graph is Bipartite if and only if it is possible to assign one of the two different colors to each vertex of the graph so that no two adjacent vertzices can be assigned the same color".

Revision example 12

—x—

Isomorphic graphs!

* sometimes the graphs (2) have exactly the same form, in the sense that there is one to one correspondance b/w the vertex sets that preserve edge.

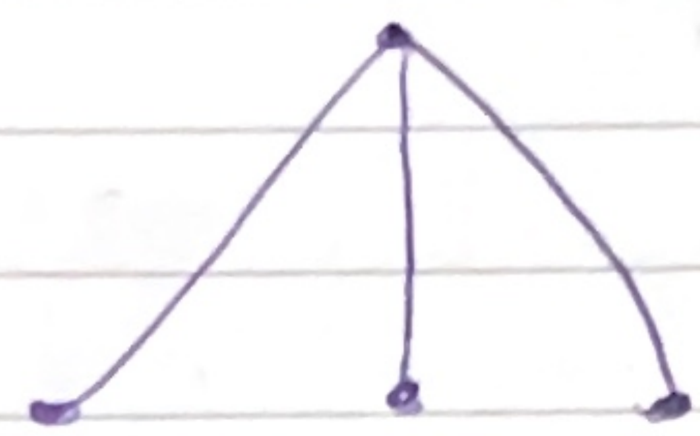
Date

COMPLETE BIPARTITE GRAPH:-

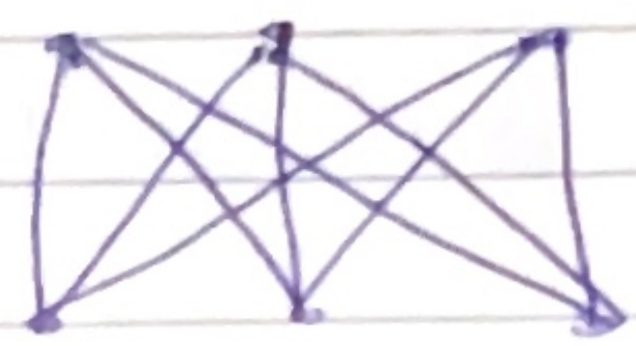
A complete Bipartite graph is denoted as $K_{m,n}$.
It is a graph that has its vertex set partitioned into two subsets say m and n vertices; with an
* edge b/w two vertices if and only if one vertex is in one subset and the other is in another subset.

$K_{2,3}$

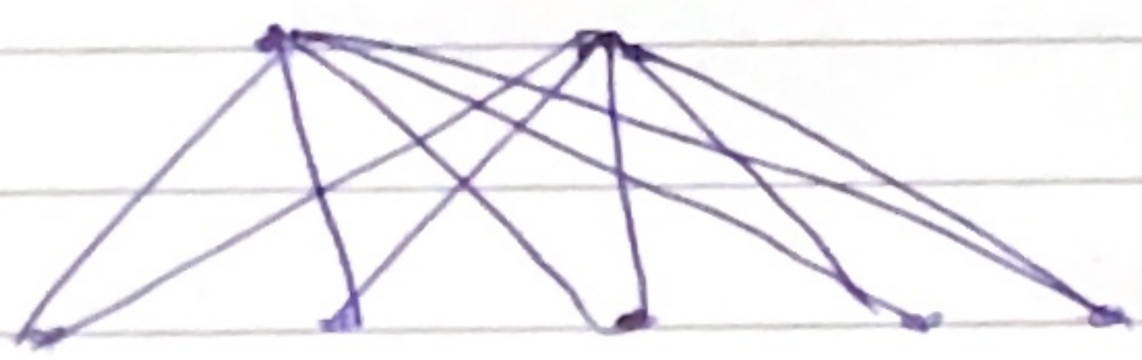
$K_{1,3}$



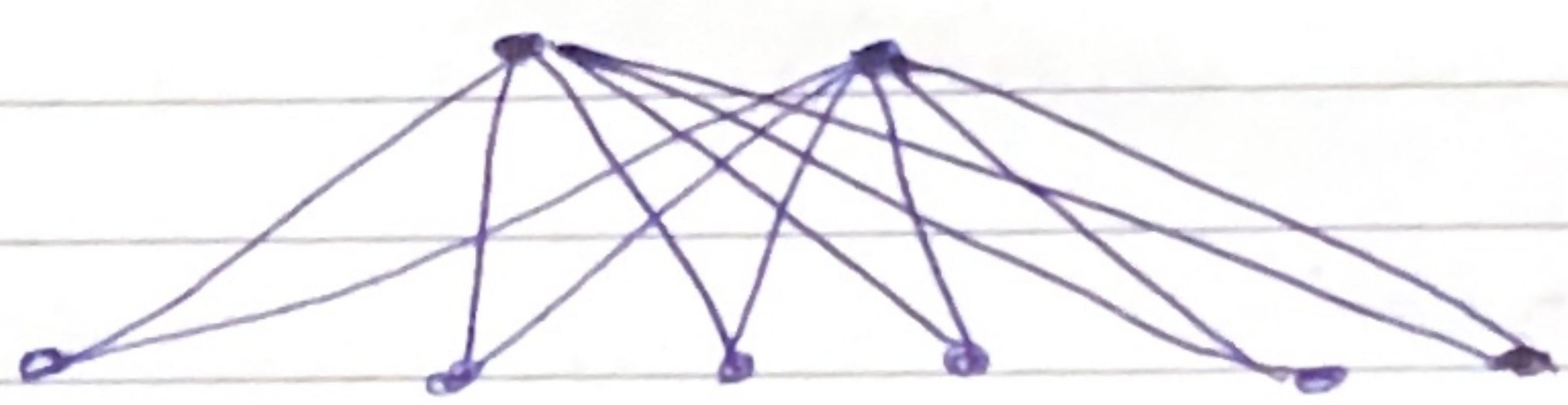
$K_{3,3}$



$K_{3,5}$



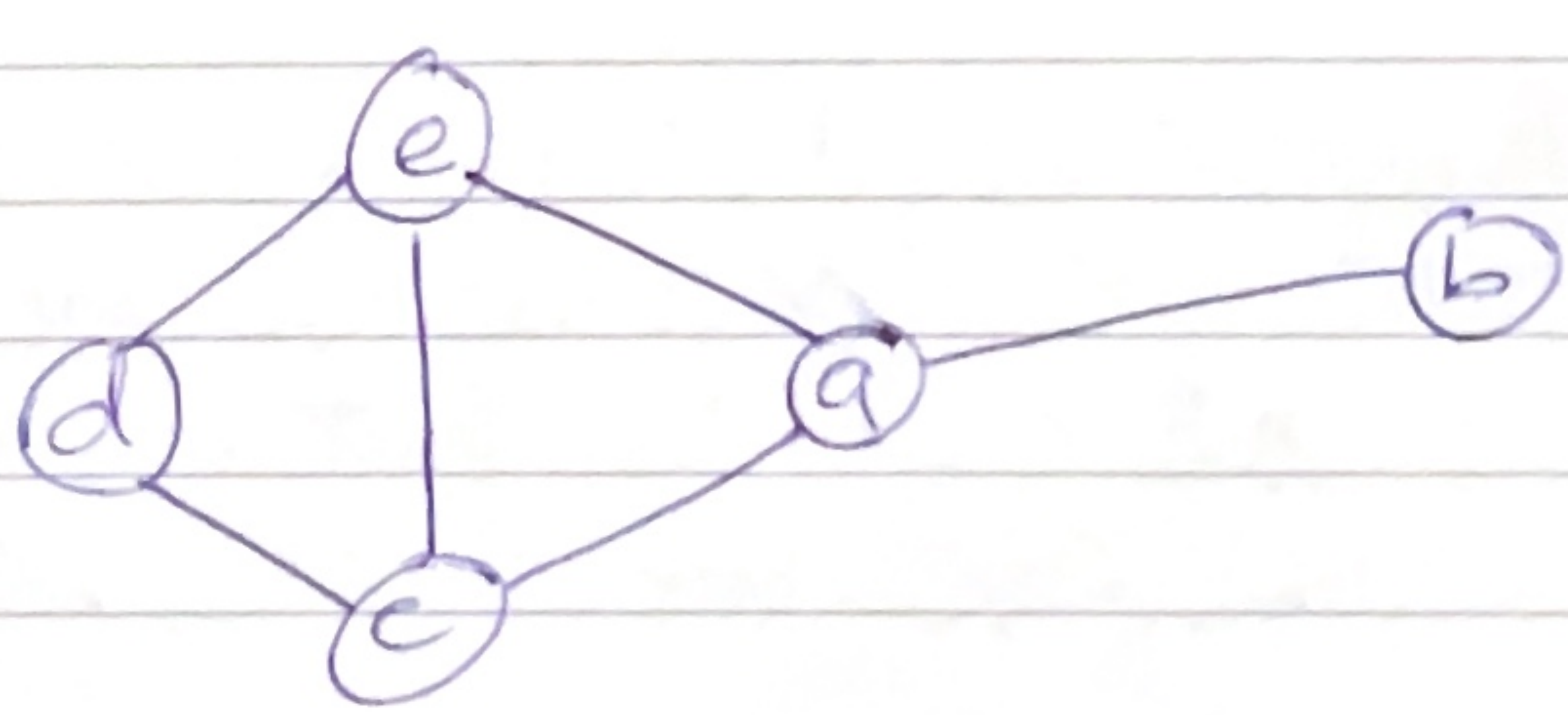
$K_{2,6}$



SECTION
10.3

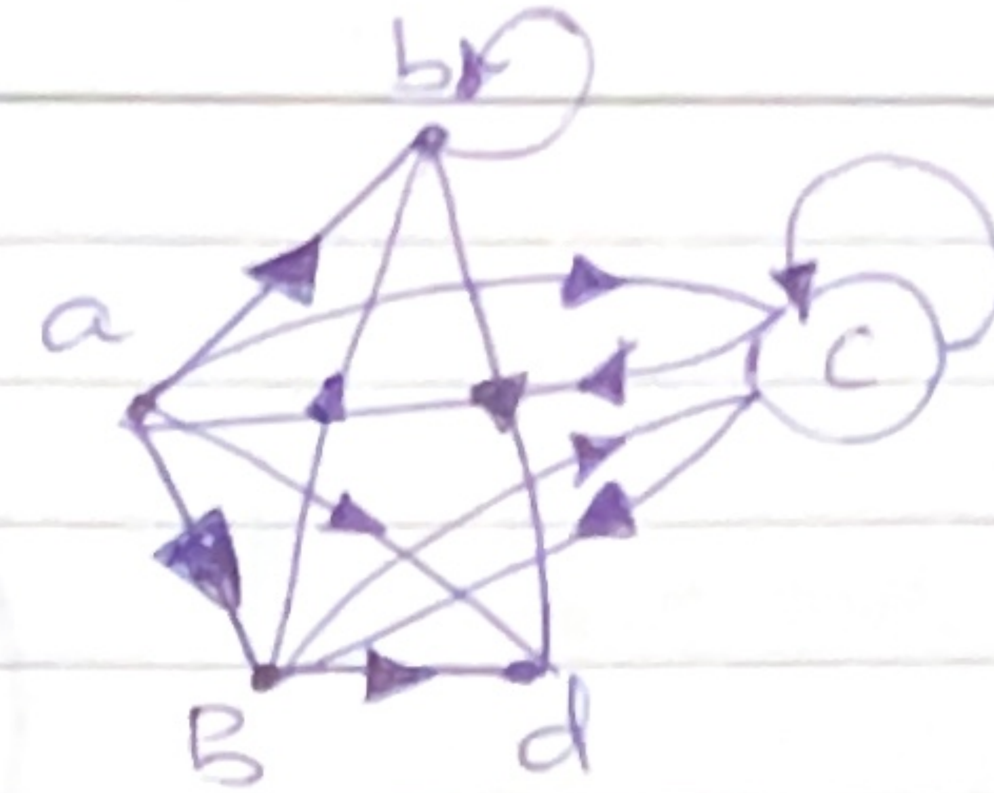
"Adjancy List"

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



Date

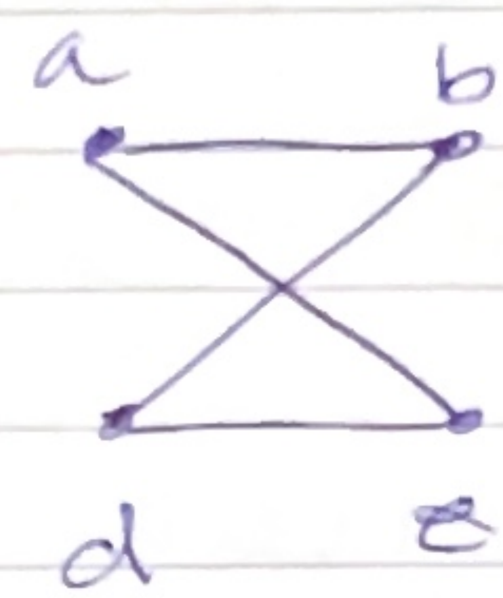
a	b, c, d, e
b	d, b
c	c, a, e,
d	-/-
e	b, c, d.



'Adjancey Matrix'

We can represent a graph on n vertices using an nxn matrix A.

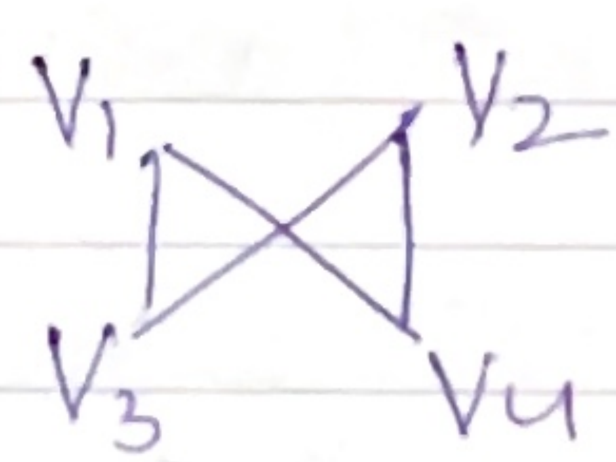
A_{ij} is 1 if there is an edge and 0 is otherwise



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

"Graph Isomorphism"

Sometimes we need to find whether the two differently represented graphs are really the same?



⇒ these two graphs have the same no. of vertices and edges and adjancey hence isomorphs.

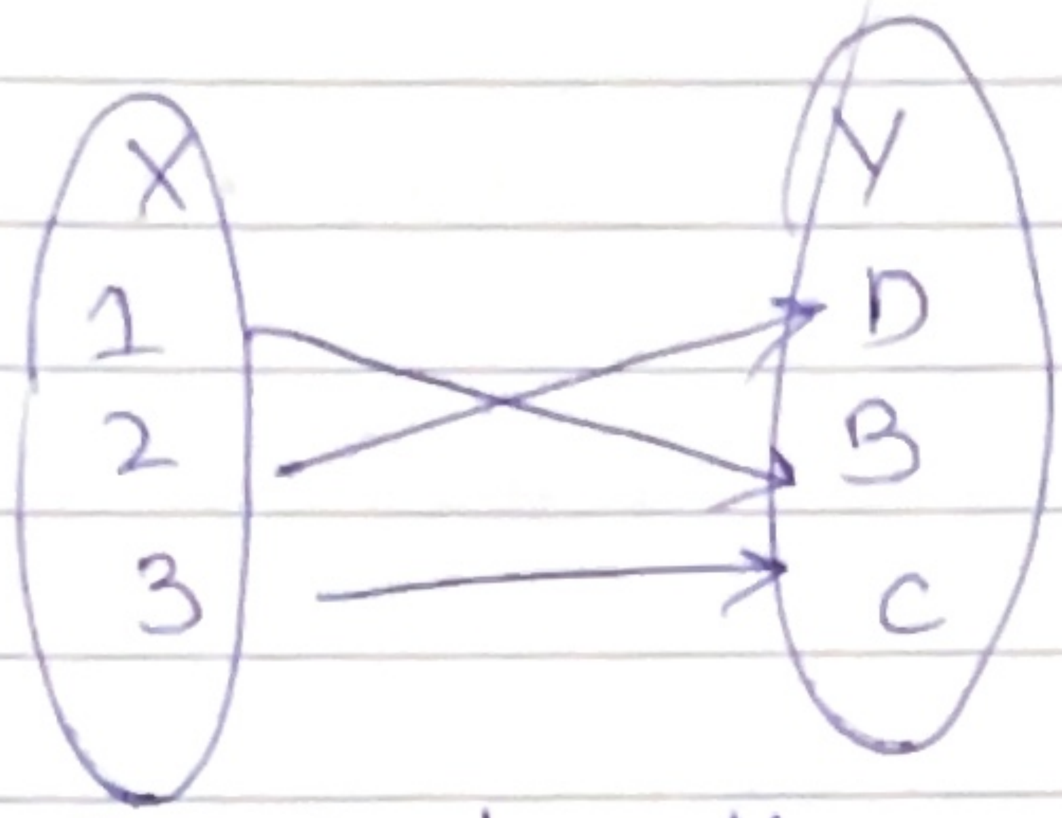
Important. (iv) no. of ones in adj matrix should be equal.

Date

ONE TO ONE FUNCTION:-

(non-surjective).

also known as the injective function. It is a function that maps distinct elements of its domain to distinct elements of its co-domain/ range



Symbolically

$\forall a, b \in X \; f(a) = f(b)$

$\Rightarrow a = b.$

which is logically equivalent to the contrapositive

$\forall a, b \in X, \; a \neq b \Rightarrow f(a) \neq f(b).$

Example.

Pseudo graph

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

ONTO FUNCTION

SURJECTIVE FUNCTION.

A function 'f' from the set X to a set Y is surjective if every element y in the codomain Y of f, there is atleast one element x in the domain X of f such that $f(x) = y$.

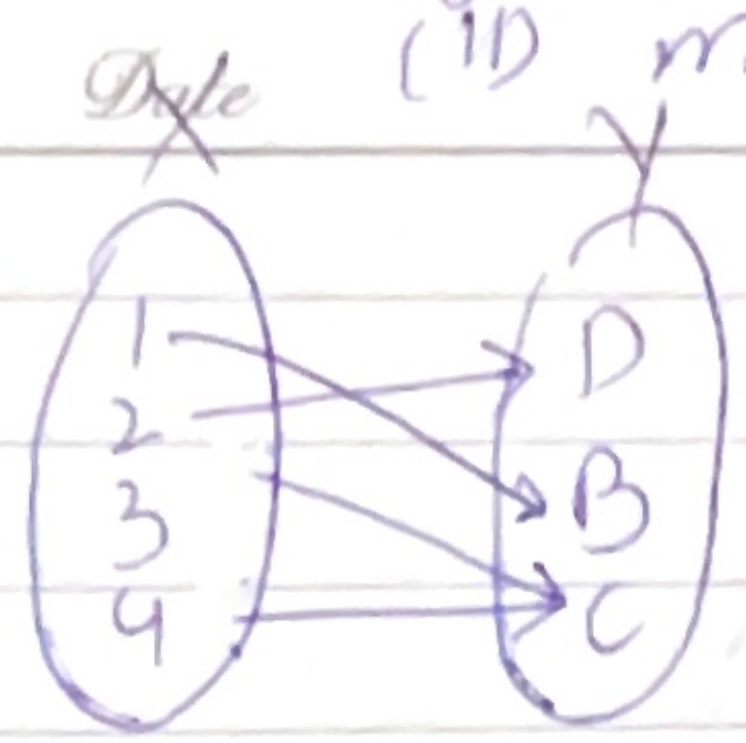
→ to prove isomorphism

(i) check invariants.

(ii) map functions

(iii) draw adjancey matrix. [should be equal for both graphs]!

checking degree of one vertex and also important the adjacent vertex.



Symbollically

if $f: X \to Y$ then f is said to be surjective if $\forall y \in Y, \exists x \in X, f(x) = y$.

"**The simple graphs** $G_1 = (V_1 E_1)$ and $G_2 = (V_2 E_2)$ are isomorphic if there exists a one to one and onto function f from V_1 to V_2 with the property that a and b are adjacent in G_1 if and only if $f(a)$ and $f(b)$ are adjacent in G_2, for all a and b in V_2. Such a function is called as isomorphism.

* two simples graphs are are isomorphic, if there is a one-to-one correspondance b/w vertices of two graphs that preserve the adjancey relationship.

* Isomorphism in graphs is an equivalence relationship.

— Graph invariants —

* a graph invariant is a property that must be satisfied by the isomorphic graphs.

↳ isomorphic graphs will have the same no. of vertices.

↳ " " will have the same vertex degrees.

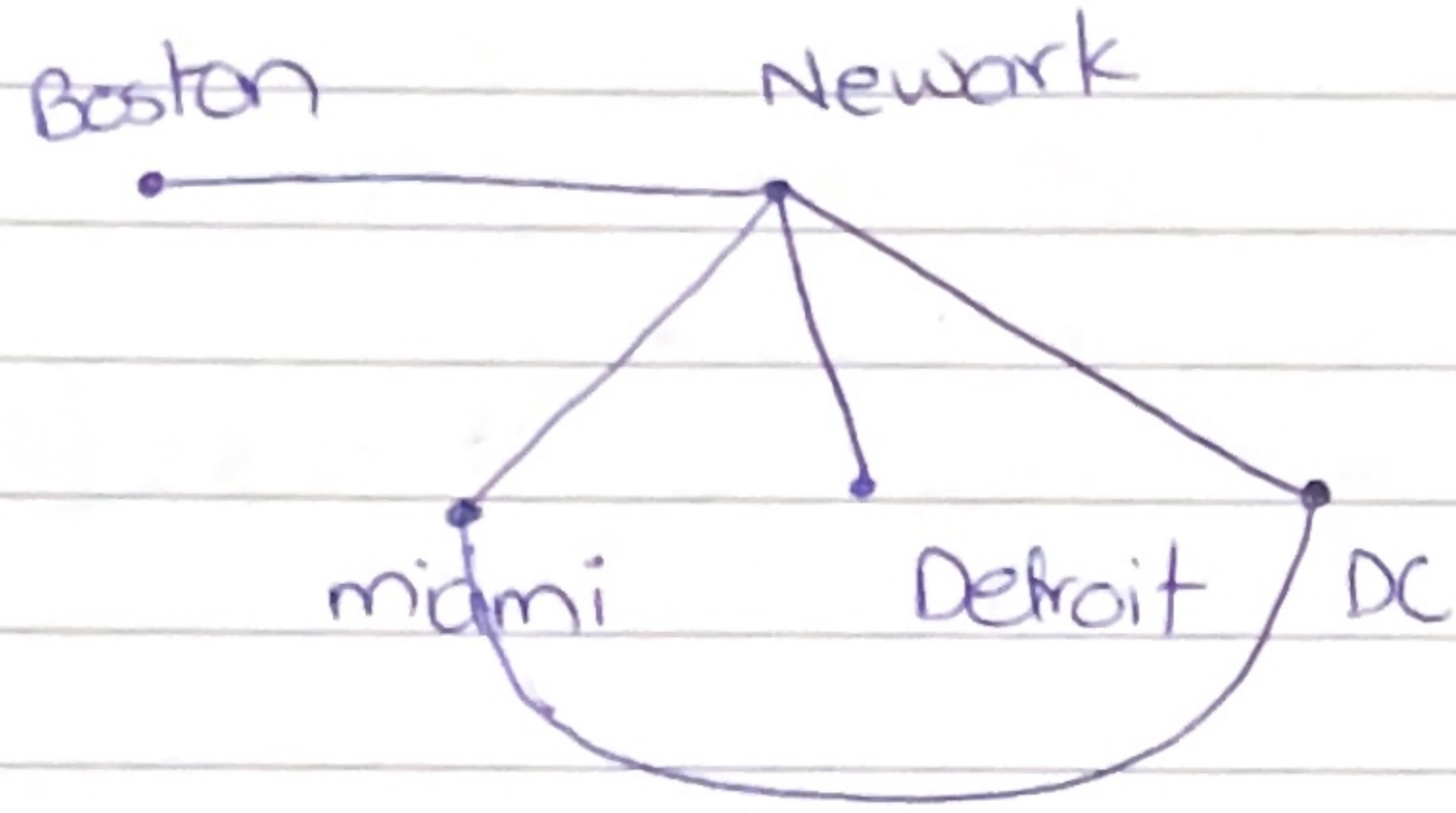
1, 3-9, 15, 16, 18, 19, 20, 25, 26*, 27*
28, 29, 30, 31*, 32, 33, 34*, 35*, 37, 38.

Date

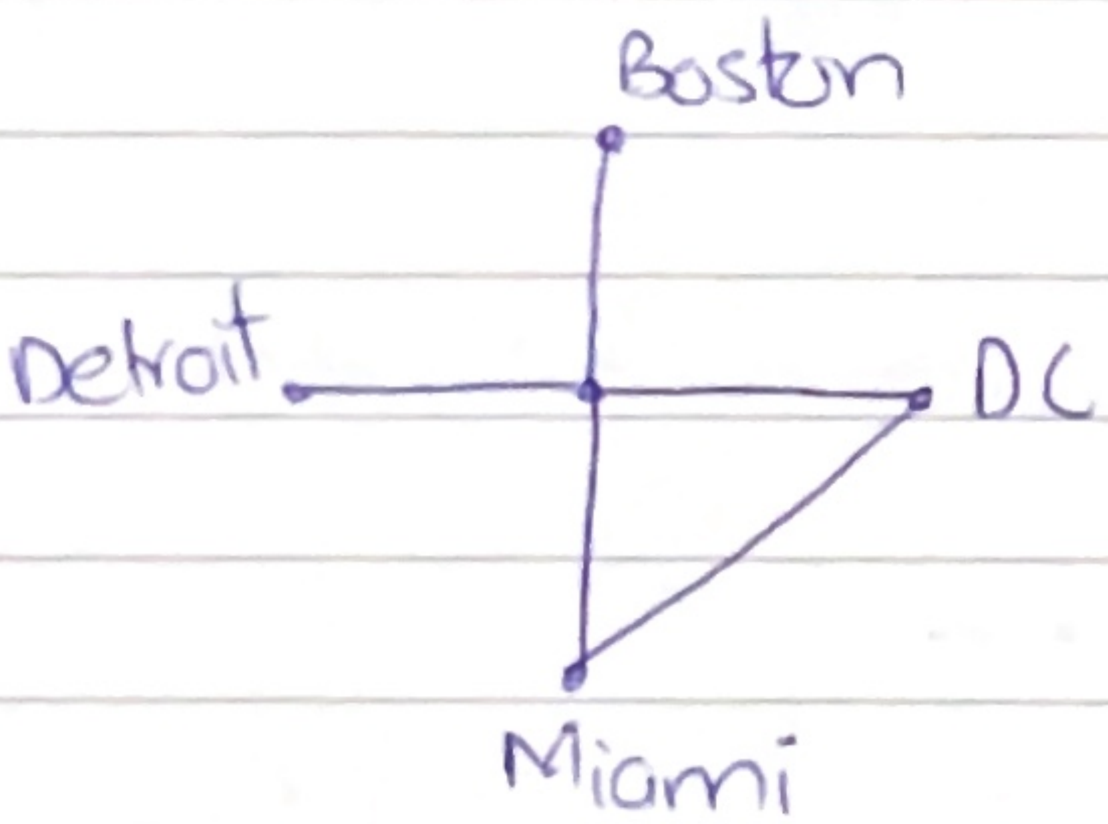
EXERCISE 10.1.

Question no 1

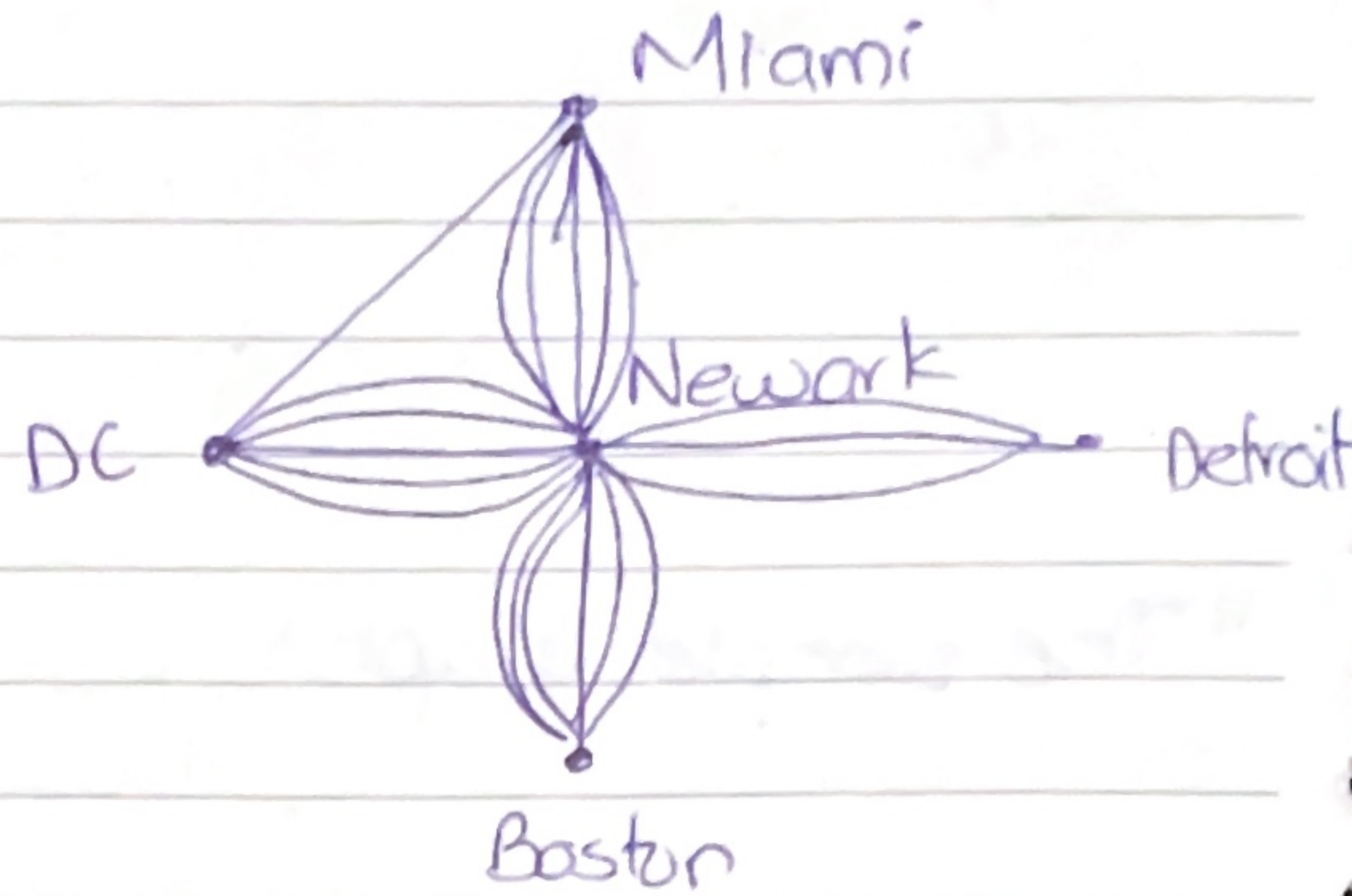
(a) one edge can represent flights in either direction.



or

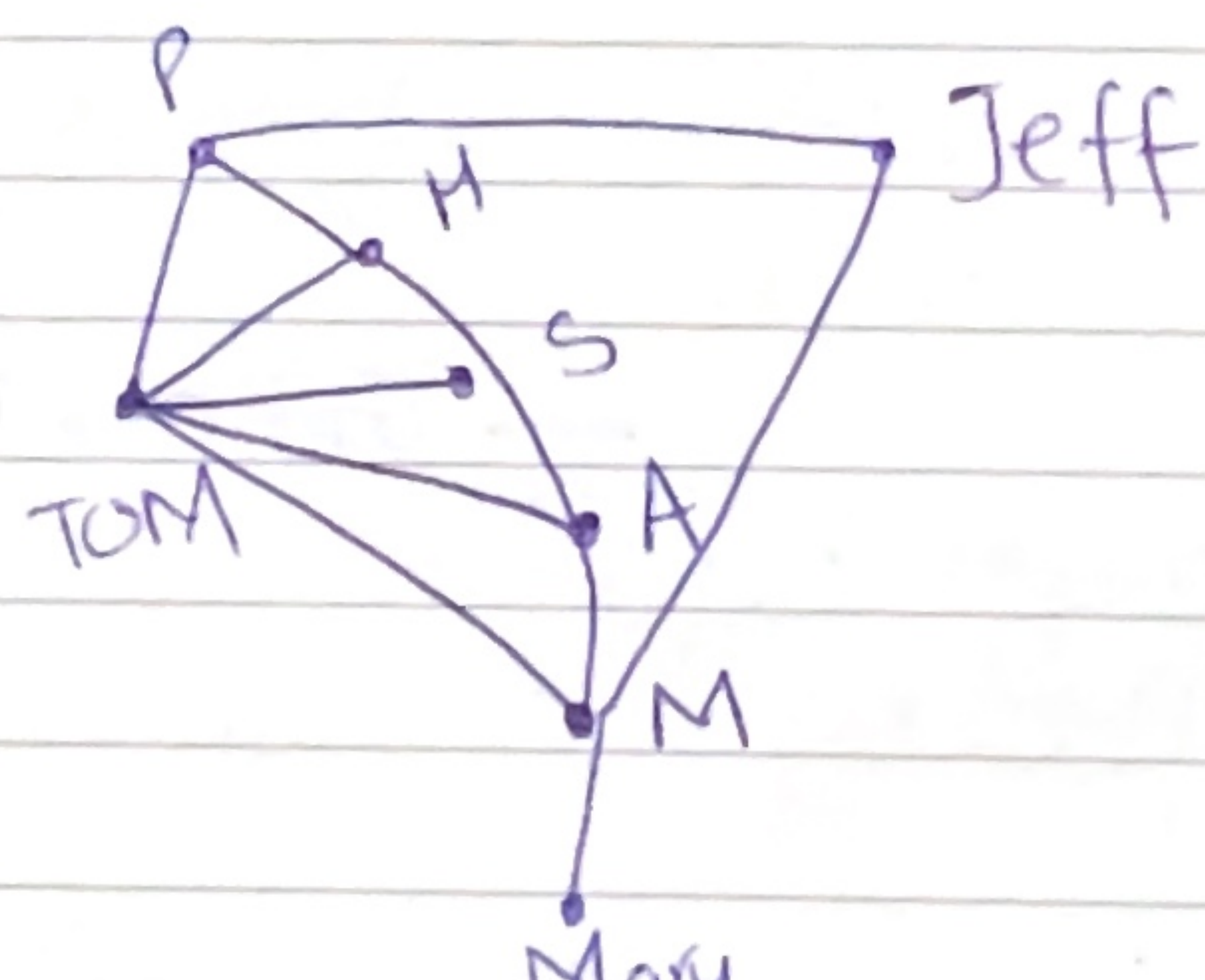


(b) now one edge represents one flight in both directions



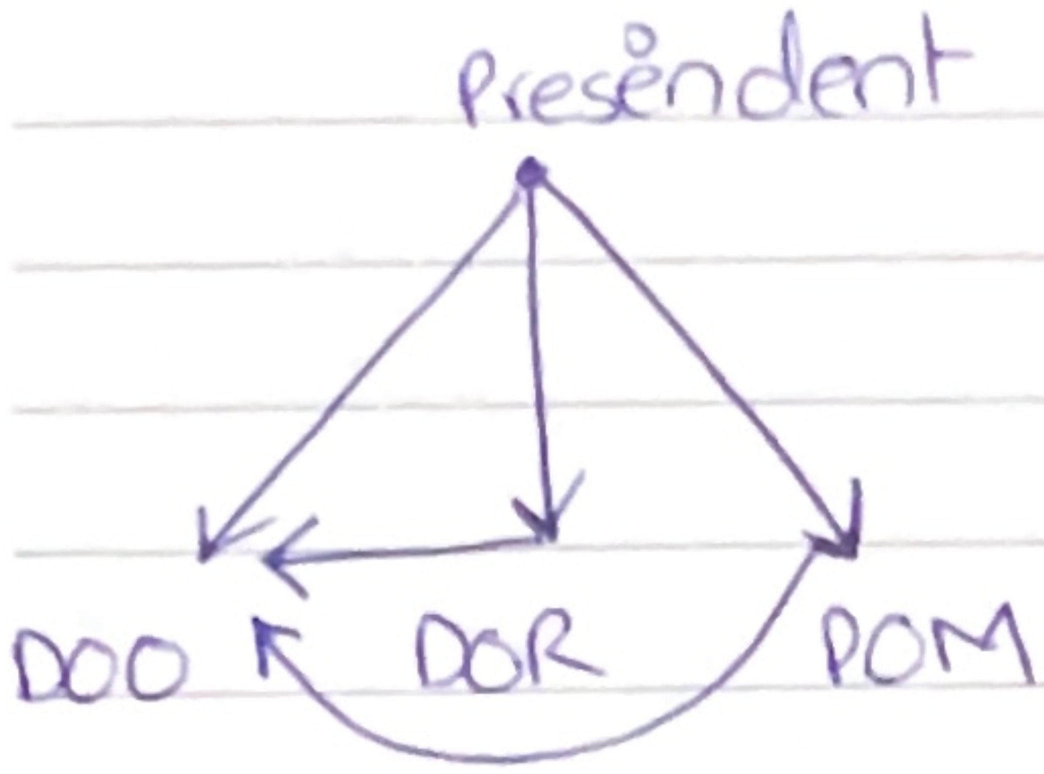
(c) easy
(d) easy
(e) easy.

Question no. 16



Date

Question no.19



Question no.20

Cheif Financial
• officer

Question no.27

One way to tackle this problem would be to assign a weight to each edge representing the particular email send it.
(address used to)
A new weight on an edge originating from a computer vertex would indicate a new email address being used to send email messages.

Question no.29:-

V = vertices = set of all people in the party.

E = edges = { (u,v) ∈ V x V | u knows v }

The edges should be directed indicating that a person knows a certain person and its not usually always the other way around e.g fans of a celebrity.

Multiple edges arent allowed because a person either knows the other person or does not.