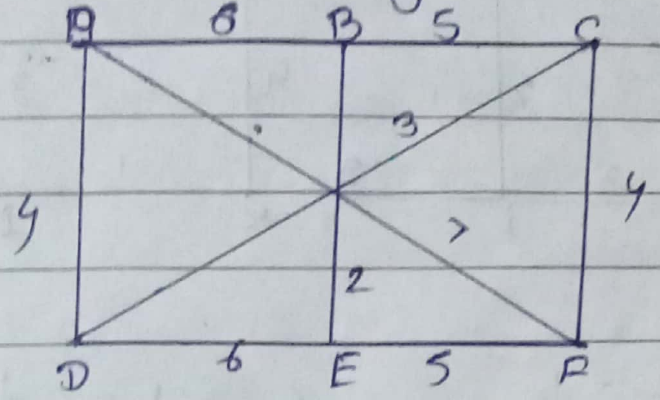
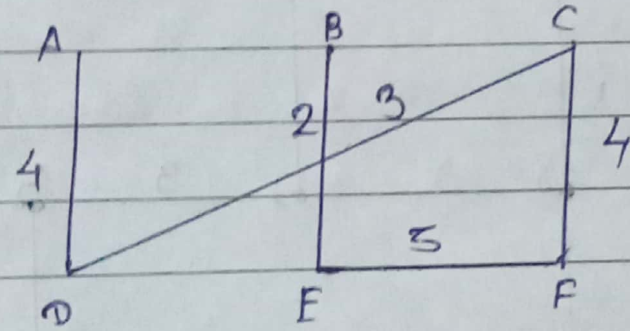
Unit - 5

Assignment No. 2

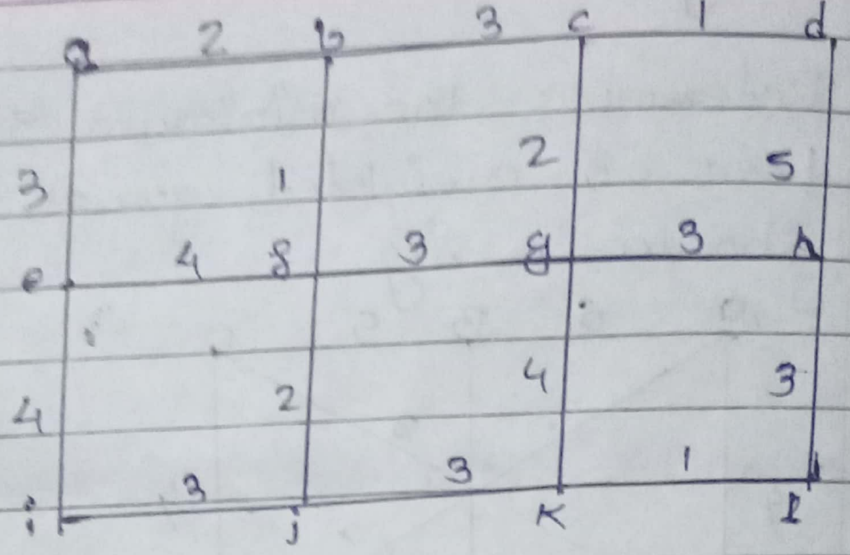
Q1) Determine the minimum spanning tree of weighted graph shown in fig.



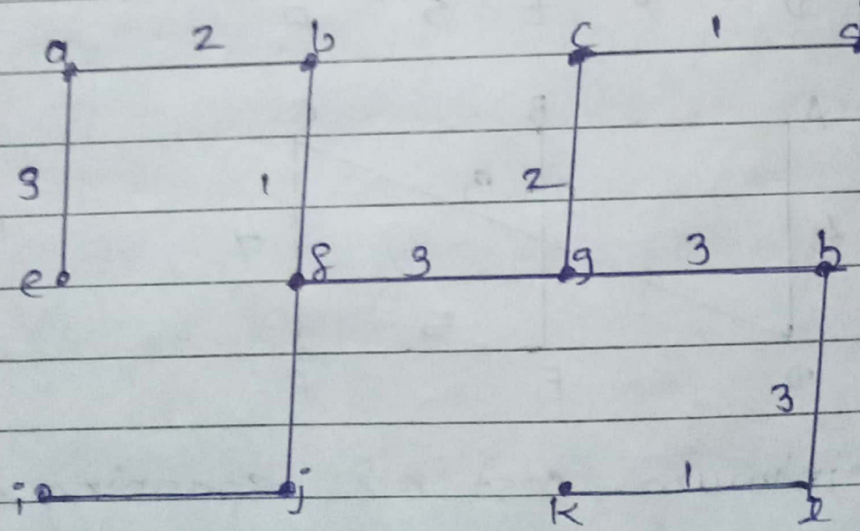


minimum cost of spanning tree is - 18

Q2) Determine the minimum spanning tree of the weighted graph shown in fig.



→



minimum cost of spanning tree is - 24.

Q3) what are the properties and application of trees.

→ A connect graph that consist no simple circuits is called as a tree these are usefull in computer science to construct useful algorithm a tree is undirected graph and it is helpful to determine some stratagies.

Properties of Trees

1) An undirected graph is a tree if an if there is a unique simple path between any 2 of its vertices.

2) A rooted tree is a tree in which one vertex has been designeted as the root and every edge is directed away from the root.

3) A ~~roo~~ tree with n vertices has n-1 edges

4) A graph is a tree if and only if it minimally and maximally connected

5) Graph is connected, undirected, asyclic.

Application of Tree

1) Tree are use to implement the file system of several popular operating system.
2) You can buit disicion tree
3) It can store large value of da
4) Trees are use in compiler design
5) It is use for huffman coding which is use for data compression technique.

Q4) Explain Prefix, Postfix and Infix notation

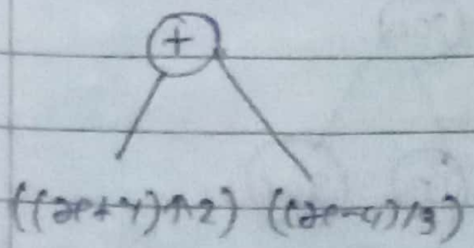
→ Infix- It is associated with inorder traversal. here operator is placed between ~~inorder~~ operand such a type of standard expression is infix notation which is ~~also called~~.

Prefix - It is associated with Preorder traversal here operator is placed before operand. which is also called as polish notation.

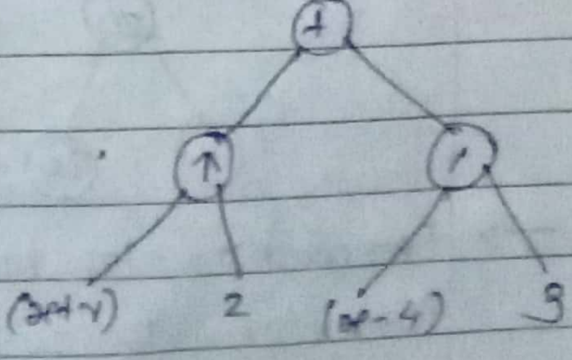
Postfix - It is associated with Postorder traversal here operator is placed after operand which is also called as reverse polish notation.

Q5) What is ordered rooted tree that represents the expression $((x+y)\uparrow 2)+((x-4)/3)$? And write these expression in prefix and postfix notation of the given tree.

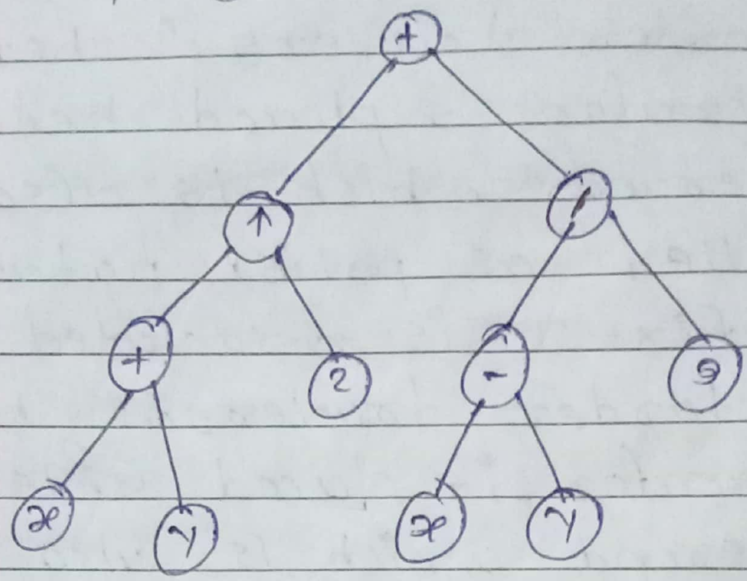
→ step ① -



step-②

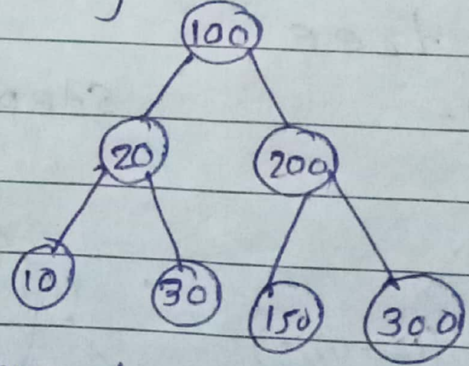


Step - ③



Prefix expression - +↑+xy2/-x43
Postfix expression - xy+2↑x4-3/+

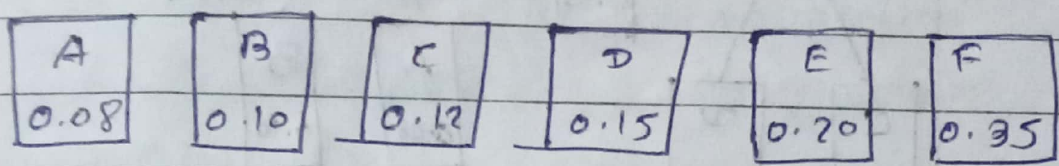
Q6) Find out preorder, postorder and inorder traversal of following tree.



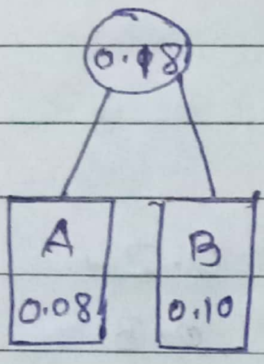
→ Preorder - 100, 20, 10, 30, 200, 150, 300
Postorder - 10, 20, 30, 100, 150, 200, 300
inorder - 10, 30, 20, 150, 300, 200, 100

Q7) Describe Huffman coding and use huffman coding to encode the following symbols with the frequencies listed: A: 0.08, B: 0.10, C: 0.12, D: 0.15, E: 0.20, F: 0.35
What is the average number of bits used to encode a character?

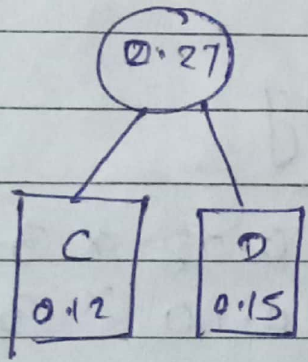
→



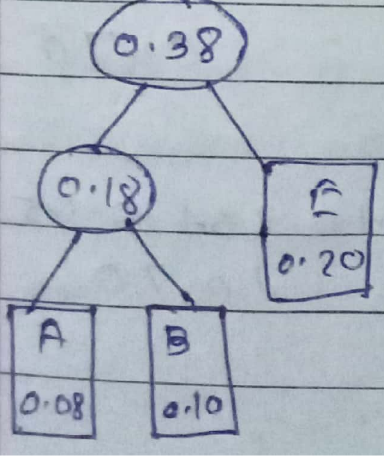
step ①



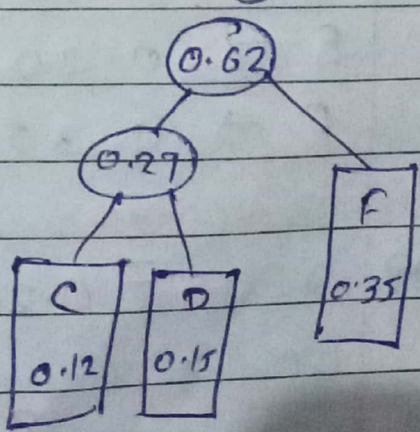
step ②



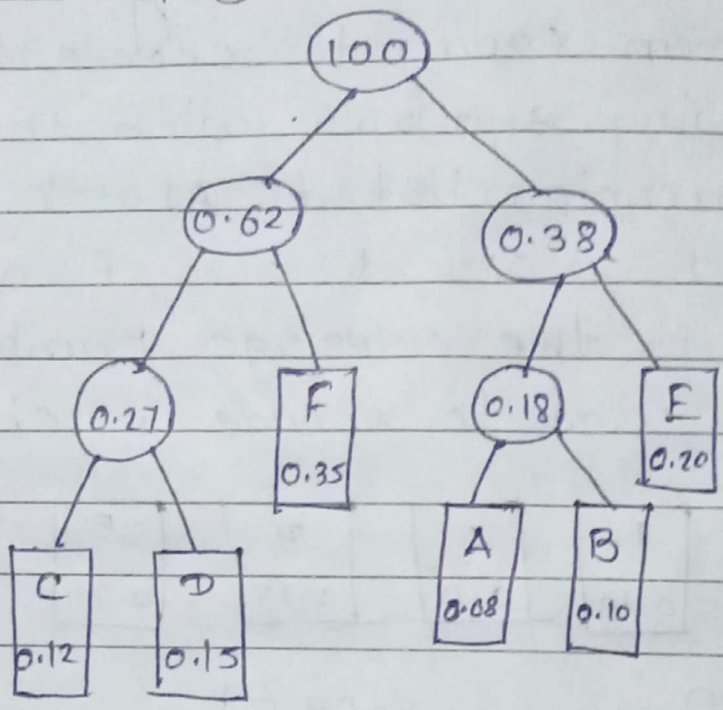
step ③



step ④



step ⑤ ~~te~~ –



coding

A → 0.08 × 3 = 0.24
B → 0.10 × 3 = 0.3
C → 0.12 × 3 = 0.36
D → 0.15 × 3 = 0.45
E → 0.20 × 2 = 0.40
F → 0.35 × 2 = 0.70

= 0.24 + 0.3 + 0.36 + 0.45 + 0.40 + 0.70

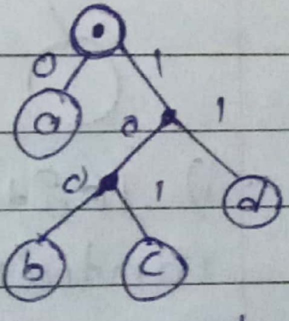
= 2.45

Q8) What are Prefix codes?

→ Prefix Code is a ~~par~~ variable length code in which no code word is prefix of another one. A Prefix code is a type of code system distinguished by it's possesion of the "Prefix Property". It is trivially true for fixed length code, so only a point of consideration in variable length code.

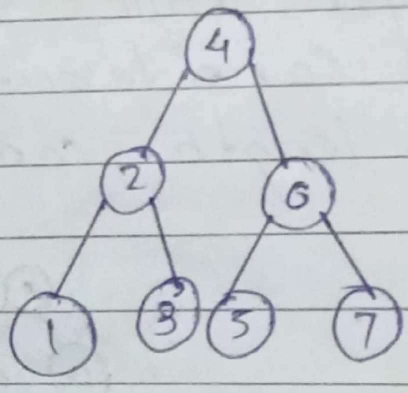
Ex:-

a = 0
b = 100
c = 101
d = 11



Consider the prefix code
0, 10, 110, 1111
The length of the codewords are 1, 2, 3, 4. However the length of last codeword can be reduced from 4 to 3 as (0, 10, 110, 111) in reduce from - 1, 2, 3, 3.

Q 9) Describe binary search tree with suitable example
→ ~~Consider~~ A Binary tree is a rooted tree that has atmost two children and it is ordered tree every node is having left & right reference.
lets consider following example
4, 2, 3, 6, 5, 7, 1



1) left subtree of a node contain only with a key lesser than the node key
2) The right subtree of a node contain only node with keys greater than the node key
3) The left and right subtrees each must also be a binary search tree.

Q10) Determine the order in which a Preorder traversal visits the vertices of the given ordered rooted tree.

→ 1)

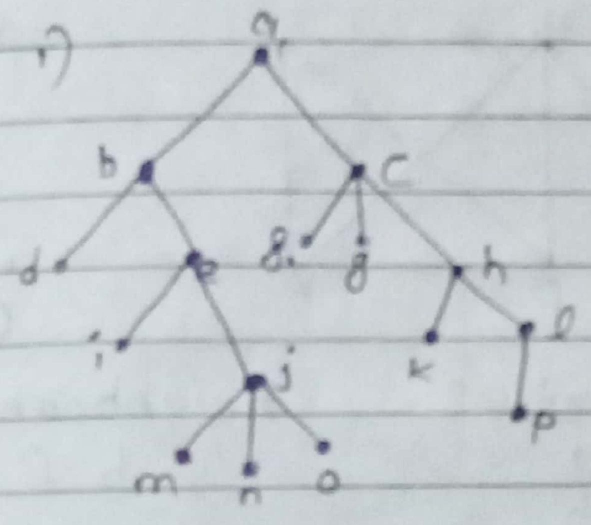
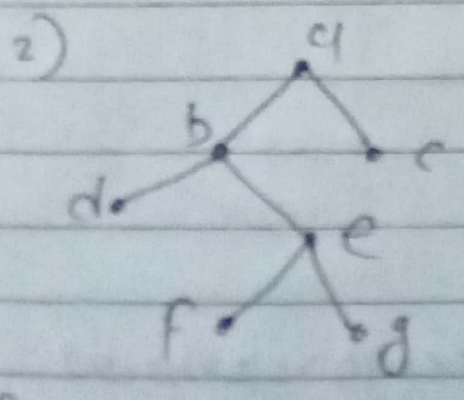


fig. a

Preorder traversal is -
abdeijmnocfghklp

2)

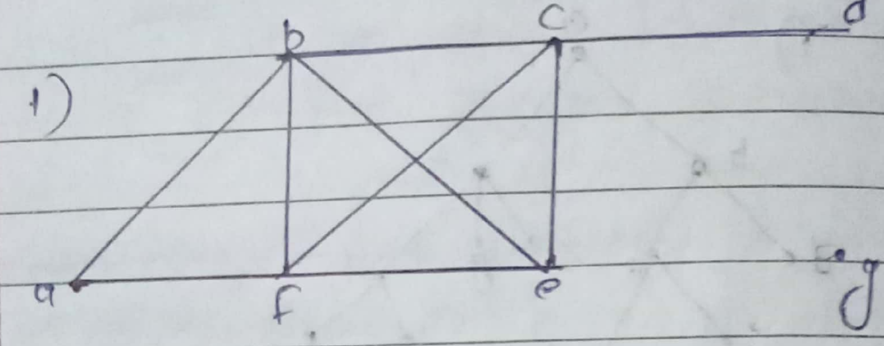


Preorder traversal is - abdefgc

Unit - 6

Q 1) What are the degree and What are the neighborhoods of the vertices in the graphs G and H displayed in below fig

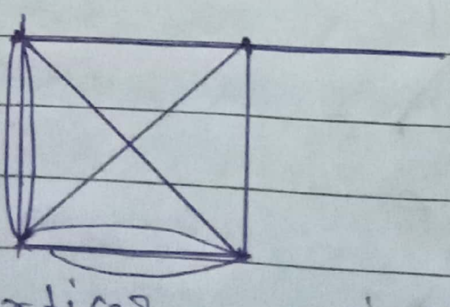
1)



Ans -

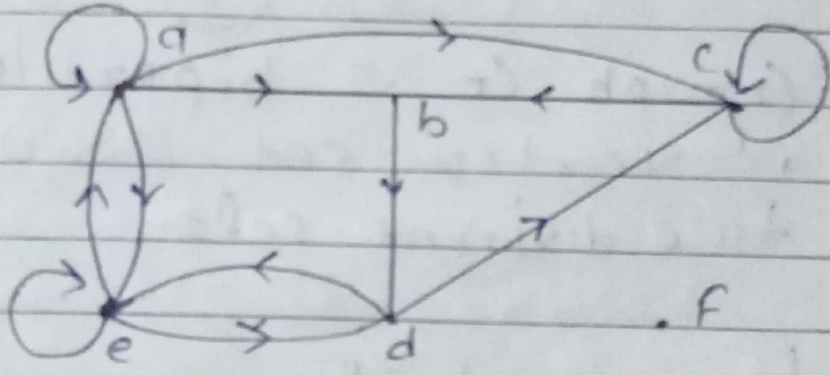
vertices	neighborhoods
a	b, f
b	c, a, f, e
c	b, e, d, f
d	c
e	f, b, c
f	a, b, c, e
g	—

2)



vertices	neighborhoods
a	e, b, d
b	b, e, a, d, e
c	b
d	e, a, b
e	b, a, d

Q2) Find the in degree and outdegree ~~and what are the neighborhoods of the vertices in graph G~~ and of each vertex in the graph G with directed edges shown in figures.

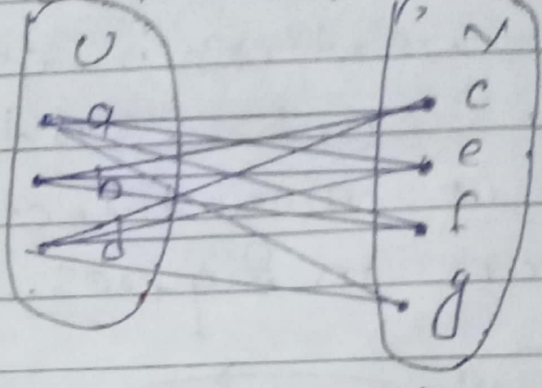


→

Indegree	outdegree	Total degree
d(a) = 2	d(a) = 3	5
d(b) = 2	d(b) = 1	3
d(c) = 3	d(c) = 2	5
d(d) = 2	d(d) = 2	4
d(e) = 3	d(e) = 3	6
		23

Q3) Are the graphs G and H displayed in figure are bipartile?

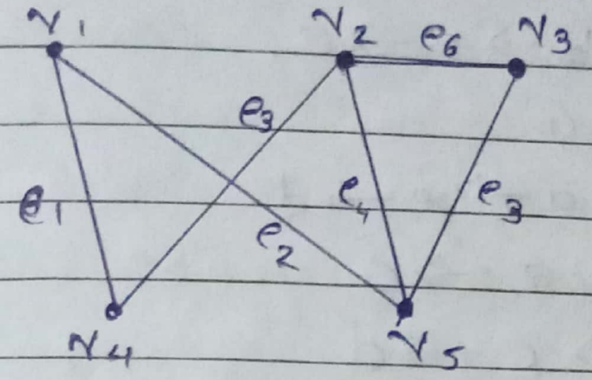
for Graph G let us consider
U = {a, b, d} , V = {c, e, f, g}



Graph G is bipartile because it vertex set is union of two disjoint sets.

For graph H -
Graph H is not bipartile because it's vertex set can't be partition into two subsets so that the edge do not connect two vertices from the same subset

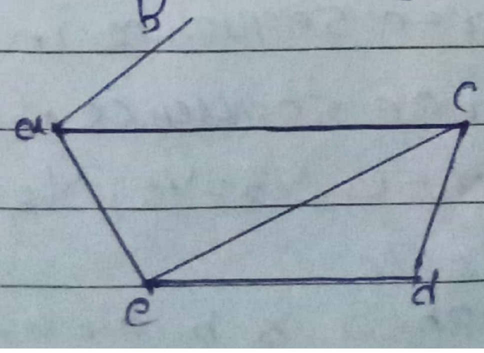
Q4) Represent the graph shown in fig. below with an incident matrix.



→ Incidence matrix of given graph is

	e_1	e_2	e_3	e_4	e_5	e_6
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	1	0	0	0	0
v_5	0	1	0	1	1	0

Q5) Use adjacency lists to describe the simple graph given in figure below



→ It is associated with link list as follow

a → b → c → e

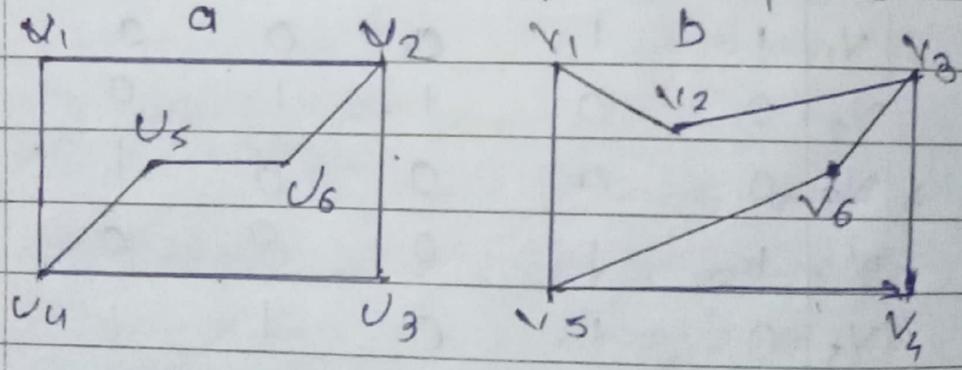
b → a

c → a → e → d

d → e → c

e → c → d

Q6) Determine whether the graphs G and H display below are Isomorphic.

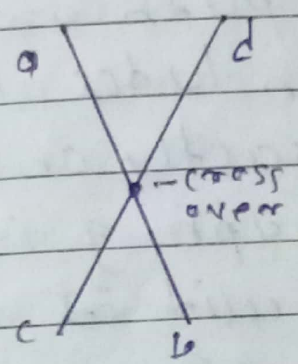


No. of Nodes in a = 6
No. of Nodes in b = 6
No. of edges in a = 7
No. of edges in b = 7
No. of degree sequence in a = 0
No. of degree sequence in b = 0
mapping - $v_1 = v_1, v_2 = v_5, v_3 = v_2, v_4 = v_3$
$v_5 = v_4, v_6 = v_6$
Hence graph a & b are Isomorphic

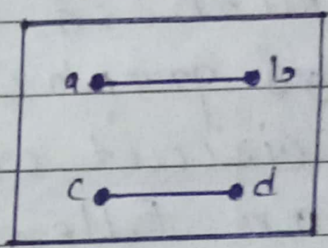
Q7) What is planar graph?
Explain with diagram.

→ A graph is said to be planor if it can be drawn on a plane so that no two edges cross each other at non-vertex point

ex-

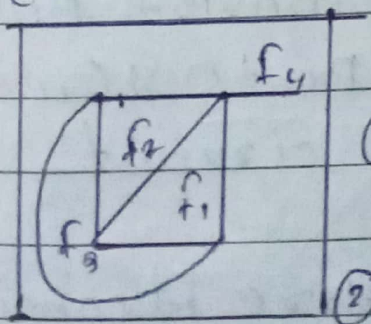


Plane



planor graph divides the surface into two parts
1) Interior region 2) Exterior region

ex-



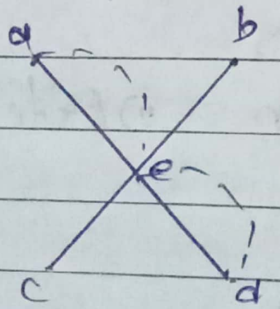
① Interior regions are - f_1, f_2, f_3

② exterior regions are - f_4

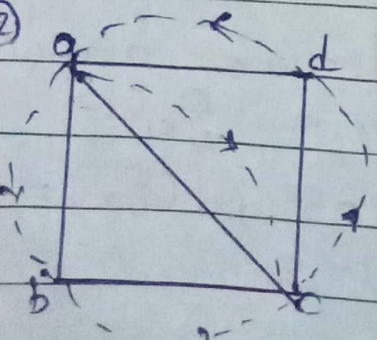
③ degree - $d(f_1) = 3$, $d(f_2) = 3$, $d(f_3) = 3$
$d(f_4) = 5$

Q8) What is Euler's & Hamilton path in a graph? Explain with diagram.

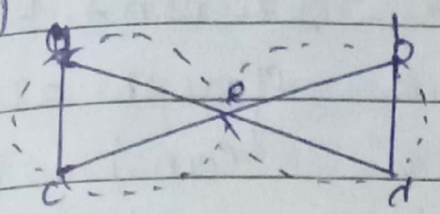
Euler Graph -
A path of a graph G is called as a Euler path or Eulerian path if it includes each edge of graph G exactly once. A circuit of graph g is called an Euler circuit if it includes each of graph G exactly once.

① 

It is not Euler graph as it doesn't consist Euler path & Euler circuit

② 

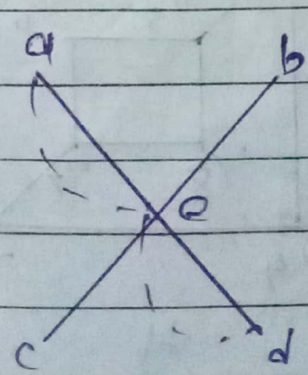
Here it consist Euler path but not Euler circuit so it is not completely Euler graph

(3) 

It is Euler graph as it consist Euler path and Euler circuit.

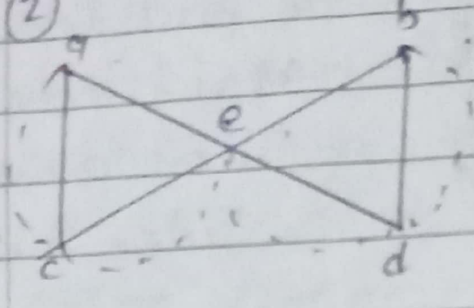
Hamiltoni graph -

A path of a graph G is called as a Hamiltonian path if it includes each vertex of graph G exactly once A circuit of a graph G is called as a Hamiltonian circuit if it include each vertex of G exactly once except the starting and end vertices which are one and the same, which appear twice a graph containing hamiltonian circuit is called as a hamiltonian graph

(1) 

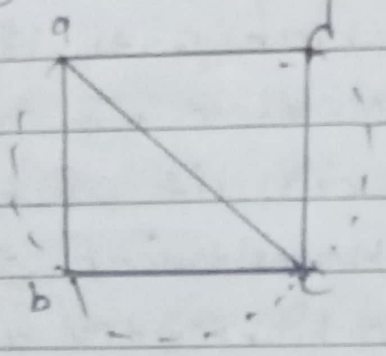
It isn't Hamilton graph as it doesn't consist hamiltonian path & circuit.

(2)



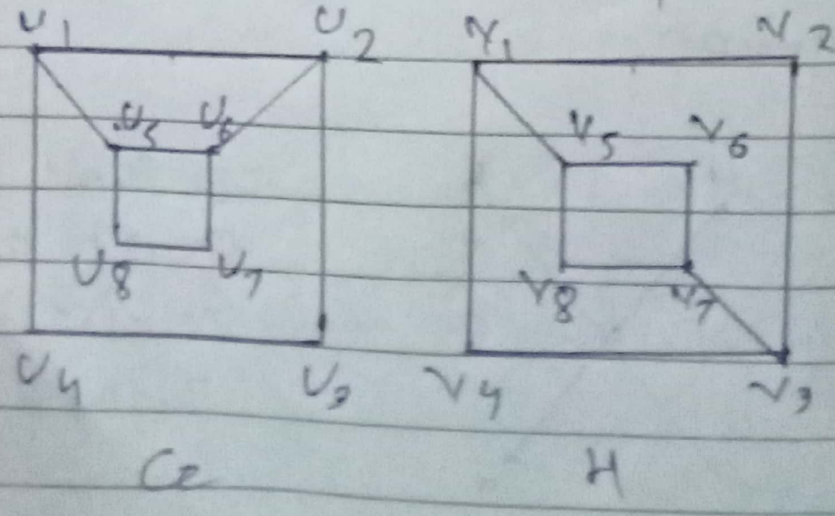
It isn't Hamitonian graph as it consist Hamiltonian path but not Hamiltonian circuit

(3)



It is Hamiltonian graph as it contain both hamiltonian path and hamitonian circuit

Q9) Determine whether the graph G and H displayed in figure below are isomorphic.



→ No. of edges in G = 10
No. of edges in H = 10
No. of vertices in G = 8
No. of vertices in H = 8
Mapping
$U_1 = V_1$
$U_2 = V_3$ ∴ Hence, it is isomorphic.
$U_3 = V_4$
$U_4 = V_2$
$U_5 = V_5$
$U_6 = V_7$
$U_7 = V_6$
$U_8 = V_8$

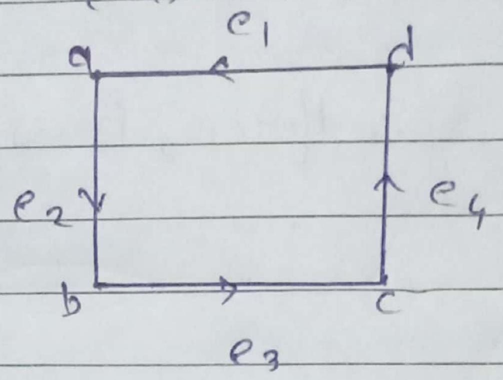
10) What are the types of graph and explain with situable example.
→ The ordered pairs of set of vertices and edges is called as graph.
following are the 10 types of graph

1) Directed graph

A graph where edges are directed

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

$E = \{e_1, e_2, e_3, e_4\}$



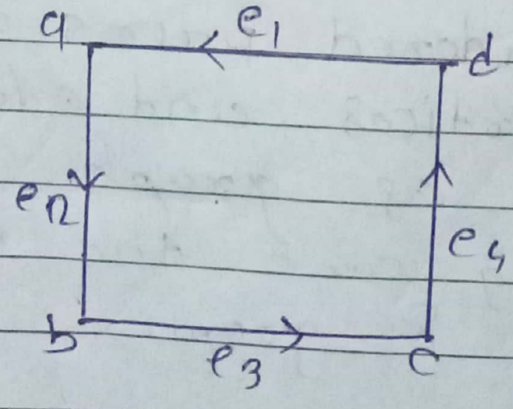
2) Unidirected graph

A graph where edges are not directed

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

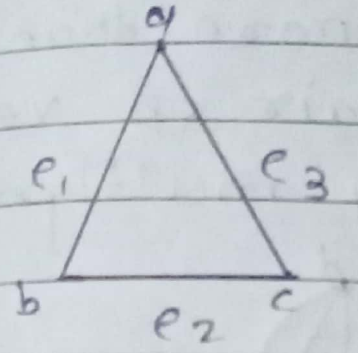
$(a, b) = (b, a)$

$E = \{e_1, e_2, e_3, e_4\}$



3) Finite graph

A graph is said to be finite no. of vertices is equal to no. of edges.



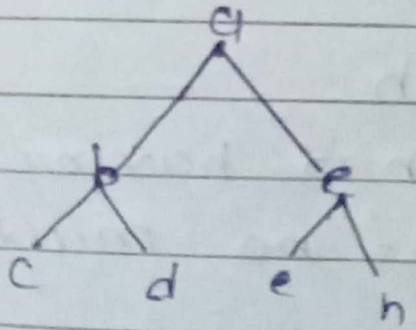
$G = \langle V, E \rangle$

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

$E = \{e_1, e_2, e_3\}$

4) Infinite graph

In a graph G both V & E are infinite set



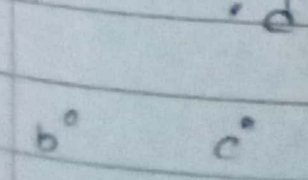
$V = \infty$, $E = \infty$

$E = \{e_1, e_2\}$

$V = \{v_1, v_2\}$

5) Null graph

A graph is said to be Null graph when every edge set is empty



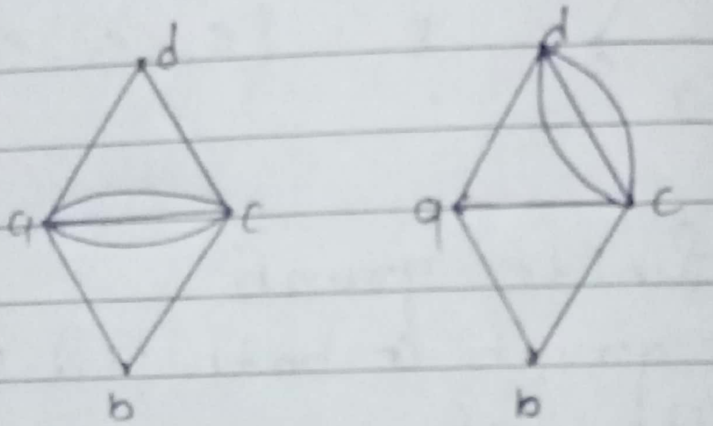
$G = \langle V, E \rangle$

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

$E = \{0\}$

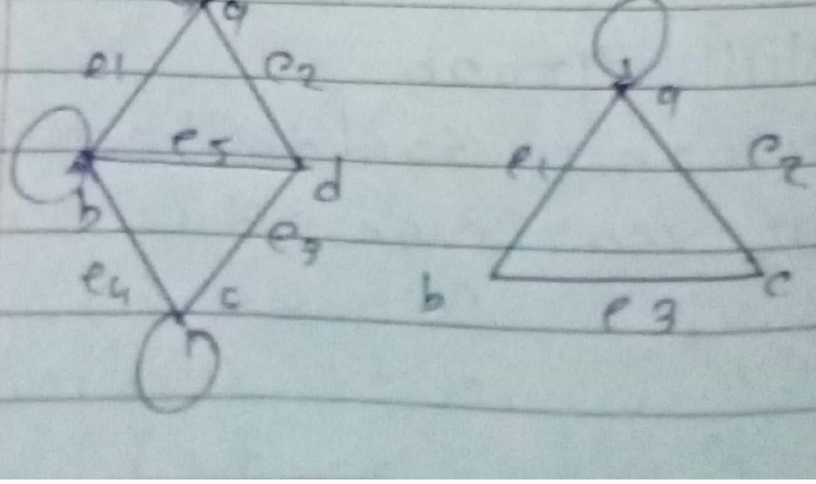
6) Multigraph

Any graph which consist some parallel edges and a graph which allows more than one edge to join pair of vertices is called as a multigraph

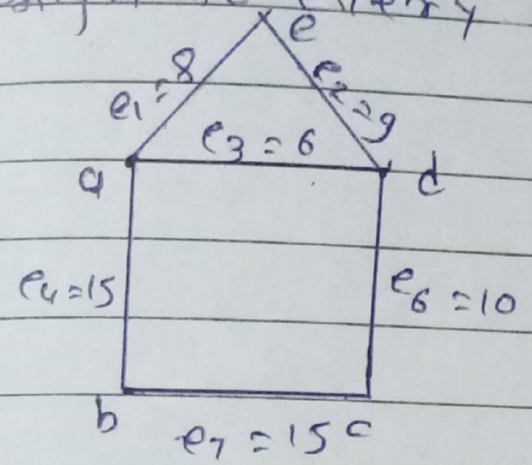


7) Pseudo graph

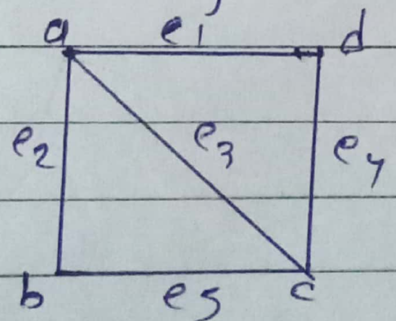
when a graph is having loops ~~and~~ but no multiple edges.



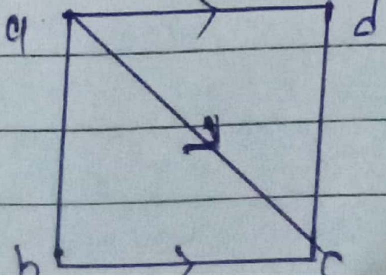
8) Weighted graph
A graph in which weight are assign to every edge



9) simple graph
A graph G in which no selfloop
such type of graph is simple graph



10) Mix graph
A graph which consist directed & undirected edges



unit - 1

Q) i) $(p \wedge q) \rightarrow (p \vee q)$

→

p	q	$p \wedge q$	$p \vee q$	$(p \wedge q) \rightarrow (p \vee q)$
T	T	T	T	T
T	F	F	T	F
F	T	F	T	F
F	F	F	F	T

Not tautology

ii) $(p \vee q) \wedge (\sim p \wedge \sim q)$

p	q	$\sim p$	$\sim q$	$p \vee q$	$\sim p \wedge \sim p$
T	T	F	F	T	F
T	F	F	T	T	F
F	T	T	F	T	F
F	F	T	T	F	T

$(p \vee q) \wedge (\sim p \wedge \sim q)$
F
F
F
F

Not tautology

iii) $(\sim p \wedge \sim q) \rightarrow (p \rightarrow q)$

p	q	$\sim p$	$\sim q$	$\sim p \wedge \sim q$	$p \rightarrow q$
T	T	F	F	F	T
T	F	F	T	F	F
F	T	T	F	F	F
F	F	T	T	T	T

$(\sim p \wedge \sim q) \rightarrow (p \rightarrow q)$
F
T
T
T

∴ Not tautology

iv) $(p \rightarrow q) \wedge (p \wedge \sim q)$

p	q	$\sim q$	$(p \rightarrow q)$	$(p \wedge \sim q)$	$(p \rightarrow q) \wedge (p \wedge \sim q)$
T	T	F	T	F	F
T	F	T	F	T	F
F	T	F	F	F	F
F	F	T	T	F	F

∴ Not tautology

v) $[p \wedge (p \to \sim q) \to q]$

p	q	$\sim q$	$(p \to \sim q)$	$(p \wedge (p \to \sim q))$	$[p \wedge (p \to \sim q) \to q]$
T	T	F	F	F	F
T	F	T	T	T	F
F	T	F	T	F	F
F	F	T	F	T	F
					F

$\therefore$ Not tautology

Q5) i) $[\sim p \wedge (p \vee q)] \to q$

p	q	$\sim p$	$(p \vee q)$	$\sim p \wedge (p \vee q)$	$\sim p \wedge (p \vee q) \to q$
T	T	F	T	F	F
T	F	F	T	F	F
F	T	T	T	T	T
F	F	T	F	F	T

$\therefore$ Not tautology

ii) $[(p \to q) \wedge (q \to r)] \to (p \to r)$

p	q	r	$p \to q$	$q \to r$	$p \to r$	$p \to q \wedge q \to r$	final
T	T	T	T	T	T	T	T
T	T	F	T	F	F	F	T
T	F	T	F	F	T	T	T
T	F	F	F	T	F	F	T
F	T	T		T	F	F	T
F	T	F	F	F	T	T	T
F	F	T	F	F	F	F	T
F	F	F	T	T	T	T	T

$\therefore$ Tautology

iii) $[p \wedge (p \to q)] \to q$

p	q	$p \to q$	$p \wedge (p \to q)$	$[p \wedge (p \to q)] \to q$
T	T	T	T	T
T	F	F	F	T
F	T	F	T	T
F	F	T	F	T

$\therefore$ tautology

iv) $[(p \vee q) \wedge (p \to q) \wedge (q \to r)] \to r$

p	q	r	$(p \vee q)$	$(p \to q)$	$(q \to r)$	1	2	3
T	T	T	T	T	T	T	F	F
T	T	F	T	T	F	T	F	T
T	F	T	T	F	F	F	F	F
T	F	F	T	F	T	F	F	T
F	T	T	T	F	T	F	F	F
F	T	F	F	F	F	F	T	F
F	F	T	F	T	F	F	T	T
F	F	F	F	T	T	F	F	T

$\therefore$ Not tautology

q7) a) $[(p \to q) \wedge (q \to r)] \to (p \to r)$

p	q	r	$(p \to q)$	$(q \to r)$	$(p \to r)$
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	F	T
T	F	F	F	T	F
F	T	T	F	T	F
F	T	F	F	F	F
F	F	T	T	F	F
F	F	F	T	T	F

$[(p \to q) \wedge (q \to r)]$	$[(p \to q) \wedge (q \to r)] \to (p \to r)$
T	T
F	T
F	F
F	T
F	T
F	F
F	T
F	F

$\therefore$ Not tautology

b) $[p \wedge (p \to q)] \leftrightarrow q$

p	q	$(p \to q)$	$p \wedge (p \to q)$	$[p \wedge (p \to q)] \to q$
T	T	T	T	T
T	F	F	F	T
F	T	F	F	F
F	F	T	F	T

$\therefore$ Not tautology

c) $\underbrace{[p \wedge (p \to q)]}_{1} \underbrace{\to (q \to r)]}_{2} \underbrace{\to r}_{3}$

p	q	r	$(p \to q)$	$(q \to r)$	1	2	3
T	T	T	T	T	T	T	T
T	T	F	T	F	T	F	T
T	F	T	F	F	F	T	T
T	F	F	F	T	F	F	T
F	T	T	F	T	F	F	T
F	T	F	F	F	F	T	F
F	F	T	T	F	F	T	T
F	F	F	T	T	F	F	T

$\therefore$ Not tautology

d) $\cancel{[(p \vee q) \wedge (p \to q) \wedge (q \to r)]} \Rightarrow r$

q8) $(p \wedge q) \to r$ and $(p \to r) \wedge (q \to r)$

p	q	r	$(p \wedge q)$	$(p \wedge q) \to r$ (1)	$(p \to r)$	$(q \to r)$
T	T	T	T	T	T	T
T	T	f	T	f	F	F
T	f	T	f	f	T	F
T	F	f	F	T	F	T
f	T	T	F	F	F	T
F	T	f	f	T	T	f
f	f	T	F	f	F	F
f	f	f	F	T	T	T

$[(p \wedge q) \to r] \to [(p \to r) \wedge (q \to r)]$	$(p \to r) \wedge (q \to r)$ (2)
T	T
T	F
T	F
F	f
T	f
f	F
f	f
T	T

Not logically equivalent.
from column No. (1) & (2)

Q4) $\sim(p \vee (\sim p \wedge q))$ and $\sim p \wedge \sim q$

p	q	$\sim p$	$\sim q$	$(\sim p \wedge q)$	$(p \vee (\sim p \wedge q))$
T	T	F	F	F	T
T	F	F	T	F	T
F	T	T	F	T	T
F	F	T	T	F	F

① $\sim(p \vee (\sim p \wedge q))$	② $\sim p \wedge \sim q$
F	F
F	F
F	F
T	T

∴ from column ① & ②
it is logically equivalent.

Unit - 2

Q.1) What is function and composition of function? Explain basic types with examplem.

→ A function is a relation between set of input and set of corresponding output each input is related to exactly 1 ob

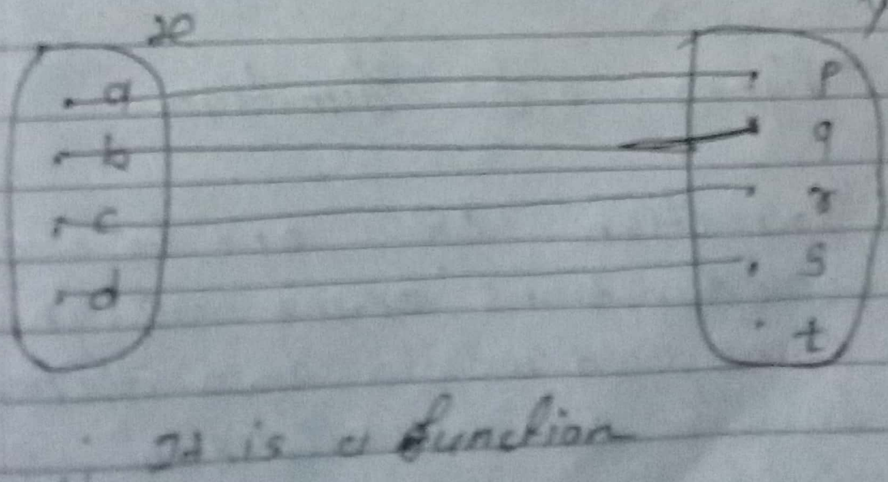
If f is a function from a to b. A & B are nonempty sets.

A function f from set A to set B Is an assignment of exactly 1 element of B to each element of A

functions are also called as a mapping of transformation

ex-

set of children   set of women



It is a function

Composition of functions:

Let f be the function from set A to B and let g be a function from set B to set C.

Let ~~t~~ f

$f: A \rightarrow B$

Let g

$g: B \rightarrow C$.

then composition of function f g written as $g(f(x))$

composition is called as a mapping or transformation of a function in composition of function output of one relation becomes the input for another relation

$g(f(x)) : A \rightarrow C$

Types of function

1) Injection OR one to one

- some functions never assigned the same value to two different domain element this funⁿ are if and only if $f(x) \neq f(y)$

ex

$$
\begin{array}{ccc}
x & \xrightarrow{f(x)=x^2} & y \\
1 & \rightarrow & 1 \\
2 & \rightarrow & 4 \\
3 & \rightarrow & 9 \\
4 & \rightarrow & 16 \\
 & & 25
\end{array}
$$

2) Surjection

$$
\begin{array}{ccc}
x & f(x)=x^2 & y \\
1 & \rightarrow & 1 \\
2 & \rightarrow & 4 \\
3 & \rightarrow & 4 \\
4 & \rightarrow & 16
\end{array}
$$

A function f from A to B is called onto or surjective funⁿ if and only if for every element $b \in B$, $a \in B$ with $f(a) = b$. A function is called as surjection.

3) Bijective

- A function f is one to one and onto then it is called as bijection.

ex -

x	$f(x) = x^2$	y
1	$\longrightarrow$	1
2	$\longrightarrow$	4
3	$\longrightarrow$	9
4	$\longrightarrow$	16
5	$\longrightarrow$	25

4) Inverse

x		y
1	$\longrightarrow$	1
2	$\longrightarrow$	4
3	$\longrightarrow$	9
4	$\longrightarrow$	16
5	$\longrightarrow$	25

Q2) $R = \{(1,1), (1,3), (3,2), (3,4), (4,2)\}$
$S = \{(2,1), (3,3), (3,4), (4,1)\}$
$R \circ S$, $S \circ R$, $R \circ R$, $S \circ S$, $(R \circ S) \circ R$,
$R \circ (S \circ R)$, R^3

→ 1) $R \circ S$

R	S
(1,1)	(2,1)
(1,3)	(3,3)
(3,2)	(3,4)
(3,4)	(4,1)
(4,2)	

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

2) $S \circ R$

S	R
(2,1)	(1,1)
(3,3)	(1,3)
(3,4)	(3,2)
(4,1)	(3,4)
	(4,2)

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

3) $R \circ R$

R	R
(1,1)	(1,1)
(1,3)	(1,3)
(3,2)	(3,2)
(3,4)	(3,4)
(4,2)	(4,2)

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

4) $S \circ S$

S	S
(2,1)	(2,1)
(3,3)	(3,3)
(3,4)	(3,4)
(4,1)	(4,1)

$S \circ S = \{(3,3), (3,4), (3,1)\}$

5) $(R \circ S) \circ R$

$R \circ S$	R
(1,3)	(1,1)
(1,4)	(1,3)
(3,1)	(3,2)
(4,1)	(3,4)
	(4,2)

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

6) $R \circ (S \circ R)$

R	$S \circ R$
(1,1)	(2,1)
(1,3)	(2,3)
(3,2)	(3,2)
(3,4)	(3,4)
(4,2)	(4,1)
	(4,3)

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

7) R^3

$(R \circ R) \circ R$

$(R \circ R)$	R
(1,1)	(1,1)
(1,3)	(1,3)
(1,2)	(3,2)
(1,4)	(3,4)
(3,4)	(4,2)

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

Unit 4

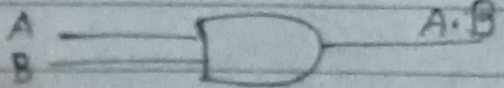
Q1) What are the logic gates, Symbols and operations

→ Logic gates are devices they are use to desine electronics circuit electronic equipment for performing same operation on it
There are three types of logic gates

1) Basic logic Gates
2) Universal logic Gates
3) Arithmetic logic Gates.

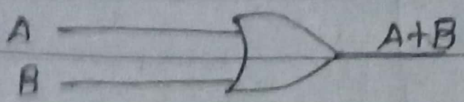
1) Basic logic Gates
a) And
b) OR
c) Not

a) And



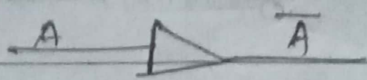
x	y	AB
0	0	0
0	1	0
1	0	0
1	1	1

b) OR



A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

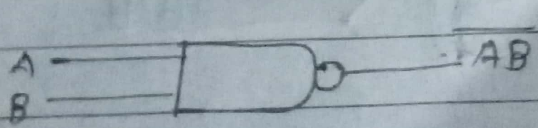
c) Not



A	B	$\overline{A}$
0		1
1		0

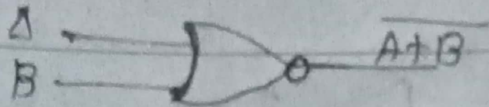
2 2) Universal logic Gates
a) Nand
b) Nor

a) Nand



A	B	$\overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

b) Nor



A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

3) Arithmetic logic Gates

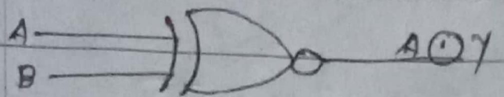
a) XOR

b) XNOR

a) XOR

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

b) XNOR



A	B	$A \odot B$
0	0	1
0	1	0
1	0	01
1	1	1

Q2) i) $x\bar{y}$ ii) $xy + \bar{x}\bar{y}$ c) $xy + x\bar{y} + \bar{x}y + \bar{x}\bar{y}$

→

i) $x\bar{y}$

$x \backslash y$	0	1
0	$_0$	$_1$
1	1 $_2$	$_3$

$f = x\bar{y}$

ii) $xy + \bar{x}\bar{y}$

$x \backslash y$	0	1
0	1_0	$_1$
1	$_2$	1_3

$f = xy + \bar{x}\bar{y}$

$f = x \odot y$

iii) $xy + x\bar{y} + \bar{x}y + \bar{x}\bar{y}$

$x \backslash y$	0	1
0	1_0	1_1
1	1_2	1_3

$f = xy + x\bar{y} + \bar{x}y + \bar{x}\bar{y}$

$f = 1$

Q 5) i) $\bar{x}yz + \bar{x}\bar{y}z$

ii) $xy\bar{z} + \bar{x}yz + xyz + \bar{x}y\bar{z}$

→ i)

$x \backslash yz$	00	01	11	10
0	0	1 1	1 3	2
1	4	5	7	6

$f = \bar{x}z$

ii)

$x \backslash yz$	00	01	11	10
0	0	1	1 3	1 2
1	4	5	1 7	1 6

$f = y$