TYPE-A:BOOLEAN LOGIC:CH-3

1. What do you understand by 'truth value' and 'truth function'? How are these related ?

sol:
A truth value shows whether a statement is true or false.
There are only two truth values:

- True (T)
- False (F)

A truth function is a rule used to find the truth value of a combined statement from the truth values of simple statements.

Examples of truth functions are:
AND, OR, NOT, IF…THEN.

2. What do you understand by 'logical function'? What is its alternative name? Give examples for logical functions.

sol:
A logical function is a rule used to combine one or more statements and find their truth value.
It takes the truth values of simple statements and gives the truth value of a new statement.
Logical functions help us decide whether a combined statement is true or false.

Another name for a logical function is Truth Function.

Some common logical functions are:

1. AND ($\wedge$)
   True only when both statements are true.

2. OR ($\vee$)
   True when at least one statement is true.

3) What is meant by tautology and fallacy? Prove that 1+ Y is a tautology and 0. Y is a fallacy.

sol:
A tautology is a logical expression that is always true, no matter what the values of the variables are.
It is true for all possible cases.

A fallacy is a logical expression that is always false, no matter what the values of the variables are.

It is false for all possible cases.

Y	**1 + Y**	**0 · Y**
0	1	0
1	1	0

4. What is a truth table? What is its significance ?

sol:

A truth table is a table that shows all possible truth values of logical statements and their final results.

It lists every possible combination of truth values and shows whether the given expression is true or false.

The significance of a truth table is that it helps us understand and test logical expressions clearly and correctly.

It is important because:

1. It shows the result of every possible case.
2. It helps to check whether an expression is a tautology, fallacy, or normal statement.
3. It helps to compare different logical expressions.
4. It reduces mistakes in logical reasoning.
5. It is useful in computer science and digital circuits.

5. What are the basic postulates of boolean algebra ?

sol:

Basic Postulates of Boolean Algebra

Boolean algebra is based on some basic rules called postulates.
These postulates help us to perform logical operations easily.

Let X and Y be two Boolean variables.
Each variable can have only two values: 0 and 1.

1. Closure Postulate

For any two variables X and Y, the result of
X + Y and X · Y is also either 0 or 1.
So, Boolean operations always give a Boolean value.

2. Identity Postulate

There are identity elements for addition and multiplication.
X + 0 = X
X · 1 = X

3. Commutative Postulate

The order of variables does not change the result.
X + Y = Y + X
X · Y = Y · X

4. Associative Postulate

The grouping of variables does not change the result.
(X + Y) + Z = X + (Y + Z)
(X · Y) · Z = X · (Y · Z)

5. Distributive Postulate

Each operation can be distributed over the other.
X · (Y + Z) = (X · Y) + (X · Z)
X + (Y · Z) = (X + Y) · (X + Z)

6. Complement Postulate

Every variable has a complement.
$X + \bar{X} = 1$
$X \cdot \bar{X} = 0$
(Here $\bar{X}$ means NOT X)

7. Null (Dominance) Postulate

Some values always dominate the result.
X + 1 = 1
X · 0 = 0

6. What does duality principle state? What is its usage in boolean algebra ?

sol:
The principle of duality in Boolean algebra states that:
Every Boolean expression has a dual expression.
The dual is obtained by:

- Replacing + (OR) with · (AND)
- Replacing · (AND) with + (OR)
- Replacing 0 with 1
- Replacing 1 with 0
- The variables remain the same.
-

The principle of duality is useful because:

- It helps to get a new Boolean law from an existing one.
- It saves time by avoiding separate proofs.
- It helps in simplifying logical expressions.
- It is useful in designing digital circuits.
- It helps in checking the correctness of formulas.

7. State the distributive laws of boolean algebra. How do they differ from the distributive laws of ordinary algebra ?

sol:

1. First Distributive Law
   AND distributes over OR:
   $X \cdot (Y + Z) = (X \cdot Y) + (X \cdot Z)$

2. Second Distributive Law
   OR distributes over AND:
   $X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$

Ordinary Algebra	Boolean Algebra
Only one distributive law	Two distributive laws
$a(b + c) = ab + ac$	$X \cdot (Y+Z) = X \cdot Y + X \cdot Z$
$a + (bc)$ is not distributive	$X + (Y \cdot Z) = (X+Y)(X+Z)$

8. Prove the complementarity law of boolean algebra with the help of a truth table.

sol:

X	$\bar{X}$	$X + \bar{X}$	$X \cdot \bar{X}$
0	1	1	0
1	0	1	0

9. Give the truth table proof for distributive law of boolean algebra.

sol:

X	Y	Z	$Y + Z$	$X \cdot (Y + Z)$	$X \cdot Y$	$X \cdot Z$	$(X \cdot Y) + (X \cdot Z)$	$X + Y$	$X + Z$	$(X + Y) \cdot (X + Z)$	$Y \cdot Z$	$X + (Y \cdot Z)$
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	1	0	0	0
0	1	0	1	0	0	0	0	1	0	0	0	0
0	1	1	1	0	0	0	0	1	1	1	1	1
1	0	0	0	0	0	0	0	1	1	1	0	1
1	0	1	1	1	0	1	1	1	1	1	0	1
1	1	0	1	1	1	0	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1

10. Give algebraic proof of absorption law of boolean algebra.

sol:

The absorption laws are:

$X + (X \cdot Y) = X$

$X \cdot (X + Y) = X$

Algebraic Proof
Let X and Y be Boolean variables.

1. Proof of: X + (X · Y) = X
   Start with the left-hand side (LHS):
   X + (X · Y)
   Using distributive law:
   = (X · 1) + (X · Y)
   = X(1 + Y)
   Using identity law (1 + Y = 1):
   = X · 1
   = X
   So,
   X + (X · Y) = X

2. Proof of: X · (X + Y) = X
   Start with the left-hand side (LHS):
   X · (X + Y)
   Using distributive law:
   = (X · X) + (X · Y)
   Using idempotent law (X · X = X):
   = X + (X · Y)
   Using absorption law (proved above):
   = x
   So,
   X · (X + Y) = X

11. What are DeMorgan's theorems ? Prove algebraically the DeMorgan's theorem.

sol:
First Theorem:
The complement of a sum is equal to the product of the complements.
$(X+Y)' = X' \cdot Y'$

Second Theorem
The complement of a product is equal to the sum of the complements.
$(X \cdot Y)' = X' + Y'$

12. Which Boolean Law/theorem do you use to determine complement of a Boolean expression?

sol:
The DeMorgan's Theorems are used to find the complement of any Boolean expression.
They state that:
$(X + Y)' = X' \cdot Y'$
$(X \cdot Y)' = X' + Y'$

13. Are dual and complement of a Boolean expression the same?

sol:

No, dual and complement of a Boolean expression are not the same.
They are different concepts in Boolean algebra.

14 Are dual and complement of a Boolean expression related ?

sol:
Yes, dual and complement of a Boolean expression are related, but they are not the same.
They are connected through DeMorgan's Theorems and the principle of duality.

15. Draw a truth table of 2 input NAND and 3 input NAND.

sol:

2 input

A	B	A · B	(A · B)′ (Output)
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

3 input

A	B	C	A · B · C	(A · B · C)′ (Output)
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

16. Draw a truth table of 2 variable and 3 variable NOR.

sol:

2 input

A	B	A + B	(A + B)′ (Output)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

3 input

A	B	C	A + B + C	(A + B + C)′ (Output)
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

17. Draw a truth table of 2 variable and 3 variable XOR.

sol:

2 input

A	B	$A \oplus B$ (Output)
0	0	0
0	1	1
1	0	1
1	1	0

3 input

A	B	C	$A \oplus B \oplus C$ (Output)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

18. Draw a truth table of 2 variable and 3 variable XNOR.

sol:

2 input

A	B	$A \oplus B$	$(A \oplus B)'$ (Output)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

3 input

A	B	C	$A \oplus B \oplus C$	$(A \oplus B \oplus C)'$ (Output)
0	0	0	0	1

0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

19. What are logic circuits ?

sol:
Logic circuits are electronic circuits that work using the rules of Boolean algebra.
They use logic gates to process information.
These circuits take binary inputs (0 and 1) and produce binary outputs (0 and 1).

20. How do you draw logic circuit diagrams ?

sol:
To draw a logic circuit diagram:

- Write the given Boolean expression.
- Identify the logic operations (AND, OR, NOT).
- Draw the required logic gates.
- Connect the input variables to the gates.
- Connect the gates step by step.
- Label the final output.