4. Operators in C

• THEORY EXERCISE: Write notes explaining each type of operator in C: arithmetic, relational, logical, assignment, increment/decrement, bitwise, and conditional operators.

1.Arithmetic Operators

Arithmetic operators are used to perform basic mathematical operations.

- **Addition (+)**: Adds two operands.
  - Example: **a + b**
- **Subtraction (-)**: Subtracts the second operand from the first.
  - Example: **a - b**
- **Multiplication (*)**: Multiplies two operands.
  - Example: **a * b**
- **Division (/)**: Divides the numerator by the denominator.
  - Example: **a / b** (Note: If both **a** and **b** are integers, the result is also an integer.)
- **Modulus (%)**: Returns the remainder of a division operation.
  - Example: **a % b**

2.Relational Operators

Relational operators are use to compare two number value.

Operator	**Meaning**	**Example**	**True when...**
==	Equal to	a == b	a is equal to b
!=	Not equal to	a != b	a is not equal to b
>	Greater than	a > b	a is greater than b
<	Less than	a < b	a is less than b
>=	Greater than or equal to	a >= b	a is greater or equal b
<=	Less than or equal to	a <= b	a is less or equal

3. Logical Operators

Logical operators are used to combine multiple conditions.

- **Logical AND (&&)**: Returns true if both operands are true.
    - Example: **(a > b) && (c > d)**
- **Logical OR (||)**: Returns true if at least one of the operands is true.
    - Example: **(a > b) || (c > d)**
- **Logical NOT (!)**: Reverses the logical state of its operand.
    - Example: **!(a > b)**

4. Assignment Operators

Assignment operators are used to assign values to variables.

- **Simple assignment (=)**: Assigns the right operand's value to the left operand.
    - Example: **a = b**
- **Add and assign (+=)**: Adds the right operand to the left operand and assigns the result to the left operand.
    - Example: **a += b** (equivalent to **a = a + b**)
- **Subtract and assign (-=)**: Subtracts the right operand from the left operand and assigns the result to the left operand.
    - Example: **a -= b**
- **Multiply and assign (*=)**: Multiplies the left operand by the right operand and assigns the result to the left operand.
    - Example: **a *= b**
- **Divide and assign (/=)**: Divides the left operand by the right operand and assigns the result to the left operand.
    - Example: **a /= b**
- **Modulus and assign (%=)**: Takes the modulus using two operands and assigns the result to the left operand.
    - Example: **a %= b**

5. Increment/Decrement Operators

These operators are used to increase or decrease the value of a variable by one.

- **Increment (++)**: Increases the value of a variable by one.
    - Example: **++a** (pre-increment) or **a++** (post-increment)
- **Decrement (--)**: Decreases the value of a variable by one.
    - Example: **--a** (pre-decrement) or **a--** (post-decrement)

6. Bitwise Operators

Bitwise operators perform operations on binary representations of integers.

- **Bitwise AND (&)**: Compares each bit of two operands; returns 1 if both bits are 1.
    - Example: **a & b**
- **Bitwise OR (|)**: Compares each bit of two operands; returns 1 if at least one bit is 1.
    - Example: **a | b**
- **Bitwise XOR (^)**: Compares each bit of two operands; returns 1 if the bits are different.
    - Example: **a ^ b**
- **Bitwise NOT (~)**: Inverts all bits of the operand.
    - Example: **~a**
- **Left shift (<<)**: Shifts bits to the left, filling with