OPERATORS

Programming is the process of designing and building an executable computer program to accomplish a specific task. It involves writing, testing, and maintaining source code written in a programming language.

Algorithm:

An algorithm is a step-by-step procedure or formula for solving a problem or accomplishing a task. It serves as the foundation for writing computer programs.

Source Code:

Source code refers to the human-readable instructions written in a programming language. It is the input that a compiler or interpreter processes to create an executable program.

Compiler:

A compiler is a program that translates source code written in a high-level programming language into machine code or an intermediate code that can be executed by a computer.

Interpreter:

An interpreter is a program that directly executes source code without the need for compilation. It translates and executes code line by line.

C Programming Language:

C Programming Language:

C is a general-purpose programming language created by Dennis Ritchie in the early 1970s. It is widely used for developing system software and applications.

Variable:

A variable is a named storage location in a program that holds a value. The value of a variable can change during the execution of the program.

Data Type:

A data type is a classification of data that determines the type of operations that can be performed on it. In C, data types include int, float, char, etc.

Function:-

A function is a modular unit of a program that performs a specific task. It consists of a set of instructions that are

executed when the function is called.

Array:

An array is a collection of elements, each identified by an index or a key. In C, arrays are used to store multiple values of the same data type.

Pointer:

A pointer is a variable that stores the memory address of another variable. It allows direct manipulation of memory and is a powerful feature in C.

Conditional Statements:

Conditional statements, such as if, else, and switch, are used to make decisions in a program based on specified conditions.

Loop:

A loop is a control structure that repeats a set of instructions until a specific condition is met. Examples include for, while, and do-while loops.

Boolean and Logical Operations:

Boolean is a data type that represents two possible values: true or false. It is commonly used in programming to make decisions based on

conditions.

Logical AND (&&):

The logical AND operator returns true if both operands are true; otherwise, it returns false.

Logical OR (||):

The logical OR operator returns true if at least one of the operands is true; otherwise, it returns false.

Logical NOT (!):

The logical NOT operator negates the value of its operand. If the operand is true, it returns false, and vice versa.

Certainly! Let's start with a comprehensive tutorial on operators in C, covering various types of operators, and then we'll delve into information about booleans in C.

Operators in C:

Operators in C are symbols that perform operations on operands. Operands can be variables, constants, or expressions. Here are the main categories of operators in C:

1. Arithmetic Operators:

+ // Addition

- // Subtraction

* // Multiplication

/ // Division

% // Modulus (remainder after division)

2. Relational Operators:

== // Equal to

!= // Not equal to

> // Greater than

< // Less than

>= // Greater than or equal to

<= // Less than or equal to

3. Logical Operators:

&& // Logical AND

|| // Logical OR

! // Logical NOT

4. Bitwise Operators:

& // Bitwise AND

| // Bitwise OR

^ // Bitwise XOR

~ // Bitwise NOT (One's complement)

<< // Left shift

>> // Right shift

5. Assignment Operators:

= // Assignment

+= // Addition assignment

-= // Subtraction assignment

*= // Multiplication assignment

/= // Division assignment

%= // Modulus assignment

6. Increment/Decrement Operators:

++ // Increment

-- // Decrement

7. Conditional (Ternary) Operator:

condition ? expr1 : expr2

// If the condition is true, expr1 is evaluated; otherwise, expr2 is evaluated.

8. sizeof Operator:

sizeof(type)

// Returns the size of the specified data type in bytes.

9. Comma Operator:

expr1, expr2, ..., exprn

// Evaluates each expression from left to right and returns the value of the last expression.

Booleans in C:

In C, the standard does not have a boolean data type, but the <stdbool.h> header introduced in the C99 standard provides a boolean data type and related macros.

1. Boolean Data Type:

#include <stdbool.h>

bool // Represents a boolean data type with values true or false

2. Boolean Literals:

true // Represents the boolean value true

false // Represents the boolean value false

3. Boolean Operations:

&& // Logical AND

|| // Logical OR

! // Logical NOT

Example:

```
#include <stdio.h>
#include <stdbool.h>

int main() {
    bool a = true;
    bool b = false;

    printf("a && b = %d\n", a && b);    // Output: 0 (false)
    printf("a || b = %d\n", a || b);    // Output: 1 (true)
    printf("!a = %d\n", !a);                    // Output: 0 (false)

    return 0;
}
```