- Modulus operator (%) can only be applied on integer operands, and not on float or double operands.
- Equality operators have lower precedence than relational operators.
- Like arithmetic expressions, logical expressions are evaluated from left to right.
- Both x++ and ++x increment the value of x, but in the former case, the value of x is returned before it is incremented. Whereas in the latter case, the value of x is returned after it is incremented.
- Conditional operator is also known as ternary operator as it takes three operands.
- Bitwise NOT or complement produces one's complement of a given binary number.
- Among all the operators, comma operator has the lowest precedence.
- `sizeof` is a unary operator used to calculate the size of data types. This operator can be applied to all data types.
- While type conversion is done implicitly, typecasting has to be done explicitly by the programmer. Typecasting is done when the value of one data type has to be converted into the value of another data type.
- C supports three types of control statements: decision control statements, iterative statements, and jump statements.
- In a `switch` statement, if the value of the variable does not match with any of the values of case statements, then default case is executed.
- Iterative statements are used to repeat the execution of a list of statements until the specified expression becomes false.
- The `break` statement is used to terminate the execution of the nearest enclosing loop in which it appears.
- When the compiler encounters a `continue` statement, then the rest of the statements in the loop are skipped and the control is unconditionally transferred to the loop-continuation portion of the nearest enclosing loop.
- A C program contains one or more functions, where each function is defined as a group of statements that perform a specific task.
- Every C program contains a `main()` function which is the starting point of the program. It is the function that is called by the operating system when the user runs the program.
- Function declaration statement identifies a function's name and the list of arguments that it accepts and the type of data it returns.
- Function definition, on the other hand, consists of a function header that identifies the function, followed by the body of the function containing the executable code for that function. When a function is defined, space is allocated for that function in the memory.
- The moment the compiler encounters a function call, the control jumps to the statements that are a part of the called function. After the called function is executed, the control is returned back to the calling function.
- Placing the function declaration statement prior to its use enables the compiler to make a check on the arguments used while calling that function.
- A function having `void` as its return type cannot return any value. Similarly, a function having `void` as its parameter list cannot accept any value.
- Call by value method passes values of the variables to the called function. Therefore, the called function uses a copy of the actual arguments to perform its intended task. This method is used when the function does not need to modify the values of the original variables in the calling function.
- In call by reference method, addresses of the variables are passed by the calling function to the called function. Hence, in this method, a function receives an implicit reference to the argument, rather than a copy of its value. This allows the function to modify the value of the variable and that change is reflected in the calling function as well.
- A pointer is a variable that contains the memory address of another variable.
- The & operator retrieves the address of the variable.
- We can 'dereference' a pointer, i.e., refer to the value of the variable to which it points by using unary * operator.
- Null pointer is a special pointer variable that does not point to any variable. This means that a null pointer does not point to any valid memory address. To declare a null pointer we may use the predefined constant NULL.
- A generic pointer is pointer variable that has `void` as its data type. The generic pointer can point to variables of any data type.
- To declare pointer to pointers, we need to add an asterisk (*) for each level of reference.