

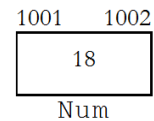
Chapter 6: Data Types and Operators

Data types: These are the means to identify the nature of the data and the set of operations that can be performed on the data.

Fundamental data types: These are the built-in data types of C++. Also known as basic data types or atomic data types. The five fundamental data types in C++ are char, int, float, double and void.

Data type modifiers: The keywords used to alter the size, range or precision of data supported by the basic data types. Important modifiers are signed, unsigned, long and short.

Variables: The names given to memory locations, by which data in these locations are referenced. In the figure, the variable name is Num and it consumes two bytes of memory at memory addresses 1001 and 1002. The content of this variable is 18. That is the *L-value* of Num is 1001 and the *R-value* is 18.



Operators: The tokens or symbols that trigger computer to carry out operations. The data on which an operation are called **operands**. An operand may be either a constant or a variable. Classified into three –unary, binary and ternary.

Arithmetic operators: Used to perform basic arithmetic operations such as addition, subtraction, multiplication and division.

Modulus operator (%): Also called as mod operator. It gives the remainder value during arithmetic division.

Relational operators: Used for comparing numeric data. These are binary operators. The result of any relational operation will be either True or False.

Expression: Composed of operators and operands. Classified into arithmetic expressions, relational expressions and logical expressions.

Arithmetic expression: An expression in which only arithmetic operators are used. Further classified into integer expressions and floating point (real) expressions.

Integer expression: An arithmetic expression that contains only integer operands and produces an integer result after performing all the operations given in the expression.

Floating point or real expression: An arithmetic expression that contains only floating point data and returns a floating point result after performing all the operations given in the expression.

Relational expression: An expression that contains relational operators. It produces Boolean type results like True (1) or False (0).

Logical expressions: An expression that uses logical operators to combine two or more relational expressions. It produces either True or False as the result.

Computer Science - XI

Type conversion: Conversion of the data type of an operand into another type. Done in two ways: implicitly and explicitly.



Type promotion: It is the implicit type conversion is performed by C++ compiler internally. C++ converts the lower sized operands to the data type of highest sized operand. The conversion is always from lower type to higher and hence the name type promotion.

Type casting: It is the explicit type conversion and is done by the programmer by specifying the data type within parentheses to the left of the operand.

Variable declaration: `data_type <variable1>,<variable2>, <variable3>,...`;

Variable initialisation: Supplying value to a variable at the time of its declaration. Syntax: `data_type variable = value;`

const access modifier: The keyword `const` is used to create symbolic constants whose value can never be changed during execution. Eg: `const float pi=3.14;` the value of `pi` remains constant (unaltered) throughout the execution of the program.

C++ Statements	Operator	Requirements	Example
Input Statement	>>	cin and a variable	<code>cin>> a;</code>
Output Statement	<<	cout and variable or constant or expression	<code>cout << a;</code> <code>cout << "hello";</code> <code>cout << 25;</code> <code>cout << a + 2;</code>
Assignment Statement	=	A variable and data (constant / variable / expression)	<code>N = 5;</code> <code>N = a;</code> <code>Num = a + 5;</code>

Pre-processors: These are the compiler directive statements which give instruction to the compiler to process the information provided before actual compilation starts. These lines always start with a # (hash) symbol.

Header files: Files available along with compiler and they are kept in the standard library. The header files contain the information about functions, objects and predefined derived data types.

using namespace statement: It tells the compiler about a namespace where it should search for the elements used in the program. In C++, `std` is an abbreviation of 'standard' and it is the standard namespace in which `cout`, `cin` and a lot of other things are defined. So, when we want to use them in a program, we need to follow the format `std::cout` and `std::cin`. This kind of explicit referencing can be avoided with the statement `using namespace std;` in the program.

main() function: An essential function in every C++ program. The execution starts at `main()` and ends within `main()`.