COMPUTER SCIENCE

11

ASSIGNMENT – 4

INTRODUCTION TO C++

J. KAVITHA, B.Sc, B.Ed, M.C.A, M.Phil.,
Computer Instructor Gr - I
GHSS, S.S.KULAM
Coimbatore – 641107.

INTRODUCTION TO C++

Introduction:

- C++ is one of the most popular programming language which supports both procedural and Object Oriented Programming paradigms. Thus, C++ is called as a **hybrid language.**
- Bjarne Stroustrup named his new language as "C with Classes". The name C++ was coined by Rick Mascitti where ++ is the C language increment operator.

Benefits of learning C++:

- C++ is a highly portable language and is often the language of choice for multi-device, multi-platform app development.
- C++ is an object-oriented programming language and includes **classes**, **inheritance**, **polymorphism**, **data abstraction** and **encapsulation**.
- C++ has a rich function library.
- C++ allows exception handling, inheritance and function overloading which are not possible in C.
- C++ is a powerful, efficient and fast language. It finds a wide range of applications – from GUI applications to 3D graphics for games to real-time mathematical simulations.

Character set:

- Character set is a set of characters which are used to write a C++ program. A character represents any alphabet, number or any other symbol (special characters) mostly available in the keyboard.

Lexical Units (Tokens):

The smallest individual unit in a program is known as a "**Token**" or "**Lexical unit**."
C++ has the following tokens: Keywords, Identifiers, Constants, Operators, Punctuators.

- **Keywords:** Keywords are the reserved words which convey specific meaning to the C++ compiler. Keywords are the essential elements to construct C++ programs.
  EX: int , void , break , do , if etc..
- **Identifiers:** Identifiers are the user-defined names given to different parts of the C++ program. These are the fundamental building blocks of a program.
  EX: name, mark, num etc..
- **Literals (Constants):** Literals are data items whose values do not change during the execution of a program. Therefore Literals are called as Constants.
- **Operators:** The symbols which are used to do some mathematical or logical operations are called as **"Operators".** The data items or values that the oper ators act upon are called as **"Operands". In C++, The operators are classified on the basis of the number of operands.**
  (i) **Unary Operators** - Require only one operand
  (ii) **Binary Operators** - Require two operands
  (iii) **Ternary Operators** - Require three operands
- **Punctuators:** Punctuators are symbols, which are used as delimiters, while constructing a C++ program. They are also called as **"Separators".**

C++ Binary Operators are classified as:

- **Arithmetic Operators:** Arithmetic operators to perform simple arithmetic operations like addition, subtraction, multiplication, division etc.,

Operator	Operation	Example
+	Addition	10 + 5 = 15
-	Subtraction	10 – 5 = 5
*	Multiplication	10 * 5 = 50
/	Division	10 / 5 = 2 (Quotient of the division)
%	Modulus (To find the reminder of a division)	10 % 3 = 1 (Remainder of the division)

- **Relational Operators:** Relational operators are used to determine the relationship between its operands, the result will be a Boolean value i.e **1** or **0** to represents **True** or **False** respectively.

Operator	Operation	Example
>	Greater than	a > b
<	Less than	a < b
>=	Greater than or equal to	a >= b
<=	Less than or equal to	a <= b
==	Equal to	a == b
!=	Not equal	a != b

- **Logical operators :** A logical operator is used to evaluate logical and relational expressions. C++ provides three logical operators.

Operator	Operation	Description
&&	**AND**	It returns 1, if both expression are true, otherwise it returns 0.
||	**OR**	It returns 1, if either one of the expression is true. It returns 0, if both the expressions are false.
!	**NOT**	It simply negates or inverts the truth value. i.e., if an operand / expression is 1 (true) then this operator returns 0 (false) and vice versa

- **Assignment Operator:** Assignment operator is used to assign a value to a variable which is on the left hand side of an assignment statement. **Let a = 10,**

Operator	Name of Operator	Example
+=	Addition Assignment	c = a += 5; (ie, a = a + 5) c = 15
-=	Subtraction Assignment	c = a -= 5; (ie. a = a – 5) c = 5
*=	Multiplication Assignment	c = a *= 5; (ie. a = a * 5) c = 50
/=	Division Assignment	c = a /= 5; (ie. a = a / 5) c = 2
%=	Modulus Assignment	c = a %= 5; (ie. a = a % 5) c = 0

- **Conditional Operator:** In C++, there is only one conditional operator. **?:** is a conditional Operator which is also known as Ternary Operator. This operator is used as an alternate to if … else control statement.

I/O Operators:

- **Input operator:** C++ provides the operator >> to get input. It extracts the value through the keyboard and assigns it to the variable on its right; hence, it is called as **"Stream extraction"** or **"get from"** operator.
- **Output Operator**: C++ provides << operator to perform output operation. The operator << is called the **"Stream insertion"** or **"put to"** operator. It is used to send the strings or values of the variables on its right to the object on its left. << is a binary operator.

Execution of C++ program:

- **Creating Source code:** Creating includes typing and editing the valid C++ code as per the rules followed by the C++ Compiler.
- **Saving source code with extension .cpp:** After typing, the source code should be saved with the extension .cpp
- **Compilation:** In compilation, compiler links the library files with the source code and verifies each and every line of code. If any mistake or error is found, it will throw error message. If there are no errors, it translates the source code into machine readable object file with an extension .obj
- **Execution:** In this stage, the object file becomes an executable file with extension .exe. Once the program becomes an executable file, the program has an independent existence. This means, you can run your application without the help of any compiler or IDE.

Types of Errors:

- **Syntax Error:** Syntax is a set of grammatical rules to construct a program. Syntax errors occur when grammatical rules of C++ are violated.
- **Semantic Error:** A Program has not produced expected result even though the program is grammatically correct. It may be happened by wrong use of variable / operator / order of execution etc. This means, program is grammatically correct, but it contains some logical error. So, Semantic error is also called as **"Logic Error**.
- **Run-time error:** A run time error occurs during the execution of a program. It occurs because of some illegal operation that takes place.

C++ Data types:

- In C++, the data types are classified as three main categories:
  (1) Fundamental data types
  (2) User-defined data types and
  (3) Derived data types.

Fundamental data types:

- Fundamental (atomic) data types are predefined data types available with C++. There are five fundamental data types in C++: **char, int, float, double** and **void**. Actually, these are the keywords for defining the data types.

Data type modifiers:

- Modifiers can be used to modify (expand or reduce) the memory allocation of any fundamental data type. They are also called as Qualifiers.
- There are four modifiers used in C++. They are:
  (1) signed (2) unsigned (3) long (4) short

Variables:

- Variables are user-defined names assigned to specific memory locations in which the values are stored. Variables are also identifiers; and hence, the rules for naming the identifiers should be followed while naming a variable.

Expression:

- An expression is a combination of operators, constants and variables arranged as per the rules of C++. It may also include function calls which return values.

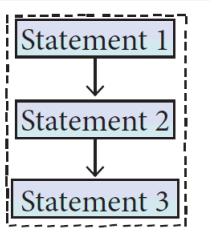
Type Conversion:

- The process of converting one fundamental data type into another is called as "Type Conversion". C++ provides two types of conversions.
  (1) Implicit type conversion (2) Explicit type conversion.

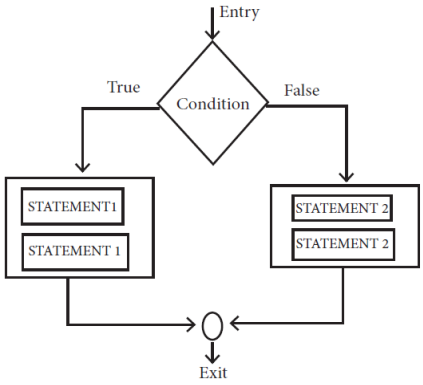
Flow of Control:

Control statements are statements that alter the sequence of flow of instructions. In a program, statements may be executed sequentially, selectively or iteratively.

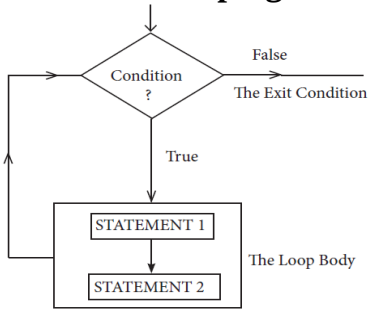
- **Sequence statement:** The **sequential statement** are the statements, that are executed one after another only once from top to bottom. They always end with a semicolon (;).



- **Selection statement:** The statement (s) executed depend upon a condition. If a condition is true, a true block (a set of statements) is executed otherwise a false block is executed. This statement is also called **decision statement.**



- **Iteration statement:** The **iteration statement** is a set of statement that are repetitively executed based upon a conditions. If a condition evaluates to true, the true block is executed again and again. As soon as the condition becomes false, the repetition stops. This is also known as **looping statement** or iteration statement.



Functions:

- A large program can be split into small sub-programs (blocks) called as functions where each sub-program can perform some specific functionality.
- Functions reduce the size and complexity of a program, makes it easier to understand, test, and check for errors.
- The functions which are available by default are known as **"Built-in"** functions and user can create their own functions known as **"User-defined"** functions.

Scope Rules of Variables:

- Scope refers to the accessibility of a variable. There are four types of scopes in C++. They are: **Local scope, Function scope, File scope** and **Class scope**.
- A scope is a region or life of the variable and broadly speaking there are four places, where variables can be declared,
  - Inside a block which is called local variables.
  - Inside a function is called function variables.
  - Outside of all functions which is called global variables.
  - Inside a class is called class variable or data members.

Arrays and Structures:

- An array is a collection of variables of the same type that are referenced by a common name.
- There are different types of arrays used in C++. They are:
  - One-dimensional arrays
  - Two-dimensional arrays
  - Multi-dimensional arrays
- Structure is a user-defined which has the combination of data items with different data types. This allows to group variables of mixed data types together into a single unit.

EDUCATION is the

MOST POWERFUL WEAPON

..... which you can use to .....

CHANGE THE WORLD.

J. Kavitha B.SC.,B.Ed.,M.C.A.,M.Phil

Computer Instructor Gr-1

GHSS, Sarkarsamakulam

Coimbatore - 641107.