Practical guide for Object Oriented Programming (C++)-01

SWT12032

Lab Sheet-01

2017-07-31

**Introduction**

A C++ program is a collection of commands, which tell the computer to do "something". This collection of commands is usually called C++ source code, source code or just code. Commands are either "functions" or "keywords". Keywords are a basic building block of the language.

C++ is a general purpose, case-sensitive, free-form programming language that supports object-oriented, procedural and generic programming.

C++ is a superset of C, and that virtually any legal C program is a legal C++ program.

C++ is being highly used to write device drivers and other software that rely on direct manipulation of hardware under real-time constraints.

C++ is widely used for teaching and research because it is clean enough for successful teaching of basic concepts.

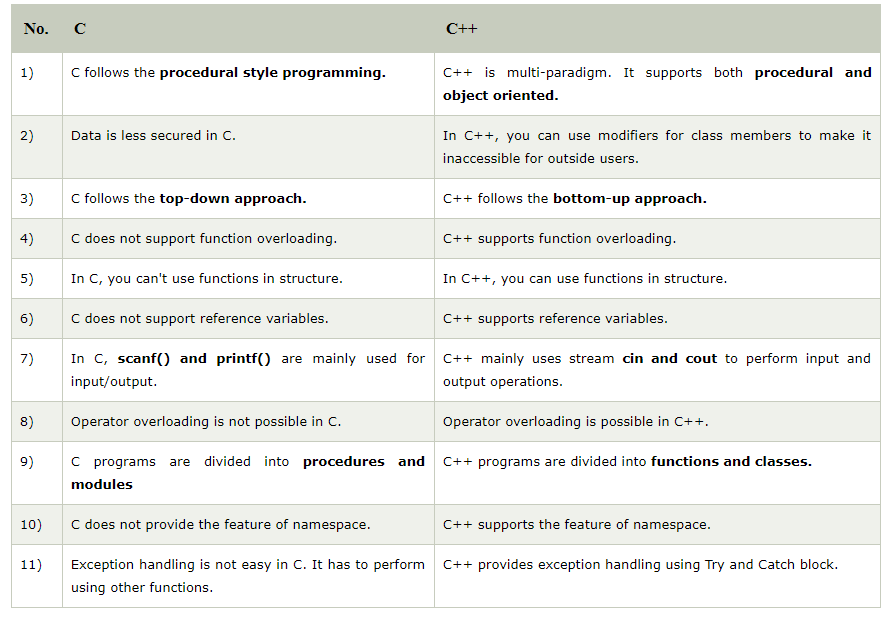
Anyone who has used either an Apple Macintosh or a PC running Windows has indirectly used C++ because the primary user interfaces of these systems are written in C++.

**Object-Oriented Programming (OOPs)**

C++ supports the object-oriented programming, the four major pillar of object oriented programming used in C++ are:

* Inheritance
* Polymorphism
* Encapsulation
* Abstraction

**C vs C++**



**Basic Syntax**

// my first program in C++

#include <iostream>

using namespace std;

int main() {

cout << "Hello World"; // prints Hello World

return 0;

}

*Line 1: // my first program in C++*

Two slash signs indicate that the rest of the line is a comment inserted by the programmer but which has no effect on the behavior of the program. Programmers use them to include short explanations or observations concerning the code or program.

*Line 2: #include <iostream>*

Lines beginning with a hash sign (#) are directives read and interpreted by what is known as the preprocessor. They are special lines interpreted before the compilation of the program itself begins. In this case, the directive #include <iostream>, instructs the preprocessor to include a section of standard C++ code, known as header iostream, that allows to perform standard input and output operations, such as writing the output of this program (Hello World) to the screen.

*Line 3: A blank line.*

Blank lines have no effect on a program. They simply improve readability of the code.

*Line 4: using namespace std;*

This line tells the compiler to use a group of functions that are part of the standard library (std). By including this line at the top of a file, you allow the program to use functions such as cout. The semicolon is part of the syntax of C++. It tells the compiler that you're at the end of a command.

*Line 4: int main ()*

This line initiates the declaration of a function. Essentially, a function is a group of code statements which are given a name: in this case, this gives the name "main" to the group of code statements that follow. Their definition is introduced with a succession of a type (int), a name (main) and a pair of parentheses (()), optionally including parameters.

The function named main is a special function in all C++ programs; it is the function called when the program is run. The execution of all C++ programs begins with the main function, regardless of where the function is actually located within the code.

*Lines 5 and 8: {and}*

The open brace ({) at line 5 indicates the beginning of main's function definition, and the closing brace (}) at line 8, indicates its end. Everything between these braces is the function's body that defines what happens when main is called. All functions use braces to indicate the beginning and end of their definitions.

*Line 6: cout << "Hello World!";*

This statement has three parts: First, cout, which identifies the **c**haracter **out**put device (usually, this is the computer screen). Second, the insertion operator (<<), which indicates that what follows is inserted into std::cout. Finally, a sentence within quotes ("Hello world!"), is the content inserted into the standard output.

**CIN>>**

The C++ standard libraries provide an extensive set of input/output capabilities.

The predefined object **cin** is an instance of **istream** class. The **cin** object is said to be attached to the standard input device, which usually is the keyboard.

The **cin** is used in conjunction with the stream extraction operator, which is written as **>>** which are two greater than signs as shown in the following example.

#include <iostream>

using namespace std;

int main( ) {

char name[50];

cout << "Please enter your name: ";

cin >> name;

cout << "Your name is: " << name << endl;

}