**Technical Writing 101: Introduction to C++**

Evelyne Adera

Lux Dev Academy

Technical Writing

Due 24/06/2023

**Abstract**

This is a basic and concise introduction to C++ programming language. It provides a comparative analysis and fundamentals of C++ in relation to other programming languages such as Python, PHP, Perl, SQL, JavaScript, and Ruby.

This paper outlines the basics of C++ as a programming language. The emphasis is on the advantages, disadvantages, constraints, and importance of the language. The evolution of C++ is traced from C with Classes to the current ANSI and ISO standards work and the explosion of use, interest, commercial activities, and the environment. C++ is rapidly displacing FORTRAN as the basic language used by physical scientists and engineers.

C++ was designed to provide efficient handling of algebras and special functions that are needed for loop calculations in theoretical quantum field theory.

**Technical Writing 101: Introduction to C++**

**Introduction to C++ Programming Language**

A programming language is a system of notation used to write computer programs. In other words, it is a computer language engineered to create a standard form of commands that can be interpreted into codes understood only by a machine. A programming language tells the computer what to do. Programming languages are used to write all computer programs and computer software.

There are three types of programming languages:

Machine language- this is the binary system that the computer understands

Assembly language- this language is a basic set of symbols to represent the 0s and 1s of a machine code

Structured programming language- this language is more sophisticated than the assembly language.

Examples of programming languages that tell the computer what to do include Python, Java, JavaScript, Ruby, C, C++, and C#.

**What is C++?**

C++ is a cross-platform language that can be used to create high-performance applications. It was developed by Bjarne Stroustrup, as an extension to the C programming language. C++ was updated 4 times in 2011, 2014, 2017, and 2020 to C++11, C++14, C++17, C++20.

C++ is one of the most popular programming languages used in the world. It can be found in today’s operating systems, Graphical User Interfaces, and embedded systems. It is an object-oriented programming language that gives a clear structure to programs and allows code to be reused. C++ is easy to learn and is close to C#, C, and Java.

In C++, there are different types of variables. A variable is a container used for storing data values, for example:

string- stores text and is surrounded by double quotes for example “Hello World”

char- stores single characters and is surrounded by single quotes, for example ‘a’

int- stores integers without decimal numbers

Double- stores floating point numbers with decimals

**C++ Syntax**

Example

*#include<iostream>*

*using namespace std;*

*Int main() {*

*Count << “Hello World!”;*

*Return 0;*

*}*

Line 1: i.e. #include <iostream> is a header file library that lets us work with input and objects. Header files add functionality to C++ programs.

Line 2: using namespace std means that we can use names for objects and variables from the standard library.

Line 3: A blank line is used to make the code more readable

Line 4: int main () - this is called a function and it executes any code inside its curly brackets

Line 5: count- is an object used together with the operation (<<) to output/print text

Line 6: return 0 ends the main function

Line 7 closing curly brackets to end the main function.

Every C++ statement ends with a semicolon; and the body of int main () could also be written as int main () { cout << "Hello World! "; return 0; } **NB:** The compiler ignores white spaces. However, multiple lines make the code more readable.

Conclusion

C++ was specifically designed with an orientation towards large systems and resource-constrained software. The best advantage of C++ is that it is super scalable and allows developers to have a lot of control over how their applications use up resources.

C++ is the most preferred choice in terms of speed. The compilation and execution time of C++ is much faster and unlike other programming languages where no compilation is required, every C++ code has to be first compiled to a low-level language and then executed.

C++ supports pointers which are often not available in other programming languages. C++ is also closer to hardware than most general-purpose programming languages. This makes it very useful in those areas when hardware and software are closely coupled together, and low-level support is needed at the software level.

**Real-World Application of C++**

C++ can be found in almost all applications today, here are some areas where C++ is popularly used:

Operating System- it is an ideal choice for developing operating systems.

Games- game developing companies use it as their primary choice to develop gaming systems.

GUI-Based Applications- most of the applications from Adobe such as Photoshop, illustrator are developed using C++.

Bank Applications- Infosys Finacle is a popular banking application developed using C++.

References

Bauer, C., Frink, A., & Kreckel, R. (2002). Introduction to the GiNaC framework for symbolic computation within the C++ programming language. *Journal of Symbolic Computation*, *33*(1), 1-12.

Syntax, J. S. O. N. (2012). // w3schools. com. *URL: https://www. w3schools. com/js/js\_json\_syntax. asp*.