Artículo en Inglés

C++ programming language: How it became the invisible foundation for everything, and what's next

Universidad Fidélitas
Introducción a la Informática

6/28/2023
Jose Daniel Aguilar Aguilar
Lic. Mariela Ugalde Diaz

jaguilar30212@ufide.ac.cr

Introduction

C++ is a powerful programming language that combines the features of both high-level and low-level programming. It was created as an extension of the C language, adding object-oriented programming capabilities and providing a wide range of tools and libraries for software development. C++ allows developers to write efficient and portable code, making it suitable for a variety of applications, including system software, game development, and scientific computing. It supports features such as classes, templates, and inheritance, enabling the creation of reusable code and the implementation of complex data structures and algorithms. It remains as a popular choice for programmers seeking control over memory management and direct hardware access, while still benefiting from other higher level programming.

C++ was created by Bjarne Stroustrup in the early 1980s. Stroustrup, a Danish computer scientist, developed C++ as an extension of the C programming language to add object-oriented programming features. The first version of C++ was released in 1985.

Article

It is said by Owen Hughes [1] that "C++'s origins date back to 1979, when Bjarne Stroustrup, the programming language's creator, first began work on the language that was then known as 'C with Classes'."C++ is an enhanced version of the C programming language, primarily aimed at improving upon its capabilities by incorporating object-oriented programming features.

Stroustrup's goal was to develop a multi-computer system with a flexible communication system, capable of utilizing shared-memory or a network. He recognized

the need for low-level, hardware-oriented code like memory managers and device drivers to facilitate separate software components running on distinct computers. Stroustrup found existing languages inadequate for his vision, prompting him to create his own solution. "C with Classes" emerged as a combination of C's hardware proximity and Simula's efficient class implementation for abstraction and code organization. As the language evolved, Stroustrup realized he no longer had sole control over it, as the project expanded to involve numerous colleagues during standardization efforts.

Owen Hughes [1] continues explaining that “1998 marked the year C++ was formally standardized and became a solid workhorse”.At that time, C++ had become one of the prevailing programming languages globally, and it continues to hold that position up until now. Stroustrup's summary follows: "When faced with a problem that demands both efficient hardware utilization and the management of significant complexity, C++ is an obvious choice. However, if you only have one of these requirements, either a low-level language optimized for efficiency or a high-level language with less efficiency would suffice."

In [1] it is also important to remark that “As of September 2020, C++ is the fourth most popular programming language globally behind C, Java and Python, and – according to the latest TIOBE index”. This is mostly due to its powerful performance capabilities, flexibility, and extensive support from the developer community. Its efficiency, cross-platform compatibility, and legacy codebase make it a popular choice for building high-performance applications in industries.

Conclusion

C++ is widely recognized and extensively used due to various factors. Firstly, its focus on efficiency and performance enables developers to create code that executes rapidly and optimizes hardware resources, making it ideal for developing applications that require high performance, such as game engines, embedded systems, and scientific computing. Secondly, C++ offers great flexibility by supporting multiple programming paradigms, including procedural, object-oriented, and generic programming. This adaptability empowers developers to choose the most suitable approach for their projects, fostering the creation of scalable and maintainable codebases. Additionally, C++ benefits from a vibrant community of developers, extensive libraries, and widespread adoption in various industries, further establishing it as a preferred language. Its compatibility across different platforms, seamless integration with existing C code, and vast codebase also contribute to its significance and extensive usage.

Referencias

[1] Hughes, O. et al. (2021) C++ programming language: How it became the invisible foundation for everything, and what’s next, TechRepublic. Available at: https://www.techrepublic.com/article/c-programming-language-how-it-became-the-invisible-foundation-for-everything-and-whats-next/