The Growth of C++

**Introduction:**

* Inspired by C, Bjarne Stroustrup initially developed C++ as an extension to the C language. Although, over time, [C++](https://www.simplilearn.com/c-plus-plus-programming-for-beginners-article) has evolved into a multi-model, general-purpose programming language. It is mostly used in Microsoft products and desktop applications. Over the last decade, C++ has grown into one of the most well-known and widely used programming languages.

**C++ expression evaluation:**

* In the C++ programming language, an expression is evaluated based on the operator precedence and associativity. When there are multiple operators in an expression, they are evaluated according to their precedence and associativity. The operator with higher precedence is evaluated first and the operator with the least precedence is evaluated last.

To understand expression evaluation in c, let us consider the following simple example expression.

Example

10 + 4 \* 3 / 2

In the above expression, there are three operators +, \* and /. Among these three operators, both multiplication and division have the same higher precedence and addition has lower precedence. So, according to the operator precedence both multiplication and division are evaluated first and then the addition is evaluated. As multiplication and division have the same precedence they are evaluated based on the associativity. Here, the associativity of multiplication and division is left to right. So, multiplication is performed first, then division and finally addition. So, the above expression is evaluated in the order of \* / and +. It is evaluated as follows...

4 \* 3 ====> 12

12 / 2 ===> 6

10 + 6 ===> 16

The expression is evaluated to 16.

**Operator Precedance:**

In any programming language, every operator has provided a preference that is used at the time of expression evaluation. In C++, the following list provides the operators' preference from higher to lower.

Pre-increment (or) pre- decrement

Parenthesis, shifting operators, size of

As trick (\*), multiplication and division

Addition, subtraction

Relational operators

Assignment operators

Post increment/post decrement

**C++ Growing Faster Than Other, Older Programming Languages**

The C++ programming language has been around for quite some time, which means it’s built up quite the mountain of legacy code. But the language’s continuing ubiquity isn’t just a matter of developers trying to maintain longtime apps and platforms; as the latest update to the TIOBE Index makes clear, C++ is actively growing.

Specifically, C++ has risen 1.48 percent over the past 12 months, reaching (in TIOBE’s estimation) a 7.11 percent market-share. That might not seem like stratospheric growth, but it beats out every single over “big” language on the list, including the ever-popular Python (which grew 0.59 percent over the past year). While C++ is currently in fourth place on TIOBE’s overall list, C is in first (with 15.95 percent), having overthrown Java (with 13.48 percent).

What’s behind the strong performance of C++? That’s an excellent question, but the note accompanying the rankings suggests that some recent updates have something to do with it: “I think that the new C++20 standard might be one of the main causes for this. Especially because of the new modules feature that is going to replace the dreadful include mechanism. C++ beats other languages with a positive trend such as R (+1.33%) and C# (+1.18%).”

In order to generate its rankings, TIOBE utilizes data from a variety of aggregators and search engines, including Google, Wikipedia, YouTube, and Amazon. For a language to rank, it must be Turing complete, have its own Wikipedia entry, and earn more than 5,000 hits for +”<language> programming” on Google. That methodology has attracted its share of critics over the years, who argue that the rankings are more a measure of these languages’ “buzz” than actual usage. In any case, “big” languages such as Python and C++ tend to stick to the upper rankings, while smaller languages often experience big ranking swings up and down.

C++ began its existence as “C with Classes,” but it’s managed to drift well away from C over the past few decades.

**Let’s look at the top programming languages:**

1. Python
2. Java
3. Javascript
4. C#
5. PHP
6. C++
7. C
8. R
9. Swift
10. Objective C

## 

**Analysis:**

C++ declined in popularity after 2003 as other programming languages like Java and Python gained momentum.

After years of negative growth, C++ is now the fastest-growing programming language in terms of popularity.

After falling to its all-time-low score in 2017, C++ ranked fourth among 24 other programming languages in September, according to the TIOBE Programming Community Index.

The community index rated C++ at 7.11%, which is lower than the ratings it gave C, Java or Python. But TIOBE’s change in rating for C++ grew the most among the top 5 programming languages.

The index ratings are based on the number of skilled engineers world-wide, courses and third party vendors. The programme's hits on search engines such as Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube and Baidu are used to calculate the ratings.

C, Java and Python continue to outperform C++ this month, although Java's popularity dropped the most this month as compared with last year, the index noted.

C++ became popular in 2003, but declined later as other languages gained momentum. It started catching up because of the new C++ standard that replaced the older coding mechanism, the TIOBE said.

**Some of the queries clarified by technological person about CPP**

[Bjarne Stroustrup designed the C++ language](http://www.infoworld.com/d/application-development/stroustrup-reveals-whats-new-in-c-11-187051) in 1979, and the general-purpose language for systems programming has become a mainstay for developers everywhere, despite competition from Java, JavaScript, Python, Go, and [Apple's newly unveiled Swift](http://www.infoworld.com/d/application-development/first-look-apples-swift-simple-first-245009).

Now a technologist at Morgan Stanley and a professor at both Columbia University and Texas A&M University, Stroustrup spoke with InfoWorld Editor at Large Paul Krill about C++'s role today and about other happenings in software development, including Google's Go and [Apple's Swift](http://www.infoworld.com/slideshow/155797/10-features-apple-stole-the-swift-programming-language-243934) languages.

**InfoWorld:** *Where do you see the role of C++ today, when you have popular scripting languages like Python and JavaScript along with languages like Java and even Google's Go? How does C++ manage to survive, thrive, and grow in such a diverse landscape with all these different languages?*

#### [ [What is GitOps? Extending devops to Kubernetes and beyond](https://www.infoworld.com/article/3566555/what-is-gitops-extending-devops-to-kubernetes-and-beyond.html) ]

**Stroustrup:** That's a good question. People have been predicting its demise quite enthusiastically for more than 20 years, but it's still growing. Basically, nothing that can handle complexity runs as fast as C++. If you go to some embedded areas, if you go to image processing, if you go to some telecom applications, if you go to some financial applications, C++ rules. You don't see it much if you're into looking at apps and such, that's not where you find it. It's things like Google, Amazon, search engines, where you really need performance, that's where it is.

***InfoWorld:****Some of these new scripting languages are intended for easy consumption by developers. Would you say C++ requires more attention than that?*

**[**[**Learn how leading CIOs are reinventing IT. Download CIO's new Think Tank report today!**](https://www.cio.com/resources/219229/cio-think-tank-roadmap-report-reinventing-it)**]**

**Stroustrup:** Oh, definitely. C++ is designed for fairly hardcore applications, and it's always been used together with some scripting language or other. When I started, I used C++ for anything that required a real programming language and real performance. Then I used the Unix shell as my scripting language. That was how it [was done], and that's also the way things are done in most of the cases today. [C++ is for] high performance, high reliability, small footprint, low energy consumption, all of these good things. I'm not saying hobbyists; I'm not saying quick apps. That's not our domain.

***Why is C++ still in use?***

* 1. When we are programming with the advanced programming languages, our main focus is on the implementation of functionalities. We follow the guidelines of best practice to avoid small errors, but it is not quite enough to attain better performance. To get better performance, you need precise profiling analysis to find out which type of codes does well and how to rewrite them more efficiently.
  2. C++ allows low-level manipulation of data. Embedded systems and compilers are created with the help of C++. You can write close-to-hardware code that is capable of running as fast as the CPU. It works in favour of applications where complex calculations might be required, such as performing scientific calculations.

**Python vs C++**

As you all know that many computing languages are highly compared to Python as it is a high level , mostly used language in the present days. So let’s see what is the main difference between these two.

Python and C++ are extremely different from each other, and most of the differences don't necessarily prove one more advantageous than the other. That being said, for most uses, it’s easy to choose a side and make a good case for or against particular language and implementation features. Python and C++, when associated with each other, can lead to a lot of opinions. Each developer will have their own opinion, so we have tried to sort a few of them to give you a clear outlook.

When it comes to their use cases, Python is the best language for machine learning and data analysis, whereas C++ is the best option for game development, mobile app development and large systems.

One important aspect of Python vs C++ is memory management. Python does not let you manage memory directly, instead, it provides automatic memory management, referred to as a garbage collector. C++ does not have that feature, and all memory management takes place manually.

Python also helps to create games, however, the main concern with using Python for games is the performance issue. It is too slow to develop fast but intense parts of the game. This speed issue does not mean you can’t use this language in combination with C++. For example, Python often produces the artificial intelligence feature in games.

Developers often merge C++ modules with Python to improve Python and compensate for its slow performance. However, coding with C++ is more complex, so it requires more contemplation and research. Utilizing C++ and Python together for your business application development means combining simplicity and speed.

Comparing C++ and Python leads to one conclusion: Python is better for beginners because of its easy-to-read code and simple syntax. Additionally, Python is a good choice for website development (backend), while C++ is a good choice for game development.

[Click Here](https://evontech.com/component/easyblog/is-c-development-still-alive.html?Itemid=159) ->**To know exactly about what is the future of C++ Programing Language**

**The Streams and areas where C++ Can be used**

Real-World Applications of C++

1) Games

2) GUI Based Applications

1. Adobe Systems
2. Win Amp Media Player

3) Database Software

MYSQL Server

4) Operating Systems

1. Apple OS
2. Microsoft Windows OS

5) Browsers Mozilla Firefox

1. Thunderbird
2. Google Applications

6) Advanced Computation and Graphics

Alias System

7) Banking Applications

Infosys Finacle

8) Cloud/Distributed System

Bloomberg

9) Compilers

10) Embedded Systems

**Final say**

All things considered, we like how C++ gives us the tools to think and code at a high level of abstraction, yet lets us get close to the machine when we need to. This is a unique combination of functionality that no other programming language can match, and it’s the major reason why C++ is still so popular in the application development world.

Evon Technologies has been one of the C++ development companies in India since 2006. Our C++ development services have attracted a long list of clientele in the past and continue to do so. Those services include native C++ application development, cross-platform C++ application development, custom C++ application development, upgrading/porting the existing application and C++ application support and maintenance. Our domains of expertise include, but are not limited to, technologies around Bluetooth, motion sensing, audio/video conferencing, content sharing, social networking etc.