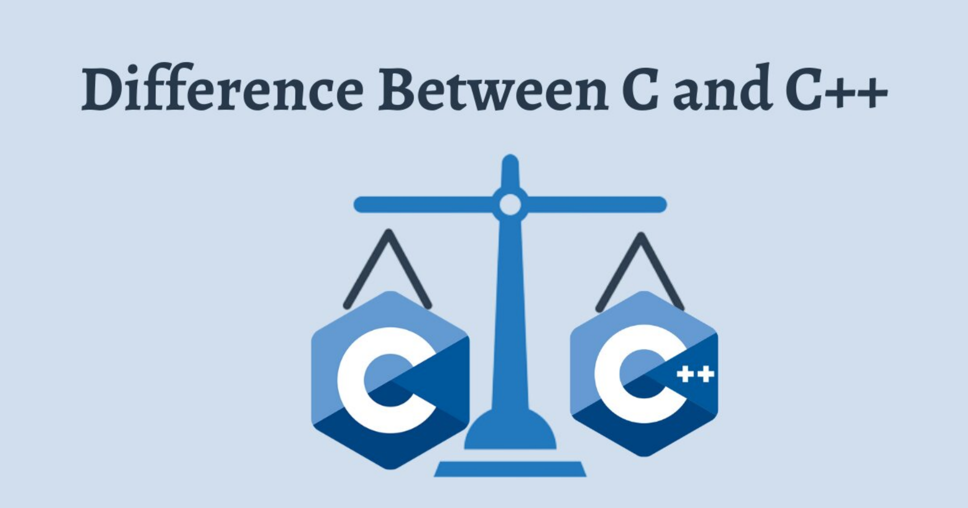
Difference Between C and C++

Atul Harsha
Senior Manager Content

Updated on Dec 22, 2023 13:00 IST

C focuses on procedural programming. It's about following specific steps or instructions. This involves using functions and structured techniques. Essentially, C revolves around functions and the order of program operations.

C++, in contrast, expands on C. It introduces object-oriented programming. This method uses classes and objects. These elements group related functions and data. This approach simplifies complex software design, understanding, and maintenance. The main difference between C and C++ is their approach. C is driven by functions and their execution sequence. C++, however, is based on OOP. It focuses on classes, objects and their interactions. This makes it more suited for handling complex software systems.



The most important point of difference between **C** and **C++** language is that C is procedural-oriented programming, and C++ is based on **object-oriented**

programming methods.

Table of Content

Key Difference Between C and C++ Language

- C++ supports classes, which are a type of object that can be used to group data and methods. C does not support classes.
- C++ supports inheritance, which allows one class to inherit the properties of another class. C does not support inheritance.
- C++ supports polymorphism, which allows objects of different types to be treated similarly. C does not support polymorphism.
- C++ is a compiled language, while C is an interpreted language. This means that C++ code is converted into machine code before it is executed, while C code is interpreted line by line.
- C++ is a more complex language than C. This is because C++ has more features, such as

classes, inheritance, and polymorphism.

Let’s understand this in detail.

C	C++
C was developed in 1972 by Dennis Ritchie at Bell Laboratories.	C++ was developed by Bjarne Stroustrup of Bell Laboratories in the early 1980s.
It is a function-driven language.	It is an object-driven language.
C is a Procedural Oriented language. It does not support object-oriented programming (OOP) features such as polymorphism, encapsulation, and inheritance programming.	C++ is both a procedural and an object-oriented programming language. It supports OOP features such as polymorphism, encapsulation, and inheritance.
C is a subset of C++.	C++ is a superset of C.
C has 32 keywords.	C++ has 63 keywords.
Supports built-in data types	Supports both built-in and user-defined data types
The file extension of a program in C language is .c.	The file extension of a C++ program is .cpp.
Does not have access modifiers.	Has access modifiers.
C uses <stdio.h> header file for input/output operations.	C++ uses <iostream.h> header file for input/output operations.
No support for information hiding.	Data is secured and hidden by Encapsulation.
C focuses on the method or process rather than data.	C++ focuses on data rather than method or procedure.
No direct support for exception handling.	Supports exception handling.
C uses scanf() and printf() functions for input/output.	C++ uses cin and cout for input/output.
Does not support function and operator overloading.	Supports function and operator overloading.
The main() function can be called through other functions used in the code.	Does not allow the main() function to be called through other functions.
Does not support any reference variables.	Supports reference variables.

Similarities Between C & C++

Some of the similarities in C and C++ programming languages are:

- **Syntax:** C++ is an extended version of C, therefore both have similar syntax, compilation and code structure.
- **Keywords:** Most of C's keywords and operators are used in C++ and perform the same function.
- **Execution:** C and C++ both follow top-down execution of the code.
- **Comment:** Inline Comment in both C and C++ is marked by //.
- **Multi-Dimensional Array:** Both C and C++ support multi-dimensional arrays.
- **Dynamic size Array:** None of them support dynamic sized arrays.
- **Statement Terminator:** Both C and C++ uses semi-colon (;) for terminating a statement.
- **Preprocessor Directive:** #include is used in both C and C++ to include/import a header file.

Preparing for a C++ interview? Check out the Top C++ Interview Questions and Answers

What is C Programming Language?

C programming is a structural or procedural-oriented programming language developed by Dennis Ritchie at Bell Laboratories in 1972. In fact, C programming language was originally developed to migrate the UNIX kernel code from assembly to a higher-level language that could do the same functions with fewer lines of code.

Check out the best C Programming Courses

C is considered a middle-level **programming language** because it has the features of low-level language as well as high-level Language. It is one of the foundational languages for beginners or new programmers as it forms the base for other programming languages.

Also Read – Understanding Operators in C++

Features of C Programming

Listed below are some of the key features of C programming –

- **Procedural programming language**: C is designed to be easy to read and understand. It follows a logical structure that makes it easy to write and debug code.
- **Structured language**: C has a clear and concise syntax and follows a logical structure. For example, C uses curly braces to define blocks of code and semicolons to mark the end of statements.
- **Portable**: C programs can be adapted to run on different platforms and hardware architectures. This makes it ideal for building software that needs to be run on a variety of different systems. For example, C programs can be compiled and run on Windows, Linux, and macOS.
- **Low-level access:** C gives programmers direct access to the hardware and operating system. This makes it ideal for writing code that needs to interact with the underlying system or perform tasks that require a high degree of control. For example, C can be used to write device drivers or system utilities that need to access the hardware directly.
- **Efficient and fast:** C is Well-suited for tasks that require a lot of processing power or that need to be completed quickly. This is because C is a compiled language, which means that it is converted into machine code before it is executed, making it faster than interpreted languages.
- **ANSI standard:** C has been formally standardized by a recognized industry organization. This ensures that C programs are consistent and portable across different systems. For example, ANSI C programs can be compiled and run on any system that supports the ANSI C standard.
- **Large and active community of developers:** There are many resources available for learning and using the language. For example, there are numerous online forums, tutorials, and documentation resources available for C programmers.
- **Supports Dynamic Memory Allocation:** C supports dynamic memory allocation, which means that you can allocate memory for variables and data structures at runtime. This makes it easier to manage memory usage in large programs.
- **Supports pointers for direct memory interaction**: C supports pointers, which are variables that store the memory addresses of other variables. Pointers can be used to directly interact with memory and can be useful for optimizing code and accessing low-

level hardware.

Applications of C Programming

- **Operating System** like **UNIX**, **Microsoft Windows**, are created in C.
- **Mac OS X** kernel was originally written in C.
- Majority of **Adobe** software like **Adobe Photoshop**, **Adobe Illustrator**, **Adobe Premier** are built using C, C++.
- Build code compilers such as **MinGW**, **Dev-C**, **Clang C**.
- Web Browsers like **Mozilla**, and **Chrome**, is designed using C programming.
- Games like **Tic-Tac-Toe**, **Snake, Half-Life, Starcraft** in built using C programming. In 2004, a high graphics FPS game Doom 3 was built using C for Windows.
- Apple's **Xcode IDE** is written in C, C++, Objective-C and Objective-C++.
- The assembly language code of **UNIX** operating system's was rewritten in C in 1972.
- **Oracle** database rewrote their assembly codes to C in 1983.
- In 1985, when **Windows 1.0** was released, its kernel was primarily written in C and assembly.
- Several **embedded systems** like alarm clock, microwave, tv, remote controlled devices, etc are mostly built using C, embedded C.

Disadvantages of C Programming

- Does not support Object-Oriented Programming (OOP) features such as Polymorphism, Inheritance, and Encapsulation.
- The bugs are not detected after each line of code, instead it is detected after writing the entire program. This makes debugging in C difficult for complex programs.
- C does not support the concept of namespace, due to which two variables cannot be declared with same name.
- C does not Exception Handling which in other languages are used to detect the bugs and generate appropriate responses.
- C language has low level of abstraction. Due to this C has minimum data hiding and this affects the overall security of the language.

Learn more: Basics of C Programming Language

What is Coding: Difference Between Coding And Programming

This blog explains what is coding. It also covers the difference between coding and programming.

All About While Loop in C

In a programming language, loops are used to repeat a specific piece of code. Loops in C are classified into three types, such as For loop, While loop, and Do…while**...read more**

Top 80+ C Programming Interview Questions and Answers

Here's a list of 80+ top frequently asked C programming interview questions and answers to help you crack your next interview. Preparing for a C programming interview? Here are the**...read more**

What is C++ Programming Language?

C++ is a high-level computer programming language created in the early 1980s by Bjarne Stroustrup of Bell Laboratories. It is an extension of the traditional C language with added support for object-oriented programming and other capabilities.

C++ is close to low-level languages and is considered one of the fastest programming languages. It provides complete control over memory allocation and management. C++ is used to develop complex and high-performance applications. C++ is close to C# and Java, making it easier for programmers to switch to learning and working with these languages.

Related Read – Binary Search in C++

Features of C++

- **Object-oriented programming:** C++ supports object-oriented programming (OOP) concepts such as inheritance, polymorphism, and encapsulation, which makes it easier to develop large, complex software systems.
- **Templates:** C++ supports templates, which are reusable code blocks that can be used to create generic functions and data types.
- **Exception handling:** C++ has built-in support for exception handling, which allows you to catch and handle errors and exceptions that occur during program execution.
- **Function overloading:** C++ allows you to create multiple functions with the same name but with different parameters, a technique known as function overloading.
- **Operator overloading:** C++ allows you to overload operators, which means that you can define how operators behave when used with different data types.
- **Namespaces:** C++ supports namespaces, which are used to organize code and prevent name collisions.
- **Standard Template Library (STL):** C++ includes the Standard Template Library (STL), which is a collection of templates and algorithms that can be used to implement common data structures and algorithms.
- **Compatibility with C:** C++ is fully compatible with C, which means that you can use C libraries and code in your C++ programs. This makes it easy to incorporate existing C code into C++ projects.

Also Explore: C++ Online Courses & Certifications

Applications of C++

- **Google** used C++ to create several products like Google Earth, and the Google Chrome browser.
- **Spotify** one of the most popular audio streaming apps has its back end written in C++.
- Operating Systems like **Windows**, **Mac**, and **Linux** are built using C++.
- **Mozilla Firefox** used a subset of C++ (C++14) to build Mozilla 59.
- **Microsoft** products such as Word, Excel, and PowerPoint are written in both C and C++.

- One of the most powerful databases **MySQL** is written in both C and C++.
- At **Microsoft** code written for the – **DirectX**, **Windows API**, and **.NET** are all based on C++.
- Very popular games like **Witcher 3**, **Counter-Strike**, **Doom III Engine**, **World of Warcraft**, **King Quest**, **Football Pro** and **Invictus** are all written in C++ .
- A popular media player **Winamp** is developed using C++.
- **MongoDB** a popular open-source NoSQL database is written using C++.
- **Unreal Engine** for most graphic-intensive 2D, 3D, VR, AR, cross-platform, single-player, or multiplayer games primarily uses C++.

Python vs. C++ – What’s the Difference?

Python and C++ both are general-purpose programming languages. However, their usage and syntax differ widely. In this article, we will highlight the most prominent differences between the two languages.

Python vs Java: Which is Better to Learn in 2024?

In the world of programming, Python and Java have long been two of the most popular languages. Both have their strengths and are suited to different tasks, but recent trends**...read more**

Introduction to Inheritance in C++

In this blog, we will explore the concept of inheritance in C++ and its access. We will also discuss different types of inheritance, such as single inheritance, multiple inheritance, and**...read more**

Disadvantages of C++

- For a beginner, C++ might be a relatively difficult language to learn and start with.
- Manual memory management in C++ is a hassle, as there is no garbage collection to automatically remove unwanted data.
- If you’re used to automated memory management, managing memory allocation manually

can be a challenge.

- Not very secure because of pointers, global variables, etc.
- C++ does not support built-in code threads due to which the process is slower and complicated.
- The pointers in C++ take up more than the required memory which might not be suitable for some devices.

Latest Update from C and C++ [updated Dec,23]

As of December 2023, the C programming language remains a significant force in the world of programming. According to the TIOBE Index for December 2023, C is ranked as the second most popular programming language with a rating of 11.44%. This reflects a slight decline from its position earlier in the year, but it still underscores the enduring relevance of C in various programming fields, particularly in systems programming and embedded systems.

The C23 standard, the latest update to the C programming language released in 2023, introduces a range of new features and enhancements. These improvements are designed to make the language more robust, efficient, and compatible with modern programming needs. Here are some of the key updates:

C++23 Updates:

C++23, the latest version of the C++ programming language, is set to replace the current C++20 standard. This new update focuses on refining the language rather than making major changes. Here's a simplified overview:

- **Garbage Collection Removed**: C++23 will no longer support garbage collection, a feature that wasn't widely used.
- **Unicode for Identifiers**: It will use the Unicode Standard Annex 31 for naming identifiers, making it clearer what characters can be used and excluding inappropriate ones like emojis.
- **Multiple Attributes Allowed**: You can now use standard attributes more than once in a

list, offering more flexibility.

- **Whitespace Handling Improved**: C++23 ensures consistent handling of spaces after a backslash in the code.
- **Fixed Member Order in Classes**: The order of class members will be set based on their declaration order, avoiding confusion.
- **Simplified String Concatenation**: The update removes complex rules for combining different types of string literals.
- **New Features**: There are several new additions, such as the 'explicit object parameter', enhanced preprocessor conditionals, and a multidimensional subscript operator.
- **Syntax Improvements for Better Readability**: The ability to use labels before declarations and other syntax improvements aligns C more closely with modern programming practices, enhancing code clarity and maintainability.
- **C++ Compatibility**: By making syntax changes that improve compatibility with C++, C23 reduces the friction in using both languages together in a project. This is particularly beneficial for projects that need to integrate C and C++ codebases.

Compiler support varies, but major compilers like GCC and Clang are already incorporating many of these features□□.

C23 Proposals:

C23, the upcoming standard for the C language, is also evolving. Here are the main areas of development:

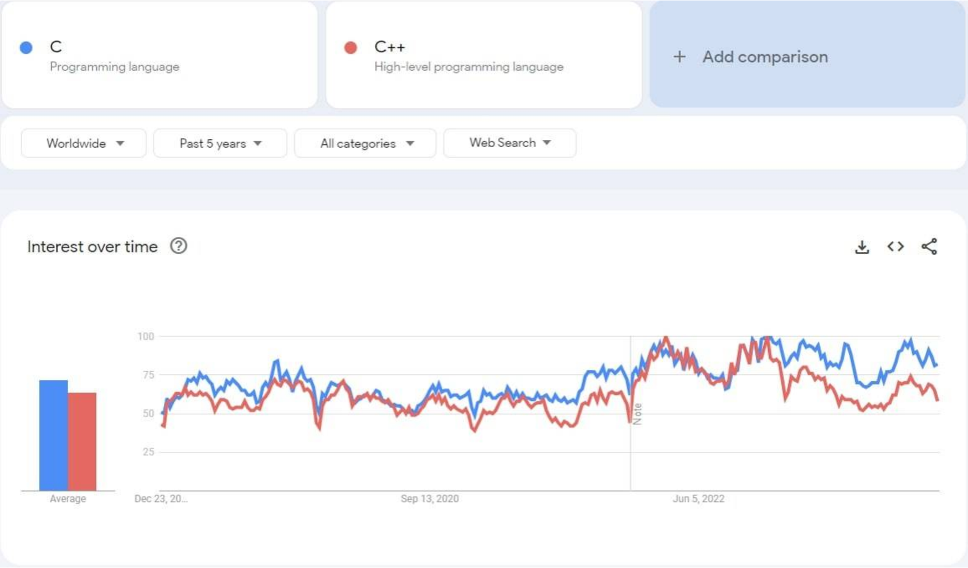
- **Clearer Integer Definitions**: The new standard aims to have a unified way of describing integer behaviour, such as overflow.
- **Unicode in Identifiers**: Like C++, C23 intends to use the Unicode Standard Annex 31 for defining valid characters in identifiers.
- **Enhanced calloc Function**: There are plans to improve calloc to handle cases where calculations may wrap around, enhancing security.
- **Safer Programming Practices**: Annex K will be revised for better safety in programming, especially in multi-threaded environments.

- **Changes to volatile Usage**: C23 is looking to deprecate certain uses of the volatile keyword, aligning it with similar changes in C++.

These updates reflect a commitment to making the languages more user-friendly, secure, and consistent with modern programming standards

Google Trends Comparison - C vs C++

Seeing the latest Google Trends of the last 5 years, the worldwide interest in both C and C++ has shown a similar trend. But recently as of date 22nd December 2023 the trend of C has surpassed C++. As per the trend, C Language is gaining more popularity showcasing an increase in popularity and demand.



● C
Programming language

● C++
High-level programming language

Worldwide
Past 5 years
All categories
Web Search

Interest over time

C vs C++ which one should I learn first in 2024 and Why?

Deciding whether to learn C or C++ first in 2024 can depend on several factors, including your career goals, the complexity of the language, and the type of projects you're interested in. Here's an updated table incorporating some recommended courses for each language:

Factors	Learning C First	Learning C++ First
Foundation	Provides a strong base in procedural programming.	Introduces object-oriented concepts early.
Complexity	Simpler and more straightforward.	More complex due to advanced features.
Career Goals	Ideal for system programming and embedded systems.	Better for application and game development.
Learning Curve	Easier for Beginners.	Steeper due to object-oriented concepts.
Transition to Other Languages	Eases understanding of other languages, including C++.	Makes transitioning to lower-level concepts in C challenging.
Community and Resources	Well-established, but more focused on foundational aspects.	Large and active, with resources for modern development.
Future Relevance	Remains relevant for foundational understanding and specific niches.	Continues to be widely relevant for diverse software development.
Best Course	C Programming For Beginners - Master the C Language, **Rating: 4.4 out of 5 (33,049 ratings) 150,317 students - Bestseller on Udemy** **Created by:** Tim Buchalka's Learn Programming Academy, Jason Fedin.	Beginning C++ Programming - From Beginner to Beyond, **Rating: 4.6 out of 5 (65,711 ratings) 283,072 students** **Created by:** Tim Buchalka's Learn Programming Academy, Dr. Frank Mitropoulos.

Learning C First: It's recommended for those who are new to programming and want to start with the basics. C's simpler syntax and procedural nature make it easier to grasp foundational programming concepts.

Learning C++ First: Suitable for individuals interested in diving directly into object-oriented programming and working on complex software applications.

Each path has its unique advantages, and the choice ultimately depends on your personal interests and career aspirations. Additionally, there are numerous high-quality online courses available to help you start your journey in either language. These courses range from beginner to advanced levels, offering a comprehensive learning experience with practical projects and hands-on coding.

Explore: Top C Programming Courses

Explore: Highly Recommended C++ Courses

Conclusion

In conclusion, C and C++ are two very different programming languages that have their own unique features and characteristics. The main difference between C and C++ is that C is function-driven procedural language with no support for objects and classes, whereas C++ is a combination of procedural and object-oriented programming languages. Ultimately, the choice between C and C++ depends on the specific needs and goals of your project, and it is important to carefully consider the trade-offs between the two languages before making a decision. We hope this blog helped you understand the features, applications, disadvantages, and differences between C and C++ to decide which language is more suitable for your projects.

Recommended Reads:

- **Difference between Python and Java**
- **Difference between JavaScript and Java**
- **Difference between Jupyter Notebook and Python IDLE**

- **Difference between C and Java**
- **Difference between C and Python**
- **Difference between C++ and Java**
- **Difference between HTML and CSS**
- **Difference between Float and Double**

FAQs

What is the difference between C and C++?

Which programming language to learn first C or C++?

Is C still used in 2023?

Where are C and C++ used?

Should I learn Python or C?

Can C code be used in C++?

Can I learn C++ without C? Is C++ hard to learn?

Which is faster C or C++?

What are the advantages of C over C++?

What are the advantages of C++ over C?

Disclaimer: This PDF is auto-generated based on the information available on Shiksha as on 23-Dec-2023.