**C and C++**

**Similarities between**[**C**](https://www.geeksforgeeks.org/c/)**and**[**C++**](https://www.geeksforgeeks.org/c-plus-plus/)**are:**

* Both the languages have a similar syntax.
* Code structure of both the languages are same.
* The compilation of both the languages is similar.
* They share the same basic syntax. Nearly all of C’s operators and keywords are also present in C++ and do the same thing.
* C++ has a slightly extended grammar than C, but the basic grammar is the same.
* Basic memory model of both is very close to the hardware.
* Same notions of stack, heap, file-scope and static variables are present in both the languages.

**Differences between**[**C**](https://www.geeksforgeeks.org/c/)**and**[**C++**](https://www.geeksforgeeks.org/c-plus-plus/)**are:**  
C++ can be said a superset of C. Major added features in C++ are [Object-Oriented Programming](https://www.geeksforgeeks.org/object-oriented-programming-in-cpp/), [Exception Handling](https://www.geeksforgeeks.org/exception-handling-c/) and rich C++ Library.

Below is the table of differences between C and C++: 

| C | C++ |
| --- | --- |
| C was developed by Dennis Ritchie between the year 1969 and 1973 at AT&T Bell Labs. | C++ was developed by Bjarne Stroustrup in 1979. |
| C does no support polymorphism, encapsulation, and inheritance which means that C does not support object oriented programming. | C++ supports [polymorphism](https://www.geeksforgeeks.org/polymorphism-in-c/), [encapsulation](https://www.geeksforgeeks.org/encapsulation-in-c/), and [inheritance](https://www.geeksforgeeks.org/inheritance-in-c/) because it is an object oriented programming language. |
| C is a subset of C++. | C++ is a superset of C. |
| C contains 32 [keywords](https://www.geeksforgeeks.org/variables-and-keywords-in-c/). | C++ contains 63 [keywords](https://www.geeksforgeeks.org/cc-tokens/). |
| For the development of code, C supports [procedural programming](https://www.geeksforgeeks.org/introduction-of-programming-paradigms/). | C++ is known as hybrid language because C++ supports both [procedural](https://www.geeksforgeeks.org/introduction-of-programming-paradigms/) and [object oriented programming paradigms](https://www.geeksforgeeks.org/introduction-of-programming-paradigms/). |
| Data and functions are separated in C because it is a procedural programming language. | Data and functions are encapsulated together in form of an object in C++. |
| C does not support information hiding. | Data is hidden by the Encapsulation to ensure that data structures and operators are used as intended. |
| Built-in data types is supported in C. | Built-in & user-defined data types is supported in C++. |
| C is a function driven language because C is a procedural programming language. | C++ is an object driven language because it is an object oriented programming. |
| Function and operator overloading is not supported in C. | Function and operator overloading is supported by C++. |
| C is a function-driven language. | C++ is an object-driven language |
| Functions in C are not defined inside structures. | Functions can be used inside a structure in C++. |
| Namespace features are not present inside the C. | [Namespace](https://www.geeksforgeeks.org/namespace-in-c/) is used by C++, which avoid name collisions. |
| Header file used by C is [stdio.h](https://www.geeksforgeeks.org/whats-difference-between-and/). | Header file used by C++ is [iostream.h](https://www.geeksforgeeks.org/basic-input-output-c/). |
| Reference variables are not supported by C. | Reference variables are supported by C++. |
| Virtual and friend functions are not supported by C. | [Virtual](https://www.geeksforgeeks.org/virtual-function-cpp/) and [friend functions](https://www.geeksforgeeks.org/friend-class-function-cpp/) are supported by C++. |
| C does not support inheritance. | C++ supports inheritance. |
| Instead of focusing on data, C focuses on method or process. | C++ focuses on data instead of focusing on method or procedure. |
| C provides [malloc()](https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/) and [calloc()](https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/) functions for [dynamic memory allocation](https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/), and [free()](https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/) for memory de-allocation. | C++ provides [new operator](https://www.geeksforgeeks.org/new-and-delete-operators-in-cpp-for-dynamic-memory/) for memory allocation and [delete operator](https://www.geeksforgeeks.org/new-and-delete-operators-in-cpp-for-dynamic-memory/) for memory de-allocation. |
| Direct support for exception handling is not supported by C. | [Exception handling](https://www.geeksforgeeks.org/exception-handling-c/) is supported by C++. |
| [scanf()](https://www.geeksforgeeks.org/scanf-and-fscanf-in-c-simple-yet-poweful/) and printf() functions are used for input/output in C. | [cin and cout](https://www.geeksforgeeks.org/basic-input-output-c/) are used for [input/output in C++](https://www.geeksforgeeks.org/basic-input-output-c/). |
| C structures don’t have access modifiers. | C ++ structures have access modifiers. |