



C++ Coding Standards: 101 Rules, Guidelines, and Best Practices

Herb Sutter , Andrei Alexandrescu

[Download now](#)

[Read Online ➔](#)

C++ Coding Standards: 101 Rules, Guidelines, and Best Practices

Herb Sutter , Andrei Alexandrescu

C++ Coding Standards: 101 Rules, Guidelines, and Best Practices Herb Sutter , Andrei Alexandrescu

Consistent, high-quality coding standards improve software quality, reduce time-to-market, promote teamwork, eliminate time wasted on inconsequential matters, and simplify maintenance. Now, two of the world's most respected C++ experts distill the rich collective experience of the global C++ community into a set of coding standards that every developer and development team can understand and use as a basis for their own coding standards.

The authors cover virtually every facet of C++ programming: design and coding style, functions, operators, class design, inheritance, construction/destruction, copying, assignment, namespaces, modules, templates, genericity, exceptions, STL containers and algorithms, and more. Each standard is described concisely, with practical examples. From type definition to error handling, this book presents C++ best practices, including some that have only recently been identified and standardized--techniques you may not know even if you've used C++ for years. Along the way, you'll find answers to questions like

What's worth standardizing--and what isn't?

What are the best ways to code for scalability?

What are the elements of a rational error handling policy?

How (and why) do you avoid unnecessary initialization, cyclic, and definitional dependencies?

When (and how) should you use static and dynamic polymorphism together?

How do you practice "safe" overriding?

When should you provide a no-fail swap?

Why and how should you prevent exceptions from propagating across module boundaries?

Why shouldn't you write namespace declarations or directives in a header file?

Why should you use STL vector and string instead of arrays?

How do you choose the right STL search or sort algorithm?

What rules should you follow to ensure type-safe code? Whether you're working alone or with others, *C++ Coding Standards* will help you write cleaner code--and write it faster, with fewer hassles and less frustration.

C++ Coding Standards: 101 Rules, Guidelines, and Best Practices Details

Date : Published November 1st 2004 by Addison-Wesley Professional (first published October 25th 2004)

ISBN : 0076092018117

Author : Herb Sutter , Andrei Alexandrescu

Format : Paperback 240 pages

Genre : Computer Science, Programming, Science, Technical, Software, Technology, Computers, Nonfiction, Programming Languages, Coding, Reference

 [Download C++ Coding Standards: 101 Rules, Guidelines, and Best P ...pdf](#)

 [Read Online C++ Coding Standards: 101 Rules, Guidelines, and Best ...pdf](#)

Download and Read Free Online C++ Coding Standards: 101 Rules, Guidelines, and Best Practices

Herb Sutter , Andrei Alexandrescu

From Reader Review C++ Coding Standards: 101 Rules, Guidelines, and Best Practices for online ebook

Marcos Santamaria says

Easy to read, with good advices and guidelines.

Daboe_ says

Sometimes it's good to repeat the obvious... Don't optimize prematurely, don't write `list.size() == 0` when `list.isEmpty()` would have accomplished the same, ... Apart from that, this book also covers 99 other recurring pitfalls and misuses in C++ programming. Most of them also apply to other programming languages.

Sascha says

It is a good read but it desperately needs an update with C++11 and 14.

Simon says

One of a few books like "Effective C++" that is an absolute must-read for C++ programmers. Ignore it at your peril!

Adam says

This was a 3-star book based on writing style that got a boost to 4 due to the quality and usefulness of the information it contained.

The information was presented in semi-list form with only a page or two to explain each point. I found that format to be lacking in the details and especially the examples that I would have liked and that would have made it much more clear and easier to understand. The book was probably intended for programmers with more skill and knowledge than I have, which I think contributed to my finding a lot of the information to be difficult to comprehend.

The tips were very useful though, and I suspect I'll be referring back to this in the future both to refresh my memory and to grasp the information better as I become more experienced.

Christophe Addinquy says

Ma note de lecture complète en français ici

Alex says

It's a nice book about C++. Totally worth all the time spent on it.

Still all the books by Scott Meyers remain my all-times favourites on the subject.

Kirill says

Der Klassiker

Setecastronomy says

Helpful, concise and to the point but by now (i.e. c++11/14/17) unfortunately already outdated

Jose Seco Sanz says

All that I hoped of this book. Short, but pretty advanced.

Alexander says

This book is "a must" for people involved in software development. The book discusses good programming practices. Though a reader should have enough experience to appreciate advises of this book.

Furious says

A very good book, full of practical advice for the serious programmer. However there's one big problem with this book; all the information you need can be found in the table of contents!

Steve says

A useful and interesting collection of guidelines for writing C++. Some of the guidelines are "obvious," but his other reviews have noted, worth repeating. Others are not so obvious and demonstrate subtle traps. One

big flaw is that, particularly for the subtle cases, they don't go into enough detail -- they explain that something is bad, maybe even give an example, but not enough to really understand The intricacies of why.

I felt that Scott Meyers' books did a much better job of explaining the why, and where they overlap, you should read his first. Unfortunately, Meyers doesn't cover everything.

Tihauan says

I enjoyed this book even if I am not a C++ programmer. Good tips for when a garbage collector is not there to help you.

Artem Komisarenko says

????????? ??????, ??????? ?? advanced ??????. ?? ?????? ?? ?????? ?????? ?????? ?????? ? ?????????????? ??
????????? ?? ??? ?????, ?? ????? ??????.

? ?????? ?????? ? ?????? ? ? ??????????.
