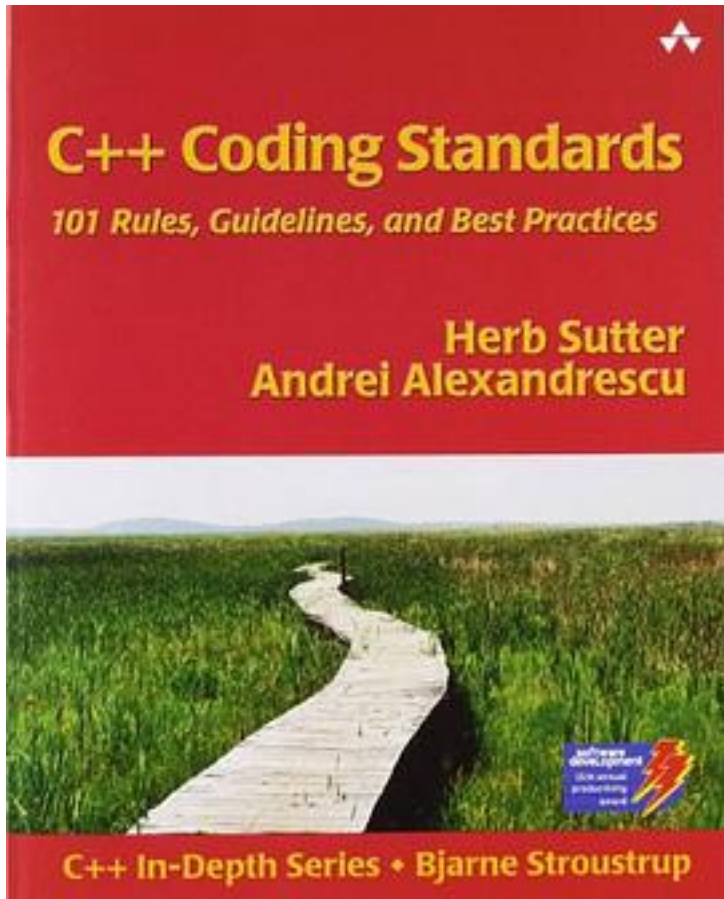


C++ Coding Standards



[C++ Coding Standards 下载链接1](#)

著者:Herb Sutter

出版者:Addison-Wesley Professional

出版时间:2004-11-4

装帧:Paperback

isbn:9780321113580

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.

作者介绍:

Herb Sutter, ISO C++标准委员会主席, C++ Users Journal杂志特邀编辑和专栏作家; 他目前在微软公司领导NET环境下C++语言扩展的设计工作除本书外, 他还撰写了三本广受赞誉的图书: Exceptional C++ Style、Exceptional C++和More Exceptional C++。

Andrei Alexandrescu, 世界顶级的C++专家. “C++ Users Journal杂志的专栏作家, 他的Modem C++ Design一书曾荣获2001年最佳C++图朽称号. 书中所开发的Loki已经成为最负盛名的C++程序库之一。

目录:

[C++ Coding Standards 下载链接1](#)

标签

C++

programming

C/C++

编程

规范

计算机

程序设计

软件开发

评论

讲解太简略，有时候不太准确，几乎没有示例。用来查缺补漏和回顾很不错。

防身必备

水平不够，examples的论证太弱又不想去满地翻quotation，结果就是不少观点都不明所以。

前言中说到，面向对象的本质不是继承或者多态，而是封装和抽象。这一句话就顶过其他很多书的全部内容。

每条建议都值得经常翻看

零零碎碎20天读完了。很有实战意义的一本书。

遵从KISS原则：Correct is better than fast.Simple is better than complex. Clear is better than cute. Safe is better than insurance.还有一些design principle 比如single responsibility, least surprise, open/close balabala 一边写一边悟吧。。

老头的东西都是浓缩了的精华，于是这本书以略读为主查漏补缺，仍然发现很多自己不明白的地方。另外这些standard光是读起来的话可能十分晦涩，重在实践，良好的代码习惯不得不说是编程进阶必备。

脱离入门必备 之一

这本书相当于对 effective c++, exceptional c++的总结, 还是那两本深刻

描述C++
代码标准规范，好久之前看过的，不过现在对于里面内容有些模糊了……但是对于C++代码风格倒是接触很多，已熟悉多数风格，心中已有自己的好坏标准（当然是建立在公认良好标准基础之上啦）。

其实，认真研习标准库或优秀的开源库代码如STL，Boost，Loki，ATL等，在学习其中的程序设计方法同时培养个人的良好代码风格。 http://huangjunkun.blogspot.com/2012/06/blog-post_14.html

可作为编码标准

越来越喜欢读Herb Sutter的书了，希望他也能出类似effective modern c++针对C++11/14/17的新版本，第一遍感觉自己还没能完全理解里面的所有细则，后面我抽时间多读几遍。

C++编码经验之谈

十分受用

[C++ Coding Standards 下载链接1](#)

书评

其实这本书很鸡肋。因为此书是对一条条的规范、原则、实践等的高度提炼，能力到了自然能理解，能力没到看完也不一定能理解，此时你需要类似《Unix编程艺术》的书，当然你仍然需要足够的实践来支持，要不就会像我一样在这里说大话！如果你是完美主义者，如果你本来就注重思维...

温故而知新，重新复习一些C++的知识，有一段时间，出现了非常多关于C++编程规范类型的书籍。这些类型的书籍，很大部分内容是相同的，个人比较喜欢看《C++编程规范》，100条，条款来自《Effective C++》、《More Effective C++》、《Effective STL》、《Exceptional C++》、《Mo...

号称是20年集大成之作，罗列了一大堆最佳实践的条款 有口号，有说明，有实作，形式上挺好 前面一些谈设计，组织，策略上的条款是很实在，后面展开谈细节，模板，异常，容器，算法之类，就难逃教条主义的嫌疑了，晦涩，模糊，说服力不强 中间用的例子有些也不是很清晰贴切

引用pongba的话：C++中众多的细节虽然在库设计者手里面有其用武之地，但普通程序员则根本无需过多关注，尤其是没有实际动机的。关注编码实践准则才是真正需要花时间掌握的东西！

本评论转自我的Blog
转载必须包含本声明、保持本文完整。并以超链形式注明作者编程随想和本文原始地址：
<http://program-think.blogspot.com/2009/01/cxx-coding-standards-101-rules.html>
全书的101个条款分布在如下的12部分中，下面来挨个介绍一下。 1、组织与策略 这部分...

[2018.10 更新] 关键字： 面向对象设计 (OOD), 封装性 (encapsulation), 抽象 (abstraction), 不变式 (invariants), 分离编译 (separate compilation), plmpl 惯用法, 聚合类 (aggregate class), 静态检查 (static checking), SOLID 原则, 替换原则 (Liskov substitution princ...

其实我想看个实践，别人的方法。副标题叫Best Practices，但是Practice很少。
这本书的内容都缺少实践，站着说话不腰疼，对于我这样的菜鸟感觉是这样的。
比如这条，比如有个巨类： CHugeClass { function0(); function1(); function2(); function3(); f...

比较轻量级的一本书。如果你已经看过 effective c++ ， exceptional c++系列，那这本书只用翻翻目录就行了。

名字起的很有吸引力，其实内容很多章节讲的东西，都是<>里面的，我不知道原版书的文采怎么样？所以我不好妄加评论，但这本中文版的译者的翻译水平，我真的不敢苟同，翻译的僵硬，晦涩，缺乏灵活性，估计就是逐字翻译的，唉，浪费了我几十块钱，买了一本多余的书！

我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了
我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了
我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看过了 我看...

光买了书，唉没时间看书啊！我电脑Z差啊，学得头都大了啊！还好，室友告诉我上猎豹网校，看那个视频课程学。嘿嘿，这是个简单容易的办法！这下不再担心买了书，束之高阁了！

[C++ Coding Standards 下载链接1](#)