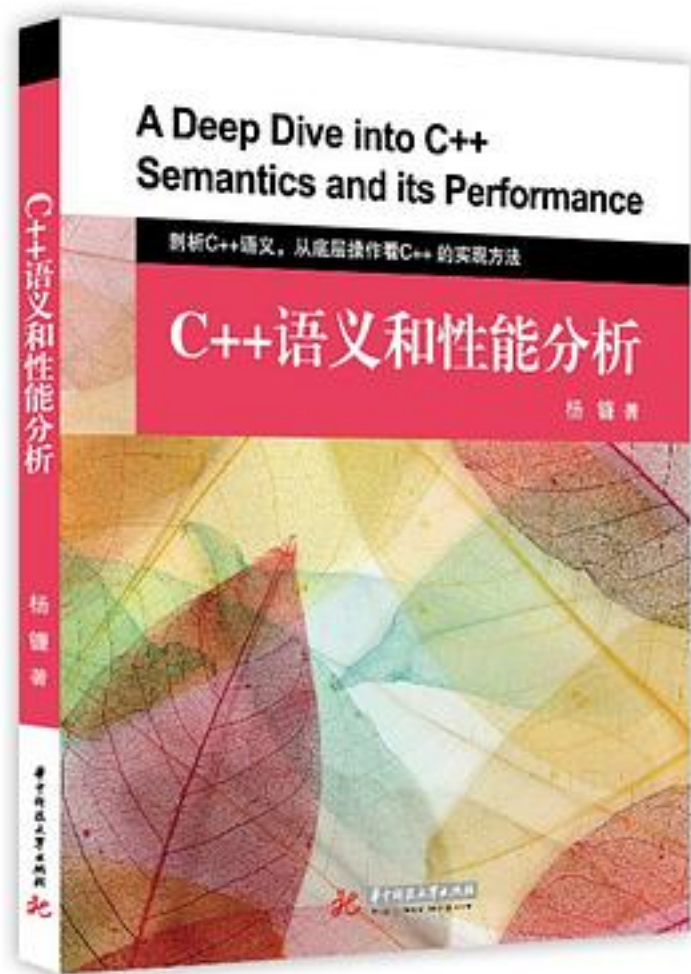


C++语义和性能分析



[C++语义和性能分析_下载链接1](#)

著者:杨健

出版者:华中科技大学出版社

出版时间:2017-11-10

装帧:平装

isbn:9787568033299

"本书从C++的发展历史、类型系统、语义以及性能的视角，给读者展现了C++语言的基

本理念和发展主线。一方面用抽象的理论框架，比如集合论、有限自动机、类型和类型系统等诠释了C++的理论方面；另一方面又从语义着手，从底层操作剖析了C++的实现方法。使读者既能站得高、看得远，又能把握C++的实质，从而全面掌握C++语言。

本书作者有20多年的C++开发经验，在微软总部工作期间，用C++参与开发过包括Windows XP系统在内的许多知名商用软件系统，在C++语言上有着深厚的功底。在本书作者看来，理解C++的精髓和灵魂，必须从C++类型系统着手，这样才能写出性能优异而又易于维护的系统。另外，从C++11到C++17的最新构造，都没有脱离C++类型系统的主体思想。熟读本书可以作为学习C++17的前奏。"

作者介绍:

杨镡，1985年毕业于华中工学院计算机系。1988年赴美留学，获波特兰州立大学计算机科学硕士学位。1995年加盟微软总部，在微软研究院从事软件安全和性能研发方面的工作。在微软工作的15年里，他先后参加了Windows Vista、Windows Server 2008、MSN以及Bing的研发，在软件工程、软件性能优化、软件安全等领域有很深厚的功底和实践。他在不同的国际会议上发表了多篇论文，内容涉及软件工程、机器学习和高级语言。在繁忙的工作之余，他是滑雪高手，对如何提高滑雪板的性能也颇有研究。

目录: "目录

第1章 C++简史 1

1.1 C++ 的历史背景 1

1.2 C++大事记 5

1.2.1 1979年：C with classes诞生 5

1.2.2 1983年8月：C++正式命名 6

1.2.3 1986年：《The C++ Programming Language》出版 7

1.2.4 1987年：C++2.0发布 7

1.2.5 1990年：《The Annotated C++ Reference Manual》出版 8

1.2.6 1994年：STL诞生 8

1.2.7 1998年：ISO C++ 正式通过 9

1.2.8 2011年：C++11颁布 9

1.3 C++的进化和改进理念 10

1.4 C++的历史贡献及未来 11

第2章 程序设计语言的语义 14

2.1 哲学基础 14

2.2 语义的形式化描述 14

2.3 操作性语义 16

2.4 语义描述涉及的主要元素 17

第3章 C++类型系统 19

3.1 类型系统简介 19

3.2 数据类型的表达规则 21

3.2.1 基本数据类型的表达规则 22

3.2.2 用户自定义类型的表达规则 23

3.3 类型实例的构造规则 25

3.3.1 类型和类型实例 25

3.3.2 类型实例的构造和析构 27

3.4 类型的操作规则 33

3.4.1 context 33

3.4.2 函数 34

3.4.3 过程 35

3.4.4 过程函数 35

3.4.5 方法	36
3.4.6 特别讨论：过程方法的自动机诠释	37
3.4.7 运算符	38
3.5 C++类型系统的特征	39
3.5.1 特征1：支持OOP	39
3.5.2 特征2：强静态类型	40
3.5.3 特征3：UDT是一等公民	41
3.5.4 特征4：类型清除	45
3.5.5 特征5：支持模板	49
3.5.6 特征6：指针和REF	53
3.5.7 特征7：类型修正符	60
3.6 其他（非C++）类型系统	62
3.7 类型系统的意义和用途	62
3.7.1 程序的精确性保障	62
3.7.2 性能的可保障性	64
第4章 C++变量语义	66
4.1 变量简介	66
4.2 C++变量的属性	68
4.2.1 C++变量的存储属性	68
4.2.2 C++ 变量的值语义	95
4.2.3 C++变量的自洽性	106
4.2.4 C++变量的总分性	109
4.2.5 C++变量的多态性	112
4.3 临时变量	113
第5章 C++对OOP的支持	116
5.1 object的内存布局	117
5.2 继承	119
5.2.1 继承object的内存布局	123
5.2.2 多继承	128
5.2.3 继承类object的语义规则	130
5.3 多态类和多态变量	137
5.4 C++对抽象的支持	139
第6章 C++ 性能分析	141
6.1 性能分析原理	141
6.2 算法的重要性	149
6.3 RAM内存读取的性能	153
6.3.1 单线程heap管理器	154
6.3.2 定制化heap管理系统	157
6.4 关于缓存	164
6.5 关于I/O	175
6.6 C++的特殊性能问题	176
6.6.1 临时变量的性能问题	176
6.6.2 STL 容器	180
6.6.3 Type Erasure	185
6.6.4 异常处理	187
6.6.5 继承的性能问题	194
6.6.6 直接函数、间接函数、内联函数	200
参考文献	204
"	

• • • • • [\(收起\)](#)

[C++语义和性能分析_下载链接1](#)

标签

计算机

Programming

C++

语义

编程

类型

变量

优化

评论

书很薄，实际上看作者的几篇博客就可以理解作者要表达啥了；后面的性能分析部分内容也有点水；核心内容值语义这部分内容也不如c++对象那本聊的深入；副标题有c++17，实际连11都没涉及，因为作者的海外工作背景，个别翻译很奇怪，比如cluster一般是集群吧。最重要的作者总结出来很多之前没听过的术语，但是并没有觉得这些术语怎么和性能优化建立的联系。不推荐。

辣鸡，目录立意很高落实到内容上属于扫盲，浪费钱。比DC的自杀小队真人电影版都坑爹。

从另一个角度阐述c++的一些设计和带来的影响

存粹探讨个技术问题，书本 6.6.2

STL容器，184页下方最后一段，“前者不具备值语义，必须通过函数Init来复制程序，花销颇大，后者具备值语义，可以通过位拷贝来复制程序，简单高效”。这句话是不是写错了，是不是反了？

[C++语义和性能分析_下载链接1](#)

书评

趁这段空闲时间，把《C++语义和性能分析》看了两遍，感觉受益良多。此书虽篇幅不长，仅仅2百页多，但以较高的技术角度讲述了C++语言的发展历程，类型系统，语言语义对于面向对象的支持，性能考虑等内容，有深度，有思想，所举的例子简单明白，让人能够理解关键点。作者不愧是...

存粹探讨个技术问题，书本 6.6.2

STL容器，184页下方最后一段，“前者不具备值语义，必须通过函数Init来复制程序，花销颇大，后者具备值语义，可以通过位拷贝来复制程序，简单高效”。这句话是不是写错了，是不是反了？ 还得凑够140个字，真是醉了，看看这段写完够140个字了吗...

[C++语义和性能分析_下载链接1](#)