

<<C++沉思录>>

图书基本信息

书名：<<C++沉思录>>

13位ISBN编号：9787115308511

10位ISBN编号：7115308519

出版时间：2013-2

出版时间：人民邮电出版社

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<C++沉思录>>

内容概要

《C++沉思录(英文版)》基于作者在知名技术杂志发表的技术文章、世界各地发表的演讲以及斯坦福大学的课程讲义整理、写作而成，融聚了作者10多年C++程序生涯的真知灼见。全书分为6篇32章，分别对C++语言的历史和特点、类和继承、STL与泛型编程、库的设计等几大技术话题进行了详细而深入的讨论，细微之处几乎涵盖了C++所有的设计思想和技术细节。全书通过精心挑选的实例，向读者传达先进的程序设计的方法和理念。

<<C++沉思录>>

作者简介

Andrew Koenig是AT&T大规模程序研发部（前贝尔实验室）成员。他从1986年开始从事C语言的研究，1977年加入贝尔实验室。他编写了一些早期的类库，并在1988年组织召开了第一个具有相当规模的C++会议。在ISO/ANSI C++委员会成立的1989年，他就加入了该委员会，并一直担任项目编辑。

书籍目录

目 录第0章 序幕 10.1 第一次尝试 10.2 不用类来实现 40.3 为什么用C++更简单 50.4 一个更大的例子 60.5 结论 6第一篇 动机第1章 为什么我用C++ 111.1 问题 111.2 历史背景 121.3 自动软件发布 121.4 进入C++ 151.5 重复利用的软件 201.6 后记 21第2章 为什么用C++工作 232.1 小项目的成功 232.2 抽象 252.3 机器应该为人服务 28第3章 生活在现实世界中 29第二篇 类和继承第4章 类设计者的核查表 37第5章 代理类 475.1 问题 475.2 经典解决方案 485.3 虚复制函数 495.4 定义代理类 505.5 小结 53第6章 句柄：第一部分 556.1 问题 556.2 一个简单的类 566.3 绑定到句柄 586.4 获取对象 586.5 简单的实现 596.6 引用计数型句柄 606.7 写时复制 626.8 讨论 63第7章 句柄：第二部分 677.1 回顾 687.2 分离引用计数 697.3 对引用计数的抽象 707.4 存取函数和写时复制 737.5 讨论 73第8章 一个面向对象程序范例 758.1 问题描述 758.2 面向对象的解决方案 768.3 句柄类 798.4 扩展1：新操作 828.5 扩展2：增加新的节点类型 858.6 反思 86第9章 一个课堂练习的分析(上) 899.1 问题描述 899.2 接口设计 919.3 补遗 939.4 测试接口 949.5 策略 959.6 方案 969.7 图像的组合 999.8 结论 102第10章 一个课堂练习的分析(下) 10310.1 策略 10310.2 体验设计的灵活性 11610.3 结论 119第11章 什么时候不应当使用虚函数 12111.1 适用的情况 12111.2 不适用的情况 12211.3 析构函数很特殊 12711.4 小结 129第三篇 模板第12章 设计容器类 13312.1 包含什么 13312.2 复制容器意味着什么 13412.3 怎样获取容器的元素 13712.4 怎样区分读和写 13812.5 怎样处理容器的增长 13912.6 容器支持哪些操作 14112.7 怎样设想容器元素的类型 14112.8 容器和继承 14312.9 设计一个类似数组的类 144第13章 访问容器中的元素 15113.1 模拟指针 15113.2 获取数据 15313.3 遗留问题 15513.4 指向const Array的Pointer 15913.5 有用的增强操作 161第14章 迭代器 16714.1 完成Pointer类 16714.2 什么是迭代器 17014.3 删除元素 17114.4 删除容器 17214.5 其他设计考虑 17314.6 讨论 174第15章 序列 17515.1 技术状况 17515.2 基本的传统观点 17615.3 增加一些额外操作 18115.4 使用范例 18415.5 再增加一些 18815.6 请你思考 190第16章 作为接口的模板 19116.1 问题 19116.2 第一个例子 19216.3 分离迭代方式 19216.4 遍历任意类型 19516.5 增加其他类型 19616.6 将存储技术抽象化 19616.7 实证 19916.8 小结 200第17章 模板和泛型算法 20317.1 一个特例 20417.2 泛型化元素类型 20517.3 推迟计数 20517.4 地址独立性 20717.5 查找非数组 20817.6 讨论 210第18章 泛型迭代器 21318.1 一个不同的算法 21318.2 需求的分类 21518.3 输入迭代器 21618.4 输出迭代器 21618.5 前向迭代器 21718.6 双向迭代器 21818.7 随机存取迭代器 21818.8 是继承吗 22018.9 性能 22018.10 小结 221第19章 使用泛型迭代器 22319.1 迭代器类型 22419.2 虚拟序列 22419.3 输出流迭代器 22719.4 输入流迭代器 22919.5 讨论 232第20章 迭代器配接器 23320.1 一个例子 23320.2 方向不对称性 23520.3 一致性和不对称性 23620.4 自动反向 23720.5 讨论 240第21章 函数对象 24121.1 一个例子 24121.2 函数指针 24421.3 函数对象 24621.4 函数对象模板 24821.5 隐藏中间类型 24921.6 一种类型包罗万象 25021.7 实现 25121.8 讨论 253第22章 函数配接器 25522.1 为什么是函数对象 25522.2 用于内建操作符的函数对象 25622.3 绑定者(Binders) 25722.4 更深入地探讨 25822.5 接口继承 25922.6 使用这些类 26022.7 讨论 261第四篇 库第23章 日常使用的库 26523.1 问题 26523.2 理解问题：第1部分 26723.3 实现：第1部分 26723.4 理解问题：第2部分 27023.5 实现：第2部分 27023.6 讨论 272第24章 一个库接口设计实例 27524.1 复杂问题 27624.2 优化接口 27724.3 温故知新 27924.4 编写代码 28024.5 结论 282第25章 库设计就是语言设计 28325.1 字符串 28325.2 内存耗尽 28425.3 复制 28725.4 隐藏实现 29025.5 缺省构造函数 29225.6 其他操作 29325.7 子字符串 29525.8 结论 296第26章 语言设计就是库设计 29726.1 抽象数据类型 29726.2 库和抽象数据类型 29926.3 内存分配 30226.4 按成员赋值(memberwise assignment)和初始化 30326.5 异常处理 30526.6 小结 306第五篇 技术第27章 自己跟踪自己的类 30927.1 设计一个跟踪类 30927.2 创建死代码 31227.3 生成对象的审计跟踪 31327.4 验证容器行为 31527.5 小结 320第28章 在簇中分配对象 32128.1 问题 32128.2 设

<<C++沉思录>>

计方案 32128.3 实现 32428.4 加入继承 32628.5 小结 327第29章 应用器、操纵器和函数对象
32929.1 问题 32929.2 一种解决方案 33229.3 另一种不同的解决方案 33229.4 多个参数
33429.5 一个例子 33529.6 简化 33729.7 思考 33829.8 历史记录、参考资料和致谢 339第30
章 将应用程序库从输入输出中分离出来 34130.1 问题 34130.2 解决方案1：技巧加蛮力
34230.3 解决方案2：抽象输出 34330.4 解决方案3：技巧而无蛮力 34530.5 评论 348第六篇
总结第31章 通过复杂性获取简单性 35131.1 世界是复杂的 35131.2 复杂性变得隐蔽 35231.3
计算机也是一样 35331.4 计算机解决实际问题 35431.5 类库和语言语义 35531.6 很难使事情变
得容易 35731.7 抽象和接口 35731.8 复杂度的守恒 358第32章 说了Hello world后再做什么
36132.1 找当地的专家 36132.2 选一种工具包并适应它 36232.3 C的某些部分是必需的
36232.4 C的其他部分不是必需的 36432.5 给自己设一些问题 36632.6 结论 368索引 371

<<C++沉思录>>

编辑推荐

《C++沉思录(英文版)》是一本关于C++的经典著作，适合有一定经验的C++程序员阅读学习，可以帮助读者加强提高技术能力，成为C++程序设计的高手。

<<C++沉思录>>

名人推荐

Koenig和Moo的《C++沉思录》将会为许多人提供有用的帮助。

对于C++是什么样的以及能够做些什么，他们的先见之明在《C++沉思录》里随处可见。

——Bjarne Stroustrup, C++之父由于源自杂志的专栏文章，因此《C++沉思录》的内容具有高度的可读性，知识密度高，表现力强。

更重要的是，这些文章是在发表之后若干年，由原作者挑选出来，经过了多年的沉淀和反思，重新编辑整理，加上自己多年的心得与思考，自然有一种千锤百炼的韧性和纯度。

也正因为如此，作者当仁不让地把这本书命名为Ruminations on C++，rumination一词，充分显现出作者的自信和对这本书的珍爱。

——孟岩

<<C++沉思录>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>