

<<日臻完善>>

图书基本信息

书名：<<日臻完善>>

13位ISBN编号：9787113107192

10位ISBN编号：7113107192

出版时间：2010-1

出版时间：中国铁道出版社

作者：石磊玉

页数：385

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>

## &lt;&lt;日臻完善&gt;&gt;

## 前言

可能你还没有注意到，与程序编码相比，程序调试在开发中的比重更大，成本更高。

有开发经验的人都知道，对于一个要实现的逻辑，要一口气顺利写完代码而且运行正确几乎是不可能的，其中出现的语法错误可以通过编译器的帮助来解决，这些错误相对简单，对于潜伏在代码中的逻辑错误，在运行时才会暴露出来，如导致程序崩溃或者给出错误的结果等，而且它们可能在发布时都没有被发现，成千上万的用户在使用时偶尔发现，这类错误会影响软件的市场。

调试程序是一件烦琐的事情，掌握必要的调试手段可以提高程序开发效率。

对于一个程序开发者，除了掌握软件开发技术之外，还要掌握调试技术，没有掌握调试技术就如同一个人没有了免疫系统一样。

能够巧妙、熟练地使用各种调试手段可以让程序开发工作变得轻松。

程序调试存在于编码、测试、维护等诸多环节，一个合格的开发者应该能够游刃有余地处理各个环节出现的错误。

调试技术更多的是一种经验，需要通过不断积累才能逐渐掌握，很多开发者都是在长时间的调试后才慢慢熟练起来。

要是有一本综合介绍调试技术的书籍，就可以大大减少初学者入门的时间。

基于这个想法，本书精选了使用VisualC++开发所必须的调试技术，从软件开发的多个方面来介绍相关的调试方法，如VisualC#工具的调试功能、共享库的调试、多线程程序的调试、内存泄漏的调试、异常处理、SQL调试等，技术针对性非常强。

读者通过阅读此书，能够迅速掌握各种程序调试手段。

全书学习地图本书总结了作者软件开发中的调试经验，将这些经验整理成册，是为了帮助读者轻松掌握调试技术。

本书着眼于软件工程，系统介绍了各种调试技术、避免错误的方法、缺陷管理等多个方面的内容。

本书的读者对象为已经学习过C/C++编程语言，掌握了基本的C++语法，有一定程序开发基础，并且可以使用VisualC++进行简单程序开发的人员。

## <<日臻完善>>

### 内容概要

本书在介绍相关内容时，通过实例阐述使晦涩的理论知识变得生动易懂。

实例中的代码都是在Visual C++ 6.0开发环境中编写的，但调试理念和思想与软件开发环境无关，只是调试工具的操作细节不同。

本书以软件开发过程的全局来介绍软件调试，涉及编码过程的规范、测试用例的编写、事后调试、缺陷管理等多个开发环节的内容，介绍了各种常用开发技术的调试方法，包括静态链接库调试、动态链接库调试、多线程调试、内存泄漏、内核对象泄漏等。

本书主要面向软件程序编码人员、程序测试人员等软件领域的技术从业人员，同时软件开发管理人员也可从本书中了解到软件缺陷管理方面的内容。

## 书籍目录

第1章 绪论 1.1 不可避免的调试工作 1.2 掌握调试 1.3 轻松调试的必备条件 第2章 编写良好的代码 2.1 代码编写规范 2.2 静态代码检查 2.3 重构 2.4 代码结构分析 2.5 测试代码 第3章 Visual C++调试基础 3.1 Visual C++调试工具 3.2 使用跟踪语句 3.3 使用断言 3.4 工程选项 3.5 调试版本和发布版本 3.6 调试发布版本 3.7 符号文件 3.8 使用预处理指令 第4章 库调试 4.1 运行库概述 4.2 创建链接库 4.3 调试静态链接库 4.4 调试动态链接库 第5章 多线程程序调试 5.1 多线程概述 5.2 线程的同步与互斥 5.3 多线程下内存操作 5.4 编写安全的线程代码 5.5 调试方法 第6章 SQL调试 6.1 SQL概述 6.2 数据库开发 6.3 编写良好的SQL语句 6.4 数据库性能调试 6.5 数据库结构分析 第7章 程序错误 7.1 程序错误类型 7.2 防御性编程 7.3 查看错误 第8章 内存漏洞及调试 第9章 内核对象泄漏及调试 第10章 结构异常处理 第11章 事后调试 第12章 软件缺陷管理 附录

## 章节摘录

插图：2.3 重构重构（refactoring）就是在不改变软件现有功能的基础上，通过调整程序代码改善软件的质量、性能，使其程序的设计模式和架构更趋合理，提高软件的扩展性和维护性。

对于很多软件开发项目，在项目开始阶段，需求往往是不明确的，某些流程或者某些细节随着项目的进展会发生变化；某些软件系统，在完成开发后，还存在升级的可能，以适应新的技术发展趋势，升级的内容或技术细节在项目开发起始阶段是很难精确确定的。

因此系统设计人员对软件项目往往只能从大方向予以把控，细枝末节很难把握，要让系统设计人员的设计能够完美的预见未来的任何变化、或者可以容纳任何扩展和改变，这几乎是不可能的。

功能的变化导致设计的调整是不可避免的，所以“测试为先，持续重构”作为良好开发习惯被越来越多的人所采纳，测试和重构是保证软件质量的法宝。

本节主要介绍代码重构的相关内容，测试的相关内容在2.5节测试代码中介绍。

2.3.1 重构概述重构是指对软件内部结构调整，目的是在不改变“软件之可察行为”的前提下，提高其可理解性，降低其修改成本。

有关重构的内容请阅读《重构—改善既有代码的设计》，在本书中，作者Martin Fowler充分展示了何处可能需要重构，如何将不好的设计改造为良好的设计等。

1. 重构的目的重构的精髓就是在不改变系统功能的情况下，改变系统的实现方式。

为什么要这么做？

投入精力不用来满足客户关心的需求，而是仅仅改变了软件的实现方式，这是否是在浪费客户的投资呢？

## &lt;&lt;日臻完善&gt;&gt;

## 编辑推荐

《日臻完善:软件调试与优化典型应用》包含书中调试基础、调试技术和缺陷管理的所有代码展示凝练代码、规范编码习惯《日臻完善:软件调试与优化典型应用》站在软件工程的高度，剖析各个环节的调试技术。

从编码规范、开发环节中的调试技术、测试环节中的调试技术、维护环节中的调试技术、软件缺陷管理等多个角度对软件调试进行了详细的介绍，并且与软件开发过程一一对应，便于读者从软件工程师的角度把握软件调试，从而在更高的层次上把握全局。

《日臻完善:软件调试与优化典型应用》注重理论结合实际。

在介绍各种技术的同时，给出了针对性很强的示例，而且尽量做到图文并茂。

《日臻完善:软件调试与优化典型应用》中不乏凝练的代码、图表，通过这些读者可以轻松地理解作者意图，把握《日臻完善:软件调试与优化典型应用》精髓。

《日臻完善:软件调试与优化典型应用》不仅针对软件中各种错误给出了调试方法，同时也给出、了预防措施，防患于未然，帮助开发者在开发中尽量少犯错误，养成良好的编码习惯。

<<日臻完善>>

#### 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>