

图书基本信息

书名：<<Visual C++面向对象程序设计教程与实验>>

13位ISBN编号：9787302186212

10位ISBN编号：7302186219

出版时间：2005-9

出版时间：清华大学出版社

作者：温秀梅

页数：352

字数：574000

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>

前言

《Visual C++面向对象程序设计教程与实验》教材2005年出版后，受到读者好评，被许多高校选做教材。

根据师生反馈意见，我们及时修编原书，形成了本版教材，该教材被列为普通高等教育“十一五”国家级规划教材。

修编后的教材保持了与原书内容、风格一致，使采用原书的教师可以自然地过渡到新版教材。

这次修编主要就以下几个方面进行了改进：根据教学改革、实践教学的需要和教师多年的教学经验，适当修改、增补了第3章、第5章、第6章和第7章部分语言和问题描述的表达方式；增加了一套模拟试题，并增加两套模拟试题的答案；修改了部分例题的源程序，对部分例题进行了详细分析；更正了原书中的错误。

作为教材，使用者可以根据自己的需求，选取相应的内容进行教学。

如果前面开设的是《C++语言程序设计》课程，而不是《C语言程序设计》，则可以略过第2章不讲，第7章的部分内容也可以不讲。

如果课时不够，第9章到第12章总体通过一个实例讲解，其余的内容可以让学生自学，也可以在课程的开始先布置让学生自己上机通过实例学习。

本版教材由温秀梅、丁学钧任主编，刘建臣、高丽婷、赵巍任副主编。

参加编写的有：丁学钧（第1~2章）、温秀梅（第3~10章、附录F）、刘建臣（第11章）、高丽婷（第12章）、李建华（附录A、B）、宋淑彩（附录C）、赵巍（附录D）、祁爱华（附录E）、全书最后由温秀梅、丁学钧、刘建臣进行审校并统稿。

由于作者水平有限，书中难免有疏漏和错误之处，恳请广大专家和读者指正。

内容概要

本书在结构上将C++面向对象程序设计的思想和方法作为重点，结合例题进行了详细的分析解释，除在每章后附有习题外，还在附录中整合了实验设计。

全书结构严谨、通俗易懂，兼有普及与提高的双重功能。

本书由三部分组成。

第一部分第1~8章结合实例深入浅出地讲解了C++面向对象程序设计的思想和方法；第二部分第9~12章是关于Visual C++的MFC程序设计，该内容写得简明扼要，通俗易懂，以便读者理解；第三部分附录包括重要的实验内容设计及Visual C++ 6.0环境介绍，这是掌握编程语言的重要环节。

本书遵循少而精的原则，力求做到版面清晰、结构紧凑、信息含量高，因此特别适宜作为计算机专业本科教材。

同时，还可以作为自学或函授学习的参考书。

作者简介

第二版前言

《Visual C++面向对象程序设计教程与实验》教材2005年出版后，受到读者好评，被许多高校选做教材。

根据师生反馈意见，我们及时修编原书，形成了本版教材，该教材被列为普通高等教育“十一五”国家级规划教材。

修编后的教材保持了与原书内容、风格一致，使采用原书的教师可以自然地过渡到新版教材。

这次修编主要就以下几个方面进行了改进：

根据教学改革、实践教学的需要和教师多年的教学经验，适当修改、增补了第3章、第5章、第6章和第7章部分语言和问题描述的表达方式；增加了一套模拟考试题，并增加两套模拟考试题的答案；修改了部分例题的源程序，对部分例题进行了详细分析；更正了原书中的错误。

作为教材，使用者可以根据自己的需求，选取相应的内容进行教学。

如果前面开设的是《C++语言程序设计》课程，而不是《C语言程序设计》，则可以略过第2章不讲，第7章的部分内容也可以不讲。

如果课时不够，第9章到第12章总体通过一个实例讲解，其余的内容可以让学生自学，也可以在课程的开始先布置让学生自己上机通过实例学习。

本版教材由温秀梅、丁学钧任主编，刘建臣、高丽婷、赵巍任副主编。

参加编写的有：丁学钧（第1~2章）、温秀梅（第3~10章、附录F）、刘建臣（第11章）、高丽婷（第12章）、李建华（附录A、B）、宋淑彩（附录C）、赵巍（附录D）、祁爱华（附录E）、全书最后由温秀梅、丁学钧、刘建臣进行审校并统稿。

由于作者水平有限，书中难免有疏漏和错误之处，恳请广大专家和读者指正。

编者

2009年1月

书籍目录

第1章 绪论 1.1 面向对象方法的起源 1.2 面向对象是软件方法学的返璞归真 1.3 结构化程序设计与面向对象程序设计 1.4 面向对象的基本概念和面向对象系统的特性 1.4.1 面向对象的基本概念 1.4.2 面向对象系统的特性 1.5 面向对象程序设计语言的四大家族 1.6 面向对象的系统开发方法 1.6.1 面向对象分析OOA 1.6.2 面向对象设计OOD 1.6.3 OOA和OOD的基本步骤 1.7 面向对象程序设计举例 习题第2章 C++语言对C语言的扩充 2.1 C++语言的特点 2.2 C++语言的文件扩展名 2.3 注释符 2.4 名字空间 2.5 C++语言的输入输出 2.6 变量的定义 2.7 强制类型转换 2.8 动态内存的分配与释放 2.9 作用域运算符(::) 2.10 引用 2.11 const修饰符 2.12 字符串 2.13 C++语言中函数的新特性 2.13.1 函数原型(function prototype) 2.13.2 内联(inline)函数 2.13.3 带默认参数的函数 2.13.4 函数重载(overload) 2.13.5 函数模板(function template) 习题第3章 类和对象 3.1 类 3.1.1 类的定义 3.1.2 类中成员函数的定义 3.2 对象 3.3 构造函数和析构函数 3.3.1 构造函数 3.3.2 析构函数 3.4 类的聚集——对象成员 3.5 静态成员 3.6 指向类成员的指针 3.7 综合举例 习题第4章 友元 4.1 友元的概念和定义 4.2 友元函数 4.3 友元成员 4.4 友元类 4.5 友元综合举例 习题第5章 继承与派生 5.1 单一继承 5.1.1 继承与派生 5.1.2 派生类的定义 5.1.3 类的继承方式 5.1.4 派生类的构造函数和析构函数 5.1.5 派生类对基类成员的继承 5.2 多重继承 5.2.1 多重继承的概念和定义 5.2.2 二义性和支配规则 5.2.3 赋值兼容规则 5.3 虚基类 5.3.1 虚基类的概念 5.3.2 多重继承的构造函数和析构函数 5.4 类模板 5.5 应用举例 习题第6章 多态性和虚函数 6.1 运算符重载 6.1.1 运算符重载概述 6.1.2 用成员函数重载运算符 6.1.3 用友元函数重载运算符 6.1.4 几个常用运算符的重载 6.2 虚函数 6.2.1 为什么要引入虚函数 6.2.2 虚函数的定义与使用 6.3 纯虚函数和抽象类 6.3.1 纯虚函数的概念 6.3.2 抽象类的概念 6.4 虚析构函数 习题第7章 C++语言的输入输出流库 7.1 C++语言标准输入输出 7.1.1 C++语言输入输出流库简介 7.1.2 C++语言格式化输入输出 7.2 用户自定义数据类型的I/O流 7.3 文件输入输出流 7.3.1 文件I/O流 7.3.2 文件的打开与关闭 7.3.3 文件的读写操作 习题第8章 异常处理 8.1 异常处理概述 8.2 C++语言异常处理的实现 8.3 重新抛出异常和异常规范 8.4 C++标准库中的异常类 习题第9章 Windows编程基础和MFC编程基础 9.1 Windows编程基础 9.2 MFC编程基础 9.2.1 MFC编程概述 9.2.2 MFC的类层次 9.2.3 常用的MFC类 9.2.4 MFC应用程序的消息映射 9.2.5 一个最简单的MFC应用程序 9.2.6 典型的Windows应用程序 习题第10章 对话框和控件 10.1 对话框和控件的基本概念 10.1.1 对话框的基本概念 10.1.2 控件的基本概念 10.2 使用AppWizard开发MFC应用程序 10.2.1 生成基于对话框的MFC应用程序框架 10.2.2 AppWizard向导自动生成的文件 10.3 基本控件 10.3.1 按钮控件 10.3.2 编辑框控件(文本框控件) 10.3.3 静态控件 10.3.4 列表框控件 10.3.5 滚动条控件 10.3.6 组合框控件 10.3.7 基本控件应用举例 10.4 通用对话框 10.4.1 CColorDialog类 10.4.2 CFileDialog类 10.4.3 CFindReplaceDialog类 10.4.4 CFontDialog类 10.4.5 CPrintDialog类 10.4.6 通用对话框应用举例 习题第11章 菜单和文档/视图结构 11.1 文档/视图的概念 11.2 文档类 11.3 视图类 11.4 菜单 11.5 菜单和文档/视图结构程序设计举例 习题第12章 图形设备接口 12.1 设备环境 12.2 映射模式 12.3 绘制基本图形 12.4 画笔和画刷 12.4.1 画笔 12.4.2 画刷 12.4.3 画笔和画刷的应用程序举例 12.5 字体 习题附录A 程序的调试与运行附录B 标准字符ASCII表附录C 实验附录D 模拟考试题一附录E 模拟考试题二附录F 参考课时安排主要参考文献

章节摘录

插图：第1章 绪论面向对象程序设计是软件系统设计与实现的新方法，这种方法是通过增加软件的可扩充性和可重用性来提高程序员的生产能力，控制软件的复杂性，降低软件维护的开销。

因此，它的应用使软件开发的难度和费用大幅度降低，已为世界软件产业带来了革命性的突破。

1.1 面向对象方法的起源 “对象”一词在现实生活中经常会遇到，它表示现实世界中的某个具体的事物。

社会的进步和计算机科学的发展是相互促进的，随着计算机的普及和应用，人们越来越希望能更直接地与计算机进行交互，而不需要经过专门学习和长时间训练后才能使用它。

这使得软件设计人员的负担越来越重，软件的实现越来越复杂，并且对计算机领域自身的发展也提出了新的要求。

利用传统的程序设计思想无法满足这一要求时，人们就开始寻求一种能帮助人类解决问题的自然方法，这就是“面向对象”技术。

20世纪50年代的程序都是用指令代码或汇编语言编写的，这种程序的设计相当复杂，编制和调试一个稍大一点的程序常常要花费很长时间，培养一个熟练的程序员更需经过长期训练和实践，这种局面严重影响了计算机的普及和应用。

20世纪60年代高级语言的出现大大简化了程序设计，缩短了软件开发周期，显示出了强大的生命力。

此后，编制程序已不再是专业软件人员才能做的事了，一般工程技术人员花较短的时间学习后，也可以使用计算机解题。

这个时期，随着计算机日益广泛地渗透到各个学科和技术领域，一系列不同风格、为不同目标服务的程序设计语言发展起来了，其中较为著名的有FORTRAN、COBOL、ALGOL、LISP、PASCAL等十几种语言。

高级语言的蓬勃兴起，使得编译原理和形式语言理论日趋完善，这是该时期的主要特征。

但是就整个程序设计方法而言，并无实质性的改进。

自20世纪60年代末到20世纪70年代初，出现了大型软件系统，如操作系统、数据库，这给程序设计带来了新的问题。

大型系统的研制需要花费大量的资金和人力，可是研制出来的产品却可靠性差、错误多、不易维护和修改。

一个大型操作系统有时需要每年几千人的工作量，而所获得的系统又常常会隐藏着几百甚至几千个错误。

当时，人们称这种现象为“软件危机”。

编辑推荐

《Visual C++面向对象程序设计教程与实验(第2版)》特点：教学目标明确，注重理论与实践的结合；教学方法灵活，培养学生自主学习的能力；教学内容行进，反映了计算机学科的最新发展，教学模式完善，提供配套的教学资源解决方案。

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>