

图书基本信息

书名：<<面向对象程序设计与Visual C++6.0教程>>

13位ISBN编号：9787561532539

10位ISBN编号：7561532539

出版时间：2009-7

出版时间：厦门大学出版社

作者：邹金安 编

页数：366

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

面向对象程序设计（Object-Oriented Programming, OOP），指的是一种程序开发的方法论。它将对象作为程序的基本单元，将操作和数据封装其中，以提高软件的重用性、灵活性和扩展性。

目前已经证实面向对象程序设计加强了程序的灵活性、重用性和可维护性，并且在大型项目设计中广为应用。

面向对象程序设计能够让人们更简单地设计并维护程序，使得程序更加便于分析、设计和理解。

OOP有三个特性：一是封装性。

定义了类，封装了数据和操作的抽象数据类型。

在OOP语言中，类是模块、封装和数据抽象的基础。

二是继承性。

从已存在的类型中继承元素（属性和方法），改变或扩展旧类型的方法。

三是多态性。

它允许使用相同的方法操作不同类型的对象（通常是子类对象），使得类的可用性进一步提高，程序也因此更容易维护和扩展。

C++是最典型的面向对象程序设计语言，能全面反映OOP的特点，而且C++和Java是目前和今后相当长的一段时间里被最为广泛使用的语言，又因为掌握了C++，再自学Java比较容易，所以本书借助C++来介绍OOP方法。

但是，仅学C++是不够的，不能很好地进行应用编程，而且Visual C++6.0是一个全面的应用程序开发环境，它为程序开发人员提供了使用面向对象的C++语言来开发Windows应用程序的强大平台，目前应用相当广泛，所以本书用一定的篇幅介绍VC++6.0的应用编程。

本书由教学经验丰富的一线教师编写，其中第1至16章由莆田学院邹金安副教授编写，第17章由福建师范大学福清分校王小峰老师、赵少卡老师编写。

本书适合应用型本科的教学使用。

莆田学院杨剑炉老师、王明昊老师和厦门大学出版社眭蔚编辑对本书提出了许多宝贵意见，莆田学院计算机专业2006级学生程锋和武红飞两位同学参与了资料整理和书中全部实例程序的调试验证，程锋同学和2007级学生蓝玉燕等多位同学帮助本书的录入与排版，在此表示衷心的感谢！

内容概要

《面向对象程序设计与Visual C++ 6.0教程》主要讲述了面向对象程序设计（Object—Oriented Programming, OOP），指的是一种程序开发的方法论。它将对象作为程序的基本单元，将操作和数据封装其中，以提高软件的重用性、灵活性和扩展性。目前已经证实面向对象程序设计加强了程序的灵活性、重用性和可维护性，并且在大型项目设计中广为应用。

面向对象程序设计能够让人们更简单地设计并维护程序，使得程序更加便于分析、设计和理解。

OOP有三个特性：一是封装性。

定义了类，封装了数据和操作的抽象数据类型。

在OOP语言中，类是模块、封装和数据抽象的基础。

二是继承性。

从已存在的类型中继承元素（属性和方法），改变或扩展旧类型的方法。

三是多态性。

它允许使用相同的方法操作不同类型的对象（通常是子类对象），使得类的可用性进一步提高，程序也因此更容易维护和扩展。

C++是最典型的面向对象程序设计语言，能全面反映OOP的特点，而且C++和JaVa是目前和今后相当长的一段时间里被最为广泛使用的语言，又因为掌握了C++，再自学Java比较容易，所以《面向对象程序设计与Visual C++ 6.0教程》借助C++来介绍OOP方法。

但是，仅学C++是不够的，不能很好地进行应用编程，而且visual C++6.0是一个全面的应用程序开发环境，它为程序开发人员提供了使用面向对象的C++语言来开发windows应用程序的强大平台，目前应用相当广泛，所以《面向对象程序设计与Visual C++ 6.0教程》用一定的篇幅介绍VC++6.0的应用编程。

书籍目录

前言第1章 Visual C++集成开发环境1.1 Visual C++概述1.1.1 Visual C++介绍1.1.2 Visual C++安装流程1.1.3 集成环境窗口介绍1.1.4 MSDN帮助系统1.2 项目开发区1.2.1 ClassView (类视图) 1.2.2 ResourceView (资源视图) 1.2.3 FileView (文件视图) 1.3 菜单栏和工具栏1.3.1 菜单栏1.3.2 工具栏本章小结习题第2章 程序设计概述2.1 程序设计流程2.2 结构化程序设计2.2.1 描述任何实体的操作序列只需要三种基本控制结构2.2.2 程序设计中的各个过程体和组成部分应以模块表示2.2.3 过程化程序设计方法2.3 对象化程序设计本章小结习题第3章 C++语言基础3.1 向量3.2 函数3.2.1 内联函数3.2.2 函数重载3.3 指针3.3.1 const指针3.3.2 函数指针3.4 引用3.5 名空间3.5.1 名空间的定义3.5.2 名空间成员的访问3.5.3 名空间的应用3.6 预编译3.6.1 #include指令3.6.2 条件编译指令3.6.3 头文件卫士3.6.4 #define指令3.7 this指针本章小结习题第4章 类4.1 类的定义4.1.1 结构体与类4.1.2 定义类4.1.3 定义对象4.2 类的成员函数4.3 静态成员4.3.1 静态数据成员4.3.2 静态成员函数4.4 友员4.4.1 友员函数4.4.2 友员类4.5 运算符重载4.6 类的设计4.6.1 数据成员设计4.6.2 成员函数设计4.6.3 案例解析本章小结习题第5章 对象5.1 构造函数5.1.1 缺省参数的构造函数5.1.2 构造函数的重载5.2 拷贝构造函数5.2.1 默认拷贝构造函数5.2.2 自定义拷贝函数5.3 析构函数5.4 构造顺序5.4.1 静态对象只被构造一次5.4.2 所有全局对象都在主函数main ()之前被构造5.4.3 全局对象相关构造时无特殊顺序5.5 案例解析本章小结习题第6章 继承6.1 继承和派生的概念6.2 派生类6.2.1 派生类对象结构6.2.2 派生类的声明6.2.3 派生类的构造6.3 继承层次中对象间的赋值6.4 继承方式6.5 继承与组合6.6 多重继承6.6.1 什么是多重继承6.6.2 虚拟继承6.6.3 多种继承的构造顺序6.7 案例解析本章小结习题第7章 面向对象程序设计的方法与步骤7.1 问题描述7.2 过程化分析7.3 基于对象的分析7.4 基于对象的解决方案本章小结习题第8章 多态与抽象类8.1 派生类同化问题8.2 多态与虚函数8.3 抽象类8.4 案例解析本章小结习题第9章 模板9.1 模板9.2 函数模板9.2.1 函数模板的定义9.2.2 函数模板的实现9.3 函数模板的参数9.3.1 形参类型9.3.2 类型匹配9.4 类模板9.4.1 类模板的定义9.4.2 模板类的实现9.5 案例解析本章小结习题第10章 异常10.1 异常处理10.1.1 错误种类10.1.2 异常处理三部曲10.2 异常捕捉10.2.1 类型匹配10.2.2 捕捉异常10.3 异常申述10.3.1 异常抛掷声明10.3.2 异常终止函数10.4 案例分析本章小结习题第11章 I/O流11.1 标准I/O流11.1.1 输入流.....第12章 创建应用程序框架第13章 MFC原理简介第14章 对话框编程第15章 文档与视图编程第16章 高级应用程序编程第17章 综合应用案例

章节摘录

对象是系统中用来描述客观事物的一个实体，是构成系统的一个基本单位，由一组属性和对这些属性进行操作的一组服务构成。

从一般意义上讲，现实世界中的任何事物都可以称为对象。

人类认识现实世界并不是逐个地认识和描述每一个对象实体，而是通过抽象，把具有共同特征的对象归结为一类，形成一般概念。

众所周知，在计算机中不同的数据类型分配的存储空间是不同的。

类是一种用户自定义的类型，可以比较简单，也可以很复杂。

当声明一个类对象时编译程序需要为对象分配存储空间，进行必要的初始化。

对象可以作为函数参数传递吗？

按照传递的语义，相当于定义一个形参实体，用实参给形参初始化。

以一个已经存在的对象给对象创建作依据，可能会带来意想不到的复杂，因为会出现指针或引用值也参与拷贝的现象。

如此，拷贝者与被拷贝者（创建的对象）会同时拥有指针指向的实体，会产生结果的不确定性。

所以对象传递问题若不解决，类机制就形同虚设。

当一个对象终结时，对象自己不能随意撤销，应该先卸下自己拥有的资源，融入资源池，再生而复用，否则，内存泄漏不可避免。

所以，对象的善后工作一定要处理好。

学习重点：
· 掌握构造函数的定义及其运行机制；
· 理解并掌握拷贝构造函数的定义及其运行机制；
· 掌握析构函数的定义及其运行机制；
· 熟悉对象的构造顺序。

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>