

<<面向对象程序设计案例教程>>

图书基本信息

书名：<<面向对象程序设计案例教程>>

13位ISBN编号：9787111314370

10位ISBN编号：7111314379

出版时间：2010-10

出版时间：刘恒洋、杨宏雨 机械工业出版社 (2010-10出版)

作者：刘恒洋，杨宏雨 著

页数：254

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>

<<面向对象程序设计案例教程>>

前言

C++语言是美国贝尔实验室于20世纪80年代在C语言的基础上开发出来的，虽然在它之后又出现了Java和C#之类的新语言，但至今它仍是人们进行中型和大型软件开发项目的最佳选择之一。

本书不同于其他C++语言的教材，而是另辟蹊径，从一个实际的案例“基于TCP/IP的点对点网络通信系统”出发，结合每一章的知识点，手把手地讲解如何利用学到的知识构架系统。

这样不仅让学生在学C++语言的语法的同时保持极大的兴趣，还能对真正地利用C++语言进行大型软件开发作一个有效的尝试，为学生今后从事开发工作打下良好的基础。

本书所用案例涉及网络协议、数据库、多线程等知识，所以要求读者对计算机网络、数据库、操作系统的基本知识有所了解。

本书主要讲述C++面向对象及高级应用，所以面向的是有C或其他语言编程基础的读者。

在组织结构上本书采用案例贯通式，以案例为主线，每一章的案例实现都是建立在前面章节的基础上的。

建议读者从头阅读，如果只选取其中的一两章进行阅读，则不能很好地理解案例的实现过程。

本书章节分配如下：第1章初步认识C++和实例系统。

本章首先介绍C++语言的特点，接着介绍本书实例系统的总体结构和功能，重点给出实例系统中的协议设计，最后建立服务器项目，为后面章节的实例系统开发做好准备。

第2章类和对象及其应用。

本章介绍C++中类和对象的定义语法，利用这些知识来设计实例系统中的核心基础类：消息实体类和在线用户信息类。

第3章C++多态性的定义及其应用。

本章主要介绍C++中继承的定义和多态性的实现，以及如何应用，即设计实例系统中数据库操作类的层次结构。

第4章C++的输入/输出流及其应用。

本章详细介绍iostream库在标准输入/输出和文件的输入/输出两个方面的应用及相关语法，并在此基础上为实例系统设计能够读写配置文件信息的输入/输出类。

第5章Windows程序设计基础。

本章介绍Windows应用程序的基本框架构成和运行机制。

第6章MFC应用程序基础。

本章以实例系统中客户端的建立为例，介绍如何利用MFC和Visual C++的应用程序向导（Application Wizard）编写Windows应用程序，以及MFC应用程序中各种标准控件的使用。

第7章MFC数据库编程及其应用。

本章介绍ODBC的概念和简单使用方法，并在此基础上设计实例系统数据库，完善实例系统中服务器端的数据库操作类。

第8章Windows多线程应用程序及其应用。

本章主要介绍Windows多线程的概念，以及线程之间的同步与通信，设计实例系统中的服务器多线程类和客户端多线程类。

第9章Winsock编程及其应用。

本章主要介绍在Win32平台上基于Winsock的TCP/IP编程基础知识，实现实例系统中网络通信的相关功能，完成整个实例系统。

为了更好地配合任课教师的教学，帮助学生解决学习中的困惑，本书还提供了配套的电子教案。

如果需要，可以登录华章网站（www.hzbook.com）或通过邮件索取。

本书由重庆理工大学的刘恒洋、杨宏雨、王勇、王柯柯、刘洁、黄同愿和但远宏老师执笔编写，同时本书得到了重庆理工大学教材出版基金资助。

尽管我们在写作过程中投入了大量的时间和精力，但由于水平有限，错误和不足仍在所难免，敬请读者批评指正。

<<面向对象程序设计案例教程>>

内容概要

《面向对象程序设计案例教程》以VC 6.0为开发平台，以一个完整的实例系统的开发过程为主线，将C++基本知识点和Windows程序设计的基础知识融入到实例系统的不同功能模块中去，并最终完成实例系统所有功能的代码。

本书的主要内容为：第1~4章主要介绍C++的基础语法知识，这部分的程序设计不涉及任何界面编程，主要是利用C++进行实例系统底层类的封装和定义；第5~6章主要介绍MFC对Windows程序设计的封装和利用MFC进行Windows程序设计的方法，并完成实例系统中与界面相关的功能；第7~9章主要介绍Windows程序设计高级应用中的数据库编程、多线程编程以及利用Winsock进行网络通信编程这三个方面的知识，同时完成实例系统中比较核心的三大功能：数据库操作、多线程通信与同步以及底层网络通信。

本书不仅仅介绍C++中比较核心和经常使用的基本语法，而且把C++编程和Windows编程有机结合起来，使读者不再孤立地用C++进行枯燥的没有界面的编程。

此外，本书把所有的知识点都与实例系统中的一个相关功能关联起来，使得读者在学习过程中充满趣味性。

本书适合有一定的C语言编程基础的读者进行C++编程入门和进阶。

<<面向对象程序设计案例教程>>

书籍目录

前言教学建议第1章 初步认识c++和实例系统1.1 c++语言概述1.1.1 c++语言的特点1.1.2 类和对象的概念1.2 实例系统概述1.2.1 实例系统的总体功能1.2.2 实例系统的总体结构1.2.3 实例系统的协议设计1.3 实例系统开发前的准备1.3.1 创建服务器项目1.3.2 定义公共头文件1.3.3 定义协议1.4 本章小结1.5 思考与练习第2章 类和对象及其应用2.1 c++基础语法2.1.1 new和delete2.1.2 const限定修饰符2.1.3 引用2.2 类和对象2.2.1 类和对象的定义2.2.2 类成员的定义和访问2.2.3 类的构造函数和析构函数2.2.4 操作符重载2.2.5 类模板和标准模板库2.3 消息实体类的设计和实现2.3.1 消息实体类的设计2.3.2 消息实体类的实现2.4 在线用户信息类的设计2.4.1 在线用户信息类的定义2.4.2 在线用户信息类的实现2.5 本章小结2.6 思考与练习第3章 c++多态性的定义及其应用3.1 基类和派生类3.1.1 基类和派生类的定义3.1.2 基类和派生类成员的定义及其访问3.2 基类和派生类的构造及其析构3.2.1 基类和派生类的构造函数3.2.2 基类和派生类的析构函数3.3 多态性在c++中的体现3.3.1 静态多态性3.3.2 动态多态性3.4 实例系统数据库操作类的设计3.5 本章小结3.6 思考与练习第4章 c++的输入/输出流及其应用4.1 c++输入/输出流概述4.1.1 c++的输入/输出流类库4.1.2 c++对流文件的操作4.2]和[[输入/输出操作符4.2.1]和[[用于标准输入/输出4.2.2]和[[用于文件输入/输出4.2.3]和[[操作符的重载4.3 get () 和put () 输入/输出函数4.3.1 get () 和put () 用于标准输入/输出4.3.2 get () 和put () 用于文件输入/输出4.4 read () 和write () 输入/输出函数4.4.1 read () 和write () 用于标准输入/输出4.4.2 read () 和write () 用于文件输入/输出4.5 格式控制4.5.1 布尔变量的输出操作控制4.5.2 带进制整数的输出操作控制4.5.3 浮点数输出操作控制4.5.4 输入操作控制4.5.5 刷新缓冲区操作控制4.5.6 字符设置操作控制4.6 实例系统输入/输出类的设计4.6.1 输入/输出需求4.6.2 输入/输出类的实现4.7 本章小结4.8 思考与练习第5章 windows程序设计基础5.1 概述5.1.1 窗口5.1.2 事件驱动5.1.3 句柄和windows消息5.2 最简单的windows应用程序5.2.1 最简单的windows应用程序的组成5.2.2 最简单的windows应用程序示例5.3 本章小结5.4 思考与练习第6章 mfc应用程序基础6.1 应用程序向导6.2 mfc应用程序中常用windows标准控件的应用6.2.1 按钮控件及其应用6.2.2 列表框控件及其应用6.2.3 编辑框控件及其应用6.2.4 组合框控件及其应用6.2.5 对话框通用控件及其应用6.3 基于对话框的mfc应用程序在实例系统中的应用6.3.1 登录对话框的设计与实现6.3.2 添加全局变量和全局函数6.3.3 注册对话框的设计与实现6.3.4 主界面对话框的设计与实现6.3.5 查看联系人信息对话框的设计与实现6.3.6 查找联系人类型对话框的设计与实现6.3.7 查找联系人结果对话框的设计与实现6.3.8 文本聊天对话框的设计与实现6.3.9 为实例系统添加自定义消息6.4 本章小结6.5 思考与练习第7章 mfc数据库编程及其应用7.1 数据库系统的基本概念7.2 odbc基本概念7.3 mfc的odbc类简介7.3.1 cdatabase类7.3.2 crecordset类7.4 设计实例系统数据库7.5 注册数据源7.6 设计实例系统数据库操作类7.7 本章小结7.8 思考与练习第8章 windows多线程应用程序及其应用8.1 进程和线程8.1.1 进程和线程的概念8.1.2 线程的创建和终止8.2 线程同步8.2.1 事件对象8.2.2 临界段8.2.3 互斥体8.2.4 信号计数器8.3 线程通信8.3.1 利用全局对象进行线程通信8.3.2 利用自定义消息进行线程通信8.4 设计服务器多线程类8.4.1 服务器线程工作流程8.4.2 服务器线程类的实现8.5 设计客户端多线程类8.5.1 客户端线程工作流程8.5.2 客户端线程类的实现8.6 本章小结8.7 思考与练习第9章 winsock编程及其应用9.1 网际协议9.1.1 tcp和udp9.1.2 定址9.1.3 套接字9.2 winsock基础9.2.1 winsock的初始化9.2.2 错误检查和控制9.2.3 面向连接的协议9.2.4 无连接协议9.3 利用winsock完善实例系统9.3.1 收发消息类的封装9.3.2 完善服务器收发消息线程9.3.3 完善服务器消息处理线程9.3.4 完善服务器主函数9.3.5 完善客户端收发消息线程9.4 本章小结9.5 思考与练习参考文献

<<面向对象程序设计案例教程>>

章节摘录

插图：c++是一门抽象性的面向对象的编程语言，本章将从封装性、继承性和多态性这三个方面来讲述C++语言的特点；同时本章还将从协议、需求和开发工具这三个方面来阐述贯穿本书的实例系统，让读者对实例系统有一个大概的了解；最后本章将为后续的实例系统的开发做一些准备。

1.1 C++语言概述 c++语言是美国贝尔实验室于20世纪80年代在c语言的基础上开发出来的，它保留了c语言的全部优点，增加了面向对象的机制，支持面向对象程序设计方法。

这使得C++语言比较适合于中型和大型的软件开发项目。

1.1.1 C++语言的特点 c++语言作为一种面向对象程序设计语言，具有封装性、继承性和多态性三大特性。

1.封装性封装是面向对象方法的一个重要原则，即将对象的属性和行为代码封装在对象的内部，形成一个独立的单位，并尽可能隐蔽对象的内部细节。

c++面向对象的封装特性包含两层含义：第一层含义是将对象的全部属性和行为封装在对象的内部，形成一个不可分割的独立单位，对象的属性值只能由这个对象的行为来读取和修改；第二层含义是“信息隐蔽”，即尽可能隐蔽对象的内部细节，对外形成一道屏障，只保留有限的对外接口与外部发生联系。

面向对象程序设计方法的信息隐蔽作用体现了自然界中事物的相对独立性，程序设计者与使用者只需关心其对外提供的接口，而不必过分注意其内部细节。

封装性使对象以外的事物不能随意获取对象的内部属性，从而有效地避免了外部的错误访问对它产生的影响，大大减轻了软件开发过程中查错的工作量，有效地减小了排错的难度。

c++一般用类来实现封装性，并通过设置对数据的访问权限来控制对内部数据的访问，即将类的部分成员作为类的外部接口，将其他成员隐藏起来，实现对数据成员的合理控制，使程序的不同部分之间的相互影响尽可能降到最小。

2.继承性客观事物既有共性，又有个性，如果只考虑事物的共性，而不考虑事物的个性，就不能真实地反映出客观世界中事物之间的层次关系。

面向对象中的抽象性意味着舍弃对象的次要特性，提取事物的本质共性，形成类。

如果在类的基础上，根据实际目标的需要合理考虑一部分对象的个性特征，则可形成一个新类。

<<面向对象程序设计案例教程>>

编辑推荐

<<面向对象程序设计案例教程>>

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>