

<<C++语言程序设计>>

图书基本信息

书名：<<C++语言程序设计>>

13位ISBN编号：9787302227984

10位ISBN编号：7302227985

出版时间：2010-7

出版时间：清华大学出版社

作者：郑莉，董江舟 编著

页数：529

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++语言程序设计>>

前言

一、版本说明本书第1版于1999年出版，第2版于2001年出版，第3版于2003年出版。第4版是在前三版的基础上，广泛听取了读者和同行的建议，参考了最新的资料，并根据作者在授课过程中的经验而形成的。

第4版主要修改内容如下。

- (1) 删除了原来的第13章。
- (2) 在第2 - 12章中，每章增加了一节“深度探索”，作为选学内容。
- (3) 编写了贯穿第4 - 10章的新的综合实例“个人银行账户管理程序”。
- (4) 重新编写了第10章。

二、本书的编写背景C++语言是从C语言发展演变而来的一种面向对象的程序设计语言。

C++语言的主要特点表现在两个方面：一是兼容C，二是支持面向对象的方法。

面向对象的程序设计（OOP）方法将数据及对数据的操作方法封装在一起，作为一个相互依存、不可分离的整体——对象。

对同类型对象抽象出其共性，形成类。

类中的大多数数据，只能用本类的方法进行处理。

类通过一个简单的外部接口，与外界发生关系，对象与对象之间通过消息进行通信。

这样，程序模块间的关系简单，程序模块的独立性、数据的安全性具有良好的保障，通过继承与多态性，使程序具有很高的可重用性，使得软件的开发和维护都更为方便。

<<C++语言程序设计>>

内容概要

本书针对程序设计的初学者，以面向对象的程序设计思想为主线，以通俗易懂的方法介绍C++语言，引导读者以最自然的方式，将人类习惯的面向对象的思维方法运用到程序设计中。主要内容包括程序设计基础知识、类与对象的基本概念、继承与多态、输入输出流，以及泛型程序设计。

此外，本教材还介绍了一些常用数据结构基础知识，使得读者学习本书后，能够解决一些简单的实际问题。

整套教材语言生动、流畅，深入浅出。

适用于各类学校的C++语言程序设计课程。

<<C++语言程序设计>>

书籍目录

第1章 绪论1	1.1 计算机程序设计语言的发展1	1.1.1 机器语言与汇编语言1	1.1.2 高级语言2
	1.1.3 面向对象的语言2	1.2 面向对象的方法2	1.2.1 面向对象方法的由来3
	1.2.2 面向对象的基本概念4	1.3 面向对象的软件开发5	1.3.1 分析5
	1.3.2 设计5	1.3.3 编程6	1.3.4 测试6
	1.3.5 维护6	1.4 信息的表示与存储6	1.4.1 计算机的数字系统6
	1.4.2 几种进位记数制之间的转换8	1.4.3 信息的存储单位10	1.4.4 二进制数的编码表示11
	1.4.5 定点数和浮点数15	1.4.6 数的表示范围15	1.4.7 非数值信息的表示16
	1.5 程序开发的基本概念16	1.5.1 基本术语16	1.5.2 完整的程序过程17
	1.6 小结18	习题18	第2章 C++简单程序设计19
	2.1 C++语言概述19	2.1.1 C++的产生19	2.1.2 C++的特点20
	2.1.3 C++程序实例20	2.1.4 字符集21	2.1.5 词法记号22
	2.2 基本数据类型和表达式23	2.2.1 基本数据类型24	2.2.2 常量25
	2.2.3 变量27	2.2.4 符号常量28	2.2.5 运算符与表达式28
	2.2.6 语句37	2.3 数据的输入与输出37	2.3.1 I/O流37
	2.3.2 预定义的插入符和提取符37	2.3.3 简单的I/O格式控制38	2.4 算法的基本控制结构39
	2.4.1 用if语句实现选择结构39	2.4.2 多重选择结构41	2.4.3 循环结构44
	2.4.4 循环结构与选择结构的嵌套49	2.4.5 其他控制语句52	2.5 自定义数据类型52
	2.5.1 typedef声明53	2.5.2 枚举类型enum53	2.6 深度探索55
	2.6.1 变量的实现机制55	2.6.2 C++表达式的执行原理58	2.7 小结60
	习题60	第3章 函数64	3.1 函数的定义与使用64
	3.1.1 函数的定义64	3.1.2 函数的调用65	3.1.3 函数的参数传递78
	3.2 内联函数81	3.3 带默认形参值的函数82	3.4 函数重载84
	3.5 C++系统函数86	3.6 深度探索89	3.6.1 运行栈与函数调用的执行89
	3.6.2 函数声明与类型安全94	3.7 小结95	习题96
	第4章 类与对象98	4.1 面向对象程序设计的基本特点98	4.1.1 抽象98
	4.1.2 封装99	4.1.3 继承100	4.1.4 多态100
	4.2 类和对象100	4.2.1 类的定义101	4.2.2 类成员的访问控制102
	4.2.3 对象103	4.2.4 类的成员函数104	4.2.5 程序实例106
	4.3 构造函数和析构函数107	4.3.1 构造函数107	4.3.2 复制构造函数109
	4.3.3 析构函数113	4.3.4 程序实例114	4.4 类的组合116
	4.4.1 组合116	4.4.2 前向引用声明120	4.5 UML图形标识122
	4.5.1 UML简介122	4.5.2 UML类图123	4.6 结构体和联合体129
	4.6.1 结构体129	4.6.2 联合体130	4.7 综合实例--个人银行账户管理程序133
	4.7.1 类的设计133	4.7.2 源程序及说明134	4.8 深度探索137
	4.8.1 位域137	4.8.2 用构造函数定义类型转换139	4.8.3 对象作为函数参数和返回值的传递方式140
	4.9 小结143	习题144	第5章 数据的共享与保护146
	5.1 标识符的作用域与可见性146	5.1.1 作用域146	5.1.2 可见性150
	5.2 对象的生存期150	5.2.1 静态生存期150	5.2.2 动态生存期151
	5.3 类的静态成员153	5.3.1 静态数据成员154	5.3.2 静态函数成员156
	5.4 类的友元158	5.4.1 友元函数160	5.4.2 友元类161
	5.5 共享数据的保护163	5.5.1 常对象163	5.5.2 用const修饰的类成员164
	5.5.3 常引用166	5.6 多文件结构和编译预处理命令168	5.6.1 C++程序的一般组织结构168
	5.6.2 外部变量与外部函数170	5.6.3 标准C++库172	5.6.4 编译预处理173
	5.7 综合实例--个人银行账户管理程序176	5.8 深度探索180	5.8.1 常成员函数的声明原则180
	5.8.2 代码的编译连接与执行过程182	5.9 小结185	习题185
	第6章 数组、指针与字符串188	6.1 数组188	6.1.1 数组的声明与使用188
	6.1.2 数组的存储与初始化190	6.1.3 数组作为函数参数193	6.1.4 对象数组194
	6.1.5 程序实例196	6.2 指针199	6.2.1 内存空间的访问方式199
	6.2.2 指针变量的声明200	6.2.3 与地址相关的运算“?”和“&”200	6.2.4 指针的赋值201
	6.2.5 指针运算203	6.2.6 用指针处理数组元素205	6.2.7 指针数组206
	6.2.8 用指针作为函数参数209	6.2.9 指针型函数210	6.2.10 指向函数的指针211
	6.2.11 对象指针213	6.3 动态内存分配218	6.4 用vector创建数组对象224
	6.5 深复制与浅复制226	6.6 字符串229	6.6.1 用字符数组存储和处理字符串230
	6.6.2 string类230	6.7 综合实例--个人银行账户管理程序234	6.8 深度探索241
	6.8.1 指针与引用241	6.8.2 指针的安全性隐患及其应对方案243	6.8.3 const_cast的应用246
	6.9 小结248	习题248	第7章 继承与派生251
	7.1 类的继承与派生251	7.1.1 继承关系举例251	7.1.2 派生类的定义252
	7.1.3 派生类生成过程254	7.2 访问控制256	7.2.1 公有继承256
	7.2.2 私有继承258	7.2.3 保护继承260	7.3 类型兼容规则262
	7.4 派生类的构造和析构函数264	7.4.1 构造函数265	7.4.2 复制构造函数267
	7.4.3 析构函数268	7.5 派生类成员的标识与访问270	7.5.1 作用域分辨符270
	7.5.2		

<<C++语言程序设计>>

虚基类276 7.5.3 虚基类及其派生类构造函数278 7.6 程序实例--用高斯消去法解线性方程组279
 7.6.1 算法基本原理279 7.6.2 程序设计分析280 7.6.3 源程序及说明281 7.6.4 运行结果与分析286
 7.7 综合实例--个人银行账户管理程序287 7.7.1 问题的提出287 7.7.2 类设计288 7.7.3 源程序及说明289
 7.7.4 运行结果与分析294 7.8 深度探索295 7.8.1 组合与继承295 7.8.2 派生类对象的内存布局298
 7.8.3 基类向派生类的转换及其安全性问题301 7.9 小结303 习题304第8章 多态性306
 8.1 多态性概述306 8.1.1 多态的类型306 8.1.2 多态的实现306 8.2 运算符重载307
 8.2.1 运算符重载的规则308 8.2.2 运算符重载为成员函数309 8.2.3 运算符重载为非成员函数313
 8.3 虚函数315 8.3.1 一般虚函数成员316 8.3.2 虚析构函数319 8.4 纯虚函数与抽象类321
 8.4.1 纯虚函数321 8.4.2 抽象类322 8.5 程序实例--变步长梯形积分算法求解函数的定积分324
 8.5.1 算法基本原理324 8.5.2 程序设计分析325 8.5.3 源程序及说明326 8.5.4 运行结果与分析329
 8.6 综合实例--对个人银行账户管理程序的改进329 8.7 深度探索335 8.7.1 多态类型与非多态类型335
 8.7.2 运行时类型识别337 8.7.3 虚函数动态绑定的实现原理340 8.8 小结343 习题344第9章 群体类和群体数据的组织346
 9.1 函数模板与类模板347 9.1.1 函数模板347 9.1.2 类模板350 9.2 线性群体353 9.2.1 线性群体的概念353 9.2.2 直接访问群体--数组类355 9.2.3 顺序访问群体--链表类363
 9.2.4 栈类368 9.2.5 队列类374 9.3 群体数据的组织377 9.3.1 插入排序377 9.3.2 选择排序379
 9.3.3 交换排序380 9.3.4 顺序查找381 9.3.5 折半查找381 9.4 综合实例--对个人银行账户管理程序的改进382
 9.5 深度探索385 9.5.1 模板的实例化机制385 9.5.2 为模板定义特殊的实现388 9.5.3 模板元编程简介392 9.6 小结395 习题395第10章 泛型程序设计
 与C++标准模板库398 10.1 泛型程序设计及STL的结构398 10.1.1 泛型程序设计的基本概念398 10.1.2 STL简介399
 10.2 迭代器402 10.2.1 输入流迭代器和输出流迭代器403 10.2.2 迭代器的分类405 10.2.3 迭代器的区间407
 10.2.4 迭代器的辅助函数409 10.3 容器410 10.3.1 容器的基本功能与分类410 10.3.2 顺序容器412
 10.3.3 关联容器427 10.4 函数对象435 10.4.1 函数对象基本概念及分类436 10.4.2 函数适配器441
 10.5 算法447 10.5.1 STL算法基础448 10.5.2 不可变序列算法448 10.5.3 可变序列算法451
 10.5.4 排序和搜索算法455 10.5.5 数值算法459 10.6 综合实例--对个人银行账户管理程序的改进461
 10.7 深度探索467 10.7.1 swap467 10.7.2 STL组件的类型特征与STL的扩展469 10.7.3 Boost简介476
 10.8 小结478 习题478第11章 流类库与输入输出481 11.1 I/O流的概念及流类库结构481
 11.2 输出流483 11.2.1 构造输出流对象483 11.2.2 使用插入运算符和操纵符484 11.2.3 文件输出流成员函数488
 11.2.4 二进制输出文件490 11.2.5 字符串输出流491 11.3 输入流492 11.3.1 构造输入流对象492
 11.3.2 使用提取运算符493 11.3.3 输入流操纵符493 11.3.4 输入流相关函数493 11.3.5 字符串输入流497
 11.4 输入输出流498 11.5 综合实例--对个人银行账户管理程序的改进498 11.6 深度探索503
 11.6.1 宽字符、宽字符串与宽流503 11.6.2 对象的串行化507 11.7 小结510 习题510第12章 异常处理512
 12.1 异常处理的基本思想512 12.2 C++异常处理的实现512 12.2.1 异常处理的语法513 12.2.2 异常接口声明515
 12.3 异常处理中的构造与析构516 12.4 标准程序库异常处理517 12.5 综合实例--对个人银行账户管理程序的改进520
 12.6 深度探索523 12.6.1 异常安全性问题523 12.6.2 避免异常发生时的资源泄漏525 12.7 小结528 习题529

章节摘录

插图：下面简单介绍一下面向对象方法中的几个基本概念。

当然我们不能期望通过几句话的简单介绍就完全理解这些概念，在本书的后续章节中，会不断帮助读者加深对这些概念的理解，以达到熟练运用的目的。

1.对象从一般意义上讲，对象是现实世界中一个实际存在的事物，它可以是有形的（比如一辆汽车），也可以是无形的（比如一项计划）。

对象是构成世界的一个独立单位，它有自己的静态特征（可以用某种数据来描述）和动态特征（对象所表现的行为或具有的功能）。

面向对象方法中的对象，是系统中用来描述客观事物的一个实体，它是用来构成系统的一个基本单位。

对象由一组属性和一组行为构成。

属性是用来描述对象静态特征的数据项，行为是用来描述对象动态特征的操作序列。

2.类把众多的事物归纳、划分成一些类，是人类在认识客观世界时经常采用的思维方法。

分类所依据的原则是抽象，即忽略事物的非本质特征，只注意那些与当前目标有关的本质特征，从而找出事物的共性，把具有共同性质的事物划分为一类，得出一个抽象的概念。

例如，石头、树木、汽车、房屋等都是人们在长期的生产和生活实践中抽象出的概念。

面向对象方法中的“类”，是具有相同属性和服务的一组对象的集合。

它为属于该类的全部对象提供了抽象的描述，其内部包括属性和行为两个主要部分。

类与对象的关系犹如模具与铸件之间的关系，一个属于某类的对象称为该类的一个实例。

3.封装封装是面向对象方法的一个重要原则，就是把对象的属性和服务结合成一个独立的系统单位，并尽可能隐蔽对象的内部细节。

这里有两个含义：第一个含义是把对象的全部属性和全部服务结合在一起，形成一个不可分割的独立单位。

第二个含义也称做“信息隐蔽”，即尽可能隐蔽对象的内部细节，对外形成一个边界（或者说一道屏障），只保留有限的对外接口使之与外部发生联系。

<<C++语言程序设计>>

编辑推荐

《C++语言程序设计(第4版)》：清华大学计算机系列教材,普通高等教育“十一五”国家级规划教材,国家精品课程教材

<<C++语言程序设计>>

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>