

21 世纪高等学校计算机教育实用规划教材

面向对象程序设计教程(C++语言描述)

题解与课程设计指导

马石安
魏文平 编著



清华大学出版社

21

世纪高等学校计算机教育实用规划教材

面向对象程序设计教程(C++语言描述) 题解与课程设计指导

马石安 魏文平 编著



清华大学出版社
北京

内 容 简 介

本书是《面向对象程序设计教程(C++语言描述)》(简称主教材)一书的配套辅助教材,其内容包括3部分:第1部分是主教材中的全部习题和参考解答;第2部分是主教材中全部实验题和参考解答;第3部分是课程设计指导,该部分首先介绍课程设计的目标,然后通过几个综合实例,循序渐进地启发学生完成设计。

本书可作为高等院校计算机及相关专业学习面向对象程序设计和C++语言程序设计的辅助教材,也可供自学者或教师参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

面向对象程序设计教程(C++语言描述)题解与课程设计指导/马石安,魏文平编著. —北京:清华大学出版社,2008.10

(21世纪高等学校计算机教育实用规划教材)

ISBN 978-7-302-17955-9

I. 面… II. ①马… ②魏… III. C语言—程序设计—高等学校—教学参考资料
IV. TP312

中国版本图书馆CIP数据核字(2008)第093035号

责任编辑:魏江江 李玮琪

责任校对:时翠兰

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:12.5 字 数:297千字

版 次:2008年10月第1版 印 次:2008年10月第1次印刷

印 数:1~3000

定 价:19.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:028848-01

读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室 计算机与信息分社营销室 收

邮编：100084

电子邮件：jsjc@tup.tsinghua.edu.cn

电话：010-62770175-4608/4409

邮购电话：010-62786544

教材名称：面向对象程序设计教程(C++语言描述)题解与课程设计指导

ISBN 978-7-302-17955-9

个人资料

姓名：_____ 年龄：_____ 所在院校/专业：_____

文化程度：_____ 通信地址：_____

联系电话：_____ 电子信箱：_____

您使用本书是作为：☐指定教材 ☐选用教材 ☐辅导教材 ☐自学教材

您对本书封面设计的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书印刷质量的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书的总体满意度：

从语言质量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

从科技含量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

本书最令您满意的是：

☐指导明确 ☐内容充实 ☐讲解详尽 ☐实例丰富

您认为本书在哪些地方应进行修改？（可附页）

您希望本书在哪些方面进行改进？（可附页）

出版说明

随着我国高等教育规模的扩大以及产业结构调整的进一步完善,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和建议下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设仍然把重点放在公共基础课和

专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材要配套,同一门课程可以有多种具有不同内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会

联系人:丁岭 dingl@tup.tsinghua.edu.cn

前 言

为了给面向对象程序设计初学者提供一本清晰的入门教材,作者于2007年8月编写了《面向对象程序设计教程(C++语言描述)》(2007年8月由清华大学出版社出版),受到师生们的欢迎,同时读者希望能有相应的习题解答便于教学和学习。为此,笔者组织编写了这本辅助教材。本书在解析主教材全部习题和实验题的基础上,增添了课程设计指导部分,使读者对面向对象程序设计方法在程序开发中的应用有一个全局性的了解。

本书具有以下特色和价值:

1. 与主教材紧密结合

把习题、实验、课程设计与主教材作为学好面向对象程序设计课程的有机组成部分,多位一体,互为补充。

2. 不同习题,不同对待

为了帮助读者更好地理解程序,对于稍难的习题,书中都作了比较详细的说明,或在程序中加了注释。

3. 循序渐进地启发学生完成课程设计

课程设计要求、设计思想和源程序分开,这样既防止作为标准答案而限制读者的创造性思维,也防止初学者由于缺少经验而走过多的弯路。

4. 课程设计实例简单、实用

课程设计实例主要是将主教材中的实例结合实际应用完成的,可以让读者集中精力学会分析问题和解决问题,而不是停留在某个知识点上。

本书由马石安和魏文平编写,全书由马石安统一修改、整理和定稿。

在本书的编写过程中,笔者参考和引用了大量书籍和文献资料,在此,向被引用文献的作者表示衷心的感谢,向给予本书帮助的所有人士表示衷心感谢。

由于作者水平有限、加之时间仓促,书中难免存在缺点与疏漏,敬请读者及同行们予以批评指正。作者联系方式: mashian@163.com,也欢迎各位同仁探讨面向对象程序设计教学中的相关问题。

作 者

2008年5月

目 录

第1部分 习题和参考解答

第1章 面向对象程序设计概论	3
第2章 从C到C++	7
第3章 类与对象	12
第4章 继承机制	23
第5章 多态性和虚函数	30
第6章 运算符重载	38
第7章 模板	45
第8章 I/O流类库	49
第9章 异常处理	56

第2部分 实验题和参考解答

实验1 简单的C++程序(2学时)	65
实验2 引用与函数(2学时)	70
实验3 构造函数与析构函数(2学时)	75
实验4 静态成员与友元(4学时)	80
实验5 继承与派生(4学时)	86
实验6 多态性与虚函数(4学时)	96
实验7 运算符重载(2学时)	104
实验8 模板(2学时)	109
实验9 I/O流(2学时)	116
实验10 异常处理(2学时)	121

第3部分 课程设计指导

第1章 课程设计目标	129
1.1 课程设计的目的与要求	129
1.2 考核方式	129
1.3 课程设计报告的主要内容	130
1.4 评价标准	130

第 2 章 课程设计预备知识	132
2.1 建立和运行包含多个文件的程序的方法	132
2.1.1 用户建立项目工作区和项目文件	132
2.1.2 用户只建立项目文件	137
2.2 面向对象的标记图 UML	139
2.2.1 类和对象的 UML 标记图	139
2.2.2 几种关系的图形标识	140
第 3 章 课程设计实例	142
3.1 对象初始化方法演示	142
3.1.1 设计任务	142
3.1.2 设计要求	142
3.1.3 程序的总体结构	142
3.1.4 详细设计	143
3.1.5 程序清单	148
3.1.6 实例输出	154
3.2 小型公司的工资管理	156
3.2.1 设计任务	156
3.2.2 设计要求	157
3.2.3 程序的总体结构	157
3.2.4 详细设计	158
3.2.5 程序清单	160
3.2.6 实例输出	164
3.3 分数计算器	165
3.3.1 设计任务	165
3.3.2 设计要求	165
3.3.3 程序的总体结构	166
3.3.4 详细设计	166
3.3.5 程序清单	170
3.3.6 实例输出	176
3.4 分析表达式	180
3.4.1 设计任务	180
3.4.2 设计要求	180
3.4.3 总体设计	180
3.4.4 详细设计	180
3.4.5 程序清单	182
3.4.6 实例输出	184
参考文献	185

第1部分

习题和参考解答

第1章

面向对象程序设计概论

一、名词解释

抽象 封装 消息

【问题解答】

面向对象方法中的抽象是指对具体问题(对象)进行概括,抽出一类对象的公共性质并加以描述的过程。

面向对象方法中的封装就是把抽象出来的对象的属性和行为结合成一个独立的单位,并尽可能隐蔽对象的内部细节。

消息是面向对象程序设计用来描述对象之间通信的机制。一个消息就是一个对象要求另一个对象实施某种操作的一个请求。

二、填空题

(1) 目前有面向过程的结构化程序设计方法和面向对象的程序设计方法两种重要的程序设计方法。

(2) 结构化程序设计方法中的模块由顺序、选择和循环 3 种基本结构组成。

(3) 在结构化程序设计方法中,程序可表示为程序 = 数据结构 + 算法;而面向对象的程序设计方法,程序可表示为程序 = 对象 + 消息。

(4) 结构化程序设计方法中的基本模块是过程;而面向对象程序设计方法中的基本模块是类。

(5) 面向对象程序设计方法具有抽象性、封装性、继承性和多态性等特点。

三、选择题(至少选一个,可以多选)

(1) 面向对象程序设计着重于(B)的设计。

A. 对象 B. 类 C. 算法 D. 数据

(2) 面向对象程序设计中,把对象的属性和行为组织在同一个模块内的机制叫做(C)。

A. 抽象 B. 继承 C. 封装 D. 多态

(3) 在面向对象程序设计中,类通过(D)与外界发生关系。

A. 对象 B. 类 C. 消息 D. 接口

(4) 面向对象程序设计中,对象与对象之间的通信机制是(C)。

A. 对象 B. 类 C. 消息 D. 接口

(5) 关于 C++ 与 C 语言的关系的描述中,(D)是错误的。

A. C 语言是 C++ 的一个子集 B. C 语言与 C++ 是兼容的
C. C++ 对 C 语言进行了一些改进 D. C++ 和 C 语言都是面向对象的

【结果分析】

C语言是面向过程的。C++语言是一种经过改进的更为优化的C语言,是一种混合型语言,既面向过程也面向对象。

(6) 面向对象的程序设计将数据结构与(A)放在一起,作为一个相互依存、不可分割的整体来处理。

- A. 算法 B. 信息 C. 数据隐藏 D. 数据抽象

(7) 下面(A)不是面向对象系统所包含的要素。

- A. 重载 B. 对象 C. 类 D. 继承

【结果分析】

面向对象=对象+类+继承+消息+多态

(8) 下面说法正确的是(BC)。

- A. 将数据结构和算法置于同一个函数内,即为数据封装
B. 一个类通过继承可以获得另一个类的特性
C. 面向对象要求程序员集中于事物的本质特征,用抽象的观点看待程序
D. 同一消息为不同的对象接受时,产生的行为是一样的,这称为一致性

【结果分析】

面向对象程序设计方法具有抽象性、封装性、继承性和多态性等特点。将数据结构和算法置于同一个类内,即为数据封装。同一消息为不同的对象接受时,产生的行为可能是不一样的,这称为多态性。

(9) 下面说法正确的是(AD)。

- A. 对象是计算机内存中的一块区域,它可以存放代码和数据
B. 对象实际是功能相对独立的一段程序
C. 各个对象间的数据可以共享是对象的一大优点
D. 在面向对象的程序中,对象之间只能通过消息相互通信

【结果分析】

对象是计算机内存中的一块区域。在对象中,不但存有数据,而且存有代码,使得每个对象在功能上相互之间保持相对独立。对象之间存在各种联系,但它们之间只能通过消息进行通信。

四、判断题

(1) 在高级程序设计语言中,一般用类来实现对象,类是具有相同属性和行为的一组对象的集合,它是创建对象的模板。 (✓)

(2) C++语言只支持面向对象技术的抽象性、封装性、继承性等特性,而不支持多态性。 (×)

【结果分析】

C++语言不仅支持面向对象技术的抽象性、封装性、继承性等特性,而且支持多态性。

(3) 面向对象程序设计中的消息应该包含“如何做”的信息。 (×)

【结果分析】

消息是面向对象程序设计用来描述对象之间通信的机制。向对象“发送消息”只需告诉对象做什么,对象根据这个消息决定如何做。

(4) 一个消息只能产生特定的响应效果。 (X)

【结果分析】

当一个对象发出消息时,由于接收对象的类型可能不同,所以,它们可能做出不同的反应。这样,一个消息可以产生不同的响应效果,这种现象叫做多态。

(5) 类的设计和类的继承机制实现了软件模块的可重用性。 (√)

(6) C++语言和Java语言均不是一个纯正的面向对象的程序设计的语言。 (X)

【结果分析】

Java语言是一个纯正的面向对象的程序设计语言。

(7) 学习C++语言是学习面向对象的程序设计方法的唯一途径。 (X)

【结果分析】

程序设计方法是独立于具体程序设计语言的一种技术,学习C++语言是学习面向对象程序设计方法的重要途径之一。

(8) 在C++语言中,类是支持数据封装的工具。 (√)

五、简答题

(1) 什么是结构化程序设计方法? 它有哪些优点和缺点?

【问题解答】

结构化程序设计方法着眼于系统要实现的功能,从系统的输入输出出发,分析系统要做哪些事情,进而考虑如何做这些事情,自顶向下地对系统的功能进行分解,来建立系统的功能结构和相应的程序模块结构,有效地将一个较复杂的程序系统设计任务分解成许多易于控制和处理的子任务,便于开发和维护。

随着程序规模与复杂性的增长,这种面向过程的结构化程序设计方法存在明显的不足之处。首先是数据安全性问题。由于数据被每个模块所共用,因此是不安全的,一旦出错,很难查明原因。其次是可维护性及可重用性差。它把数据结构和算法分离为相互独立的实体,一旦数据结构需要改变时,常常要涉及整个程序,修改工作量极大并容易产生新的错误。每一种相对于老问题的新方法都要带来额外的开销。另外,图形用户界面的应用程序,很难用过程来描述和实现,开发和维护也都很困难。

(2) 什么是面向对象程序设计方法? 它有哪些优点?

【问题解答】

面向对象的程序设计方法中,将程序设计为一组相互协作的对象而不是一组相互协作的函数。在程序中,属性用数据表示,用来描述对象静态特征;行为用程序代码实现,用来描述对象动态特征。可见,在面向对象的程序设计方法中,对象是数据结构和算法的封装体。对象之间存在各种联系,它们之间通过消息进行通信。程序可表示为:

程序 = 对象 + 消息

在面向对象程序设计中应着重于类的设计。类正是面向对象语言的基本程序模块,通过类的设计来完成实体的建模任务。类通过一个简单的外部接口与外界发生关系。一个类中的操作不会处理到另一个类中的数据,这样程序模块的独立性、数据的安全性就有了良好的保障。程序的执行取决于事件发生的顺序,由顺序产生的消息来驱动程序的执行。不必预先确定消息产生的顺序,更符合客观世界的实际。并且面向对象程序设计方法提供了软件重用、解决大问题和复杂问题的有效途径,具有抽象性、封装性、继承性和多态性等特点。

【问题解答】 所谓“同”是指, 同不指同类型函数, 而是指于由, 和函数出类, 是四个一

。言理十好親對的意快向而的玉個个一最自語 say

第2章

从C到C++

一、名词解释

引用 内联函数 重载函数

【问题解答】

所谓引用就是给对象取一个别名,使用该别名可以存取该对象。换句话说就是使新对象和原对象共用一个地址。

内联函数是使用 inline 关键字声明的函数。

重载函数指在同一个作用域内名字相同而参数不同的函数。重载函数通常用来对具有相似行为而数据类型或数据个数不同的操作提供一个通用的名称。

二、填空题

- (1) 一般情况下,用 C++ 语言编写的程序是由函数加上类组成的。
- (2) C++ 有两种注释符号,一种是//,另一种是/* */。
- (3) 使用 C++ 风格的输入输出,在程序中必须包含头文件“iostream”。
- (4) cin 是预定义的标准输入流对象,>>是输入操作符,也称提取运算符。
- (5) cout 是预定义的标准输出流对象,<<是输出操作符,也称插入运算符。
- (6) 指针的值是它所指向那个对象的地址值。指针的类型是它所指向对象的类型。指针的内容便是它所指向对象的值。
- (7) C++ 使用运算符& 来定义一个引用,对引用的存取都是对它所引用的对象的存取。
- (8) 当一个函数调用出现在函数定义之前时,必须先用函数原型对函数进行声明。
- (9) C++ 有值传递和引用传递两种参数传递机制。
- (10) 使用关键字inline 声明的函数称为内联函数。
- (11) 运算符new 用于进行动态内存分配,运算符delete 用于释放动态分配的内存。
- (12) 下面程序的输出结果为x=10, y=10;
x=100, y=100。

```
#include<iostream>
using namespace std;
int main()
```

```
{
    int x=10,&y=x;
    cout<<"x="<<x<<" ,y="<<y<<endl;
    int *p=&y;
    *p=100;
    cout<<"x="<<x<<" ,y="<<y<<endl;
```

三、选择题(至少选一个,可以多选)

```
A. int p1, * p2, p3;
```

```
B. int * p2, p1, * p3;
```

```
C. int p1, * p2=&p1, * p3;
```

```
D. int * p2, p1, * p3 = &p1;
```

指针定义的具体格式如下所示:

(2) 若有定义“double xx=3.14, * pp=&xx;”, 则 * pp 等价于 (C)。

A. & xx

B. * xx

C. 3.14

D. xx

pp 指向 xx 所在的内存单元, 这样 * pp 和 xx 等价。

A. 引用是某个变量或对象的别名

B. 建立引用时,要对它初始化

C. 对引用初始化可以使用任意类型的变量

D. 引用与其代表的对象具有相同的地址

所谓引用就是给对象取一个别名,使用该别名可以存取该对象,所以对引用初始化必须使用同类型的变量。

(4) 函数没有返回值的时候,应该选择(A)的函数类型。

A. void

B. int

C. 不确定

D. float

(5) 在函数的定义格式中,下面各组成部分中,(D)是可以省略的。

A. 函数名

B. 函数体

C. 返回值类型

D. 函数参数

函数的定义可以缺省形式参数,此时称为无参函数。

(6) 对重载的函数来说,下面叙述不正确的是(D)。

A. 参数的类型不同

B. 参数的顺序不同

C. 参数的个数不同

D. 参数的个数、类型、顺序都相同,但函数的返回值类型不同

对重载的函数来说,编译系统将根据函数参数的类型和个数来判断使用哪一个函数,所以重载函数参数的个数、类型、顺序不能都相同。

(7) 下列有关设置函数参数默认值的描述中, (D) 是正确的。

A. 对设置函数参数默认值的顺序没有任何规定

B. 函数具有一个参数时不能设置默认值