高等学校教材·计算机科学与技术

可下载教学资料
http://www.tup.tsinghua.edu.cn

Visual C++面向对象程序设计教程与实验

温秀梅　丁学钧　主编

清华大学出版社

高等学校教材·计算机科学与技术

Visual C++面向对象程序设计教程与实验

温秀梅　丁学钧　主编

清华大学出版社
北京

内容简介

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。本书的附录部分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。

图书在版编目(CIP)数据

Visual C++面向对象程序设计教程与实验/温秀梅，丁学钧主编. —北京：清华大学出版社，2005.10
(高等学校教材·计算机科学与技术)
ISBN 7-302-11856-6

Ⅰ. V…　Ⅱ. ①温… ②丁…　Ⅲ. C语言—程序设计—高等学校—教材　Ⅳ. TP312

中国版本图书馆CIP数据核字(2005)第109315号

出 版 者：清华大学出版社　　**地　　址**：北京清华大学学研大厦
http://www.tup.com.cn　　**邮　　编**：100084
社 总 机：010-62770175　　**客户服务**：010-62776969
责任编辑：魏江江
印 装 者：北京鑫海金澳胶印有限公司
发 行 者：新华书店总店北京发行所
开　　本：185×260　**印张**：23.25　**字数**：574千字
版　　次：2005年10月第1版　2005年10月第1次印刷
书　　号：ISBN 7-302-11856-6/TP·7704
印　　数：1～3000
定　　价：33.00元

高等学校教材·计算机科学与技术

编审委员会成员

（按地区排序）

清华大学	周立柱	教授
	覃　征	教授
	王建民	教授
	刘　强	副教授
	冯建华	副教授
北京大学	杨冬青	教授
	陈　钟	教授
	陈立军	副教授
北京航空航天大学	马殿富	教授
	吴超英	副教授
	姚淑珍	教授
中国人民大学	王　珊	教授
	孟小峰	教授
	陈　红	教授
北京交通大学	阮秋琦	教授
北京信息工程学院	孟庆昌	教授
北京科技大学	杨炳儒	教授
石油大学	陈　明	教授
天津大学	艾德才	教授
复旦大学	吴立德	教授
	吴百锋	教授
	杨卫东	副教授
华东理工大学	邵志清	教授
华东师范大学	杨宗源	教授
	应吉康	教授
东华大学	乐嘉锦	教授
上海第二工业大学	蒋川群	教授
浙江大学	吴朝晖	教授
	李善平	教授
南京大学	骆　斌	教授
南京航空航天大学	秦小麟	教授
南京理工大学	张功萱	教授
南京邮电学院	朱秀昌	教授

苏州大学	龚声蓉　教授
江苏大学	宋余庆　教授
武汉大学	何炎祥　教授
华中科技大学	刘乐善　教授
中南财经政法大学	刘腾红　教授
华中师范大学	王林平　副教授
	魏开平　教授
武汉理工大学	李中年　教授
国防科技大学	赵克佳　教授
	肖　侬　副教授
中南大学	陈松乔　教授
湖南大学	林亚平　教授
	邹北骥　教授
西安交通大学	沈钧毅　教授
	齐　勇　教授
西北大学	周明全　教授
长安大学	巨永峰　教授
西安石油学院	方　明　教授
西安邮电学院	陈莉君　副教授
哈尔滨工业大学	郭茂祖　教授
吉林大学	徐一平　教授
	毕　强　教授
长春工程学院	沙胜贤　教授
山东大学	孟祥旭　教授
	郝兴伟　教授
山东科技大学	郑永果　教授
中山大学	潘小轰　教授
厦门大学	冯少荣　教授
福州大学	林世平　副教授
云南大学	刘惟一　教授
重庆邮电学院	王国胤　教授
西南交通大学	杨　燕　副教授

出版说明

改革开放以来，特别是党的十五大以来，我国教育事业取得了举世瞩目的辉煌成就，高等教育实现了历史性的跨越，已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上，高等教育规模取得如此快速的发展，创造了世界教育发展史上的奇迹。当前，教育工作既面临着千载难逢的良好机遇，同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾，是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001 年 8 月，教育部下发了《关于加强高等学校本科教学工作，提高教学质量的若干意见》，提出了十二条加强本科教学工作提高教学质量的措施和意见。2003 年 6 月和 2004 年 2 月，教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件，指出“高等学校教学质量和教学改革工程”是教育部正在制定的《2003—2007 年教育振兴行动计划》的重要组成部分，精品课程建设是“质量工程”的重要内容之一。教育部计划用五年时间（2003—2007 年）建设 1500 门国家级精品课程，利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放，以实现优质教学资源共享，提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作，提高教学质量的若干意见》精神，紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”，在有关专家、教授的倡议和有关部门的大力支持下，我们组织并成立了“清华大学出版社教材编审委员会”（以下简称“编委会”），旨在配合教育部制定精品课程教材的出版规划，讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师，其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求，“编委会”一致认为，精品课程的建设工作从开始就要坚持高标准、严要求，处于一个比较高的起点上；精品课程教材应该能够反映各高校教学改革与课程建设的需要，要有特色风格、有创新性（新体系、新内容、新手段、新思路，教材的内容体系有较高的科学创新、技术创新和理念创新的含量）、先进性（对原有的学科体系有实质性的改革和发展、顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向）、示范性（教材所体现的课程体系具有较广泛的辐射性和示范性）和一定的前瞻

性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括:

(1) 高等学校教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 高等学校教材·电子信息——高等学校电子信息相关专业的教材。

(4) 高等学校教材·软件工程——高等学校软件工程相关专业的教材。

(5) 高等学校教材·信息管理与信息系统。

清华大学出版社经过近 20 年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材经过 20 多年的精雕细刻,形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

E-mail: dingl@tup.tsinghua.edu.cn

前言

作为一种计算机语言，C++ 有很多优点。它既可以进行过程化程序设计，也可以进行面向对象程序设计，很多复杂的算法和设计可以比较容易地用 C++ 面向对象的思想来实现。

在编写本书之前，作者已在高校从事了多年的“C++ 语言程序设计”、“面向对象程序设计”教学及科研工作，对于该语言的概念、功能及应用有着较深入的理解和丰富的实践经验。在教学过程中，我们发现很多教材在讲解 C++ 语言时既包括结构化程序设计又包括面向对象程序设计，而在面向对象程序设计部分讲得不透彻，不适合计算机专业的学生学习。故组织编写了这本教材，旨在通过本教材在内容安排、教学深度及实验要求等方面满足计算机专业本科生“面向对象程序设计”课程的教学要求。

作为一本教材，本书具有如下特点：

(1) 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，并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。使全书结构严谨、通俗易懂，兼有普及与提高的双重功能。

(2) 本书没有涉及面向过程的程序设计内容，只在第 2 章中讲解了 C++ 在结构化程序设计方面对 C 的扩充，因此学生应在学习了相关的基础知识之后再使用本教材。

(3) 本书以现代教育理念为指导，在讲授方式上注意结合应用开发实例，注重培养学生理解面向对象程序设计思想，以提高分析问题和解决实际问题的能力。

(4) 本书中的所有程序都是在 VC 6.0 环境下编译调试通过的。

本书由温秀梅、丁学钧主编并统稿，孟凡兴、刘建臣任副主编。参加编写的有：丁学钧(第 1～2 章)，温秀梅(第 3～11 章、附录 E)，孟凡兴(第 12 章)，李建华(附录 A、B)，宋淑彩(附录 C)，周丽莉(附录 D)，刘建臣担任本书的审校工作。参加本书部分内容编写工作的还有赵巍、徐晓君、岳杰、庞慧、董颢霞、王庆林、司亚超、刘海龙等。在本书的大纲讨论和分工编写过程中，我们始终互相帮助，彼此鼓励，是一次非常难忘的经历。

在此还要特别感谢我们的学生梁金龙，他为本书做了很多前期工作。

由于时间仓促，加之水平有限，书中难免有疏漏和错误之处，恳请广大读者和专家指正。

编者

目 录

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更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"。

为了克服20世纪60年代出现的软件危机，1968年北约组织提出"软件工程"的概念。对程序设计语言的认识从强调表达能力为重点转向以结构化和简明性为重点，将程序从语句序列转向相互作用的模块集合。1969年，E. W. Dijkstra首先提出了结构化程序设计的概念，他强调从程序结构和风格上来研究程序设计。在软件工程迫切需要改进的背景下，20世纪70年代结构化语言获得蓬勃发展并得到广泛应用。使用结构化程序设计方法可显著地减少软件的复杂性，提高软件的可靠性、可测试性和可维护性。经过几年的探索和实践，结构化程序设计的应用确实取得了成效，用结构化程序设计的方法编写出来的程序不仅结构良好、易写易读，而且易于验证其正确性。

进入20世纪80年代，由于一系列高技术的研究，如第五代计算机、计算机辅助制造CAM和知识工程等领域的研究，都迫切要求大型的软件系统作支撑。他们所用的数据类型也超出了常规的结构化数据类型的范畴，并且提出了对图像、声音、规则等非结构化信息的管理。为了满足这些应用领域的需要，就迫切要求软件模块具有更强的独立自治性，以便于大型软件的管理、维护和重用。由于结构化语言的数据类型较为简单，采用过程调用机制也不够灵活，独立性较差所以不能胜任对非结构化数据的定义与管理。

为了适应高技术发展的需要，消除结构化程序设计语言的局限，自20世纪80年代以来，出现了面向对象程序设计流派，研制出了多种面向对象程序设计语言(简称为OOPL：Object Oriented Programming Language)，如Ada、Smalltalk、C++语言和当前使用在Internet上的平台无关语言Java等。

由于OOPL的对象、类具有高度的抽象性，所以能很好地表达复杂的数据类型，并且，OOPL也允许程序员灵活地定义自己所需要的数据类型。类本身具有完整的封装性，可以使用它作为编程中的模块单元，满足模块独立自治的要求。另外，类的继承性和多态性功能更有助于简化大型软件和大量重复定义的模块，从而增强了模块的可重用性，提高了软件的可靠性，缩短了软件的开发周期。

1.2 面向对象是软件方法学的返璞归真

客观世界是由许多具体事物、抽象概念、规则等组成的，人们将任何感兴趣或要加以研究的事、物、概念统称为对象(Object)。每个对象都有各自的内部状态和运动规律，不同对象之间通过消息传递进行相互作用和联系就构成了各种不同的系统。面向对象的方法正是以对象作为基本元素的一种分析问题和解决问题的方法。

传统的结构化方法强调的是功能抽象和模块化，每个模块都是一个过程，结构化方法处理问题是以过程为中心的。对象包含数据和对数据的操作，是对数据和功能的抽象和统一。而面向对象强调的是功能抽象和数据抽象，用对象来描述事物和过程，面向对象方法处理问题的过程是对一系列相关对象的操纵，即发送消息到目标对象，由目标对象执行相应的操作。因此面向对象方法是以对象为中心的，这种以对象为中心的方法更自然、更直接地反映现实世界的问题空间，从而具有独特的抽象性、封装性、继承性和多态性的特点，更好地适应

了复杂大系统不断发展与变化的要求。

采用对象的观点看待所要解决的问题，并将其抽象为应用系统是极其自然与简单的，因为它符合人类的思维习惯，使得应用系统更容易理解。同时，由于应用系统是由相互独立的对象构成的，系统的修改可以局部化，因此系统维护更加容易。

软件开发从本质上讲就是对软件所要处理的问题域进行正确的认识，并把这种认识正确地描述出来。既然如此，那就应该直接面对问题域中客观存在的事物来进行软件开发，这就是面向对象。另一方面，人类在认识世界的过程中形成的普遍有效的思维方法，在软件开发中也是适用的。在软件开发中尽量采用人们日常生活中习惯的思维方式和表达方式，这就是面向对象方法所强调的基本原则。软件开发从过分专业化的方法、规则和技巧中脱离出来，并重新回到了客观世界，回到了人们的日常思维当中，所以说面向对象方法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

要想真正了解面向对象程序设计，首先需要回顾一下结构化程序设计的含义。

1. 结构化程序设计

结构化程序设计是20世纪60年代诞生的，在70年代到80年代已遍及全球，成为软件开发设计所有领域及每个程序员都采用的程序设计方法，它的产生和发展形成了现代软件工程的基础。

结构化程序设计的设计思想是：自顶向下、逐步求精；其程序结构按功能划分为若干个基本模块，这些模块形成一个树状结构；各模块之间的关系尽可能简单，在功能上相对独立；每一模块内部均由顺序、选择和循环三种基本结构组成；其模块化实现的具体方法是使用子程序、过程或函数。

结构化程序设计由于采用了模块分解和功能抽象、自顶向下、分而治之的手段，从而有效地将一个复杂的软件系统的设计任务分成许多容易控制和处理的子任务，这些子任务都是可独立编程实验的子程序模块。每一个子程序都有一个清晰的界面，使用起来非常方便。

结构化程序设计方法虽然具有许多的优点，但它仍是一种面向过程的设计方法，它把数据和过程分离为相互独立的实体，程序员在编程时必须时刻考虑所要处理的数据格式。对于不同的数据格式即使做同样的处理或对相同的数据格式做不同的处理都需要编写不同的程序。因此结构化程序的可重用性不好；另一方面，当数据和过程相互独立时，总存在着用错误的数据调用正确的程序模块或用正确的数据调用错误的程序模块的可能性。因此，要使数据和程序始终保持相容，已成为程序员的一个沉重负担，并且随着软件系统的规模越来越大，程序的复杂性越来越难以控制。上述这些问题，结构化程序设计方法本身是解决不了的，需要借助于下面要讨论的面向对象程序设计方法给予解决。

程序设计的任务是描述问题并解决问题，在结构化程序设计中可以用下面的式子表示程序：

程序＝数据结构＋算法＋程序设计语言＋语言环境

图 1.1 所示为结构化程序设计中程序的结构。

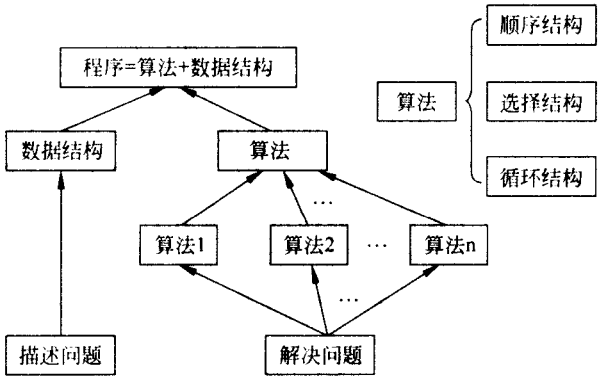

图 1.1 结构化程序设计中程序的结构

2. 面向对象程序设计——程序设计的新思维

面向对象程序设计既吸取了结构化程序设计的一切优点，又考虑了现实世界与面向对象空间的映射关系，它所追求的目标是将现实世界问题的求解尽可能简单化。

面向对象程序设计将数据及对数据的操作放在一起，作为一个相互依存、不可分割的整体来处理，它采用了数据抽象和信息隐藏技术。它将对象及对对象的操作抽象成一种新的数据类型——类，并且考虑不同对象之间的联系和对象所在类的可重用性。

面向对象程序设计优于传统的结构化程序设计，其优越性表现在，它有希望解决软件工程的两个主要的问题——软件复杂性控制和软件生产率的提高，此外它还符合人类的思维习惯，能够自然地表现现实世界的实体和问题，它对软件开发过程具有重要的意义。

面向对象程序设计能支持的软件开发策略有：

(1) 编写可重用代码。

(2) 编写可维护代码。

(3) 共享代码。

(4) 精减已有代码。

有了高质量的可重用代码就能有效地降低软件的复杂度、提高开发效率。面向对象方法，尤其是它的继承性，是一种代码重用的有效途径。开发者在设计软件时可以利用一些已经精心设计好并且经过测试的代码，这些可重用的代码以类的形式被组织存放在程序设计环境的类库中。类库中的这些类的存在，使以后的程序设计过程变得简单，程序复杂性不断降低、正确性不断提高，程序越来越容易理解、修改和扩充。

在面向对象程序设计中可以用下面的式子表示程序：

程序＝对象＋对象＋……＋对象

对象＝算法＋数据结构＋程序设计语言＋语言环境

图 1.2 所示为面向对象程序设计中程序的结构。