高等教育计算机学科“应用型”教材

C/C++ 程序设计教程
——面向过程分册

郑秋生　主编
王黎明　主审

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

高等教育计算机学科“应用型”教材

C/C++程序设计教程
——面向过程分册

郑秋生　主　编
夏敏捷　罗　菁　副主编
王黎明　主　审

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

《C/C++程序设计教程》系列教材分为面向过程和面向对象两个分册，适合分两个学期讲授。本书为面向过程分册。

本书系统地阐述了C++语言中过程化程序设计的思想、语法、方法，主要内容包括C++程序设计的基础知识、基本数据类型和表达式、C++的程序控制语句、数组与函数、指针和引用、用户定义数据类型、文件等。书中内容讲解清晰、实例丰富、力避代码复杂冗长，注重算法设计和程序设计思想。简短的实例有助于初学者更好地理解、把握解决问题的精髓，帮助读者快速地掌握程序设计的基本方法。

本书的特点是实例丰富，重点突出，叙述深入浅出，分析问题透彻，既有完整的语法，又有大量的实例，突出程序设计的算法、方法，将C语言程序设计和C++程序设计有机地进行统一。

本书适合作为计算机学科各应用型本科、专科的C语言程序设计和C++程序设计教材，也可作为其他理工科各专业的教材和相关技术人员的自学参考书。

图书在版编目(CIP)数据

C/C++程序设计教程. 面向过程分册/郑秋生主编. —北京：电子工业出版社，2007.9
高等教育计算机学科"应用型"教材
ISBN 978-7-121-04915-6
Ⅰ. C… Ⅱ. 郑… 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(2007)第131858号

策划编辑：张 旭
责任编辑：张燕虹 张 慧
印 刷：北京市天竺颖华印刷厂
装 订：三河市金马印装有限公司
出版发行：电子工业出版社
北京市海淀区万寿路173信箱 邮编 100036
开 本：787×1092 1/16 印张：17 字数：435千字
印 次：2007年9月第1次印刷
定 价：24.00元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010)88254888。

质量投诉请发邮件至zlts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线：(010)88258888。

前　言

本书的主要作者都是具有丰富教学经验的一线教师，从事 C/C++ 程序设计课程教学多年，深知学生在学习 C++ 程序设计这门课程后，对程序设计方法、算法设计、调试程序、习题解答的茫然和问题。因此本书在介绍理论知识、相关概念和语言语法时，始终强调其在程序设计中的作用，使语言语法与程序设计相结合。同类书籍大部分偏重于对语言语法和概念的介绍，虽然在书中有针对一个语法和知识点的程序实例，但学生对每章内容在实际程序设计中的作用缺乏了解，而本书每章节后都附有针对性较强的应用实例分析，尽可能使初学者在学习每章的内容后，拿到题目，既能够独立设计程序、解决实际问题，又不至于无从下手。

本书有以下五个鲜明特点：

(1) 改变了传统的教学模式。先介绍 C 语言程序设计，再介绍 C++ 对 C 语言的扩展、面向对象的程序设计。本教材将 C/C++ 语言的学习很好地融合在一起，让读者把面向过程和面向对象的程序设计方法有机地结合在一起，面向过程和面向对象两分册统一使用 Visual C++ 6.0 编译器。

(2) 克服了传统教材以语言、语法学习为重点的缺点，本教材从基本的语言、语法学习上升到程序的“设计、算法、编程、调试”层次。为了让学生更好地掌握程序开发思想、方法和算法，书中提供了大量简短精辟的代码，有助于初学者学习解决问题的方法和诀窍。在每章后都有单独一节讲述关于程序综合设计的内容，有一个或多个较大的程序，以帮助学生更好地掌握程序设计方法和解决实际问题的能力。

(3) 教材强调程序的设计方法，大量例题有流程图、N－S 图和 UML 图，即突出程序的算法和设计，而不仅仅是语法和编程，培养学生程序设计能力和程序调试技能，养成好的编程习惯，为专业程序员的培养打下良好的基础。

(4) 培养学生面向对象程序设计的能力，引导学生建立程序设计的大局观，帮助学生掌握从客观事物中抽象出 C++ 类的方法。通过系统的学习，使学生的编程能力上一个台阶，具备解决复杂问题的程序设计能力。

(5) 根据当前实际大型软件项目开发的需要，加大了异常处理、模板等内容，新增了 STL 标准模板库，并通过流行的 UML 工具设计 C++ 类。

本教材的编写充分考虑了目前应用型本科 C/C++ 语言程序设计课程教学的实际情况和存在的问题：

(1) 学生在大一阶段的基础课程较多，不可能投入过多的精力来学习本门课程。

(2) 大学生对这门课学习的期望值很高，但对学习时可能遇到的困难估计不足。

(3) 大学生现有的上机实践条件大大改善，特别有利于贯彻先进的精讲多练的教学思想。

(4) 学生学会了语言的语法，仍不具备解决实际问题的能力，学生的程序设计、算法设计、编程、调试的能力相对较差。

本教材正是考虑了学生的这些实际问题，而精心编写了这套面向应用型本科的 C/C++ 程序设计教程，特别适合于分两个学期系统讲授 C/C++ 程序设计。第 1 学期讲授面向过程

分册，第2学期讲授面向对象分册。

本书共9章，第1章主要讨论C++语言特点和编辑环境，第2章～第8章主要介绍用C/C++进行过程化程序设计的基本方法，内容包括表达式及运算符、数据类型、函数、数组、指针等，第9章主要介绍文件处理方法。

为了方便使用本教材的教师备课，我们还提供了配套的电子教案，公开放在网站上，供任课教师自由下载使用。相信我们多年的教学经验会对广大师生的教和学有所帮助。建议本分册的教学学时为60个学时，其中理教为44学时，课内上机实践为16学时，课外上机不少于32学时。

本教材的编写得到了河南省计算机学会的大力支持，组织了河南多所高校编写了高等教育计算机学科“应用型”系列教材。参编本教材的高校有中原工学院、郑州大学、洛阳师范学院、河南工程学院、河南科技大学、郑州轻工业学院。

本分册第1章和第7章由郑秋生编写，第2章由马宗梅编写，第3章和附录由夏敏捷编写，第4章由张瑞玲编写，第5章由杜献峰编写，第6章由丁钰编写，第8章的前8节由王岚编写，其余由罗菁编写，第9章由赵丹编写。全书由郑秋生修改并统稿。为本书提出改进意见和建议的老师有郑州大学的钱晓捷和卢红星老师，在此向他们表示衷心的感谢。

由于编者水平有限，时间仓促，书中难免有错，敬请广大读者批评指正，在此表示感谢。

作者 E-mail：zqs@zzti.edu.cn

作者

2007年7月

目　录

第1章

C++概述

C++语言是一种高效率实用的程序设计语言，使用它既可进行过程化程序设计，也可进行面向对象的程序设计。本章主要介绍计算机语言的发展过程、过程化程序设计和面向对象的程序设计方法、C++程序的基本结构以及程序开发的步骤。

通过本章学习，应该重点掌握以下内容：

- 计算机语言发展的历程
- 过程化程序设计和面向对象的程序设计的基本思想和主要特点
- 简单的C++程序结构
- C++开发程序的步骤

1.1 计算机程序设计语言的发展

计算机语言通常是能完整、准确和规则地表达人们的意图，并用以指挥或控制计算机工作的“符号系统”。当使用计算机解决问题时，首先将解决问题的方法和步骤按照一定的顺序和规则用计算机语言描述出来，形成指令序列，然后由计算机执行指令，完成所需的功能。

计算机程序设计语言的发展，经历了从机器语言、汇编语言到高级语言的历程。

1.1.1 机器语言阶段

众所周知，在计算机内部采用二进制表示信息。机器语言(Machine Language)是用二进制代码表示的、计算机能够直接识别和执行的一种机器指令的集合。它是面向机器的语言，是计算机唯一可直接识别的语言。用机器语言编写的程序称为机器语言程序(又称为目标程序)。每一条机器指令的格式和含义都是由设计者规定的，并按照这个规定设计制造硬件。一个计算机系统的全部机器指令的总和，称为指令系统。不同类型的计算机的指令系统不同。

例如，某种计算机的指令为：

```
10110110 00000000        表示进行一次加法操作
10110101 00000000        表示进行一次减法操作
```

它们的前 8 位表示操作码，而后 8 位表示地址码。从上面两条指令可以看出，它们只是在操作码中从左边第 0 位算起的第 6 位与第 7 位不同。这种机型可包含 256(2^8)个不同的指令。

用机器语言编写的程序，能够直接在计算机上运行，运行的速度快，效率高。但机器语言难于记忆，也难于操作，代码编程烦琐、易出错；而且编写的程序紧密依赖计算机硬件，程序的可移植性差。

机器语言是第一代计算机语言。

1.1.2 汇编语言阶段

为了克服机器语言的缺点，使语言便于记忆和理解，人们采用能反映指令功能的助记符来表达计算机语言，称为汇编语言(Assembly Language)。汇编语言采用的助记符比机器语言直观、容易记忆和理解。汇编语言也是面向机器的程序设计语言，每条汇编语言的指令对应一条机器语言的指令，不同类型的计算机系统一般有不同的汇编语言。

例如，用汇编语言编写的程序如下：

```
MOV   AL   10D        // 将十进制数 10 送往累加器
SUB   AL   12D        // 从累加器中减去十进制数 12
......
```

用汇编语言编写程序比用机器语言要容易得多，但计算机不能直接执行汇编语言程序。必须把它翻译成相应的机器语言程序才能运行。将汇编语言程序翻译成机器语言程序的过程称为汇编，汇编过程是由计算机运行汇编程序自动完成的，如图 1-1 所示。

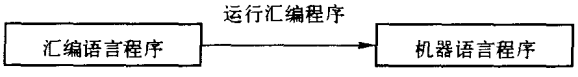

图 1-1 汇编过程

在计算机语言系统中，汇编语言仍然列入“低级语言”的范畴，它仍然依赖于计算机的硬件，并且移植性差。但汇编语言比起机器语言在很多方面都有优越性，如编写容易、修改方便、阅读简单、程序清楚等，针对计算机硬件而编制的汇编语言程序，能准确地发挥计算机硬件的功能和特长，程序精练且质量高，所以至今仍是一种常用的程序设计语言。

汇编语言是第二代计算机语言。

1.1.3 高级语言阶段

机器语言和汇编语言都是面向机器（计算机硬件）的语言（低级语言），受机器硬件的限制，通用性差，也不容易学习，一般只适用于专业人员。人们意识到，应该设计一种语言：它接近于数学语言或自然语言，同时又不依赖于计算机的硬件，编出的程序能在所有的计算机上通用。高级语言（High-Level Language）就是这样的语言。例如，用C++语言编写的程序片断如下：

```
int i, j, k;          //定义变量 i, j, k
cin >> i >> j;        //输入 i, j 的值
k=i*j;                //将变量 i, j 的值相乘，结果赋给变量 k
cout << k;            //输出求积结果
```

如上例，使用高级语言编写程序时，不需要了解计算机的内部结构，只需告诉计算机“做什么”。至于计算机用什么机器指令去完成（即“怎么做”），编程者不需要关心。高级语言是面向用户的。

用高级语言编写的程序称为高级语言源程序，计算机无法直接执行，必须翻译或解释成机器语言目标程序才能被计算机执行。翻译过程分为两步，即编译和连接，编译和连接过程如图1-2所示。

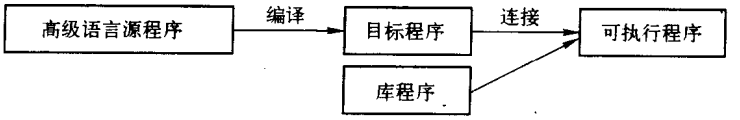

图1-2 编译和连接过程

在上图中，高级语言经过编译后，得到目标程序（.obj），再与库程序连接生成可执行程序（.exe）。

程序设计语言从机器语言到高级语言的抽象，带来如下主要好处：

（1）高级语言接近自然语言，易学、易掌握，一般工程技术人员只需几周时间的培训就可以胜任程序员的工作。

（2）高级语言为程序员提供了结构化程序设计的语法，使得设计出来的程序可读性好、可维护性强、可靠性高。

（3）高级语言远离机器语言，与具体的计算机硬件关系不大，因而所写出来的程序可移植性好、代码重用率高。

（4）由于把繁杂琐碎的事务交给了编译程序去做，所以自动化程度高、开发周期短，并且程序员得到解脱，可以集中时间和精力去设计算法和从事更为重要的创造性劳动，以提高程序的质量。

高级语言是第三代计算机语言。目前广泛应用的高级语言有多种，如 BASIC、FORTRAN、PASCAL、C、C++、Java 及 C# 等。

1.1.4 从 C 到 C++

C 语言是 AT&T 贝尔实验室的 Dennis Ritchie 在 B 语言的基础上开发出来的，1972 年在一台 DEC PDP-11 计算机上实现了最初的 C 语言。C 语言最初作为 UNIX 操作系统的开发语言，UNIX 操作系统的 90% 的代码由 C 语言编写，10% 的代码由汇编语言编写。由于 UNIX 的成功和广泛使用，也使 C 语言成为一种普遍使用的程序设计语言。

C 语言具有如下优点：

(1) 语言简洁、紧凑，使用方便、灵活。C 语言只有 32 个关键字，程序书写形式自由。

(2) 运算符丰富，数据结构丰富，具有现代化语言的各种数据结构。

(3) 具有结构化的控制语句(如 if...else 语句、while 语句、for 语句)。

(4) 语法限制不很严格，程序设计自由度大。

(5) C 语言允许直接访问物理地址。

(6) 生成目标代码质量高，程序执行效率高。

(7) 用 C 语言编写的程序可移植性好。

但是，C 语言也有它的局限性：

(1) C 语言数据类型检查机制较弱，这使得程序中的一些错误不能在编译时被自动发现。

(2) 当程序的规模大到一定程度时，很难控制复杂性。

为了解决这些问题，研制 C++ 语言的一个首要目标就是使 C++ 语言突破 C 语言的局限性。同时，在 C++ 中引入了类等机制来支持面向对象的程序设计。所研制的这个语言最初称为“带类的 C”，1983 年取名为 C++ (C Plus Plus)。C++ 的喻义是对 C 语言进行“增值”。1994 年，制定了 ANSI C++ 草案。后来又经过不断的完善和发展，成为今天的 C++，目前 C++ 仍在不断的发展中。

C++ 是由 C 发展而来的，与 C 兼容。C++ 包含了 C 的全部特征、属性和优点，是 C 的超集，同时 C++ 添加了面向对象编程的完全支持，是一种功能强大的面向对象的程序设计语言。

1.2 过程化程序设计

程序设计的基本目标是：用算法对问题的原始数据进行处理，从而获得所期望的效果。但是，这仅仅是程序设计的基本要求，要想全面提高程序的质量、编程效率，使程序具有良好的可读性、可靠性、可维护性以及良好的结构，就必须掌握正确的程序设计方法和技术。

在面向对象的程序设计方法出现以前，我们都采用面向过程的程序设计方法。例如计算炮弹的飞行轨迹，为了完成计算，就必须设计一个计算方法或解决问题的过程。由于处理的问题日益复杂，程序也就越来越复杂和庞大。20 世纪 60 年代产生的结构化程序设计方法，为用面向过程方法解决复杂问题提供了有力的手段。结构化程序设计的基本程序结构为：顺序结构、选择结构和循环结构。

过程化程序设计方法的主要思想是：将任务按功能进行分解，自顶向下、逐步求精。当一

个任务十分复杂以至无法描述时，可按功能划分为若干个基本模块，各模块之间的关系尽可能简单，在功能上相对独立，如果每个模块的功能实现了，复杂任务也就得以解决。

例如，一个简单的学生成绩管理系统是一个较为复杂的任务，可以采用过程化设计思想完成。学生成绩管理系统设计如图1-3所示。

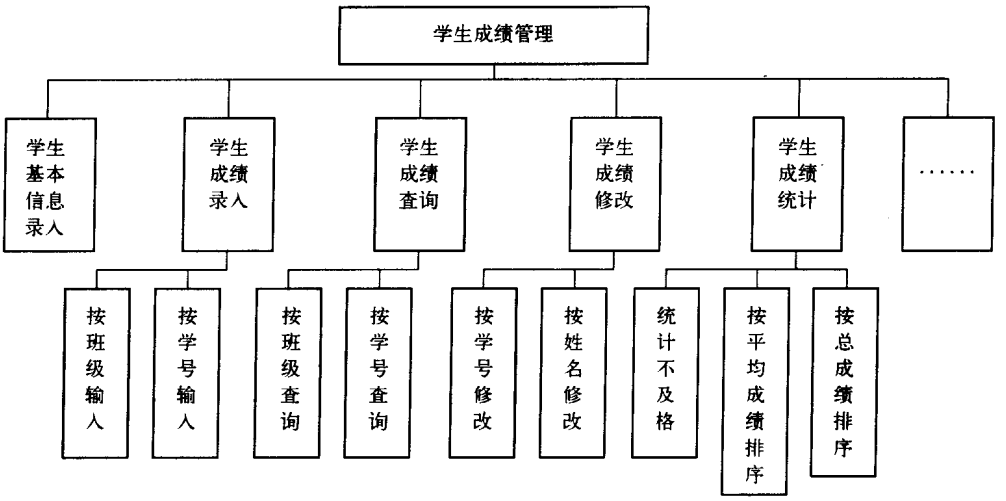

图1-3　学生成绩管理系统设计

过程化程序设计方法中，数据与处理数据的算法是分离的，重用性差。编程的主要技巧在于追踪过程调用及哪些数据发生了变化。过程化编程思想是设计数据结构和算法：

程序＝数据结构＋算法

过程化程序设计能够较好地解决一些复杂的问题，但也有很多缺点。例如，当需要处理的数据量较复杂时，数据与处理这些数据的方法之间的分离使程序变得越来越难以理解和维护。当数据结构发生变化时，必须对程序进行修改，代码的重用性差；而面向对象的程序设计方法能较好地解决这些问题。

1.3　面向对象的程序设计

面向对象的程序设计不仅吸取了结构化程序设计的优点，又考虑了现实世界与面向对象的映射关系而提出的一种新思想，它所追求的目标是将现实世界的问题求解尽可能简化，使程序设计更加贴近实现世界，用于开发较大规模的程序，以提高程序开发的效率。面向对象程序设计的实现需要数据封装、继承和多态技术。

1.3.1　基本概念

1. 对象

对象又称为实例，是客观世界中一个实际存在的事物。它既具有静态的属性（或称为状态），又具有动态的行为（或称为操作）。现实世界中的对象一般可以表示为：属性＋行为。例

如,一个盒子就是一个对象,它具有的属性为该盒子的长、宽和高等;具有的操作为求盒子的容量等。

2. 类

在面向对象程序设计中,类是具有相同属性数据和操作的对象的集合,它是对一类对象的抽象描述。例如,将所有的盒子的共同属性抽象出来就是盒子类。

类是创建对象的模板,它包含着所创建对象的属性描述和方法定义。一般是先定义类,再由类创建其对象,按照类模板创建一个个具体的对象(实例)。

3. 面向对象程序设计(Object Oriented Programming,OOP)

面向对象程序设计是将数据(属性)及对数据的操作算法(行为)封装在一起,作为一个相互依存、不可分割的整体来处理。面向对象程序设计的结构如下所示:

对象 = 数据(属性) + 算法(行为)
程序 = 对象 + 对象 + … + 对象

面向对象程序设计的优点表现在:可以解决软件工程的两个主要问题——软件复杂性控制和软件生产效率的提高;另外,它还符合人类的思维方式,能自然地表现出现实世界的实体和问题。

1.3.2 面向对象程序设计的特点

面向对象程序设计具有封装、继承、多态三大特性。

1. 封装性

封装是一种数据隐藏技术。在面向对象程序设计中,可以把数据和与数据有关的操作集中在一起形成类,将类的一部分属性和操作隐藏起来,不允许用户访问;将另一部分作为类的外部接口,允许用户访问。类通过接口与外部发生联系、沟通信息,用户只能通过类的外部接口使用类提供的服务,发送和接收信息;而内部的具体实现细节则被隐藏起来,对外是不可见的,增强了系统的可维护性。

2. 继承性

在面向对象程序设计中,继承是指新建的类从已有的类那里获得已有的属性和操作。已有的类称为基类或父类,继承基类而产生的新建类称为基类的子类或派生类。由父类产生子类的过程称为类的派生。继承有效地实现了软件代码的重用,增强了系统的可扩充性;同时,也提高了软件开发效率。

3. 多态性

在面向对象程序设计中,多态性是面向对象的另一重要特征。

所谓多态性是指当不同的对象收到相同的消息时,产生不同的动作。其好处是,用户不必知道某个对象所属的类就可以执行多态行为,从而为程序设计带来更大方便。利用多态性可以设计和实现一个易于扩展的系统。

1.4 简单的 C++ 程序介绍

下面通过一个简单的程序来说明 C++ 程序的基本结构。