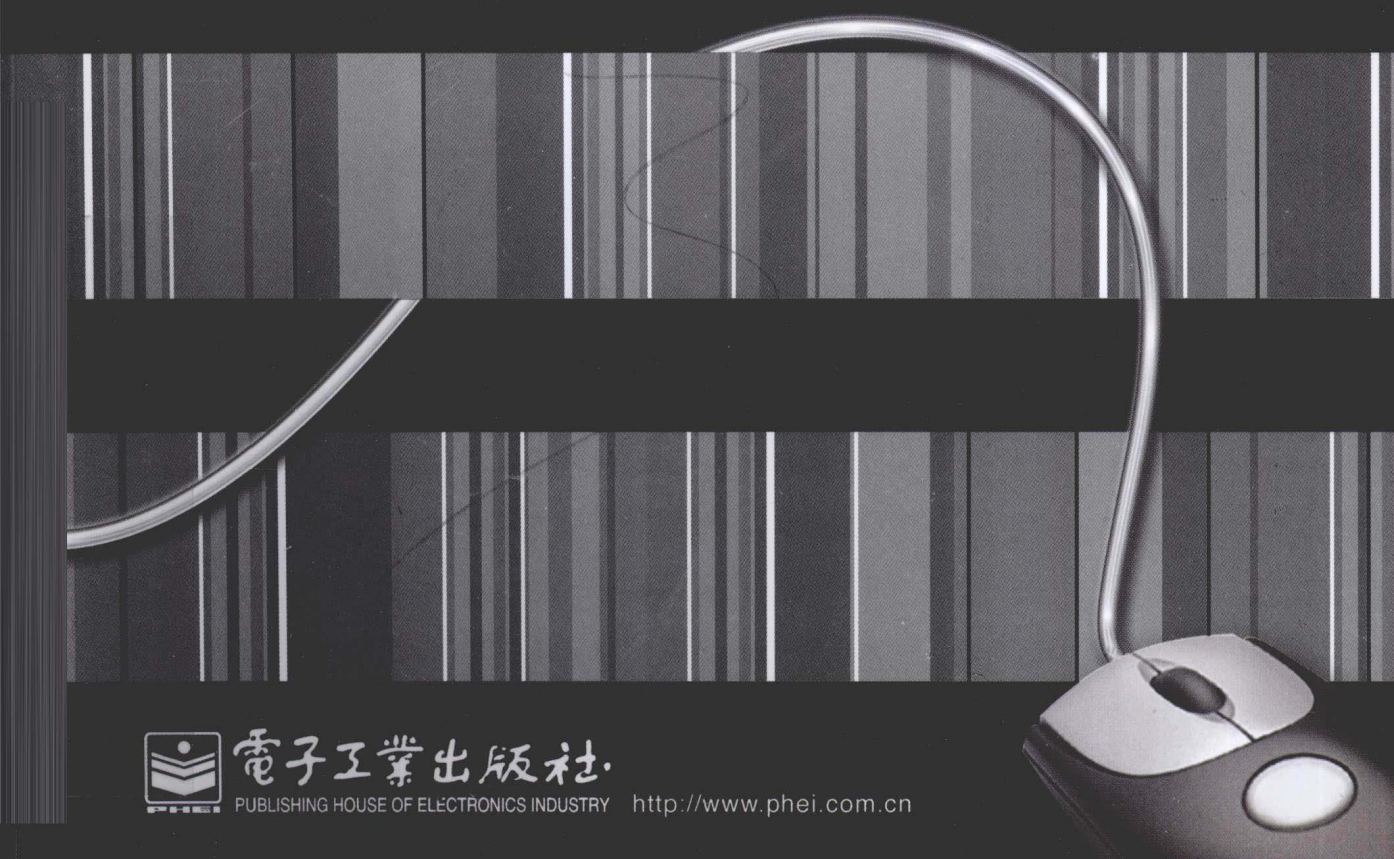
普通高等教育“十一五”国家级规划教材

21世纪大学计算机系列教材

C/C++ 程序设计教程（第3版）

孙淑霞　肖阳春　魏　琴　等编著



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

普通高等教育“十一五”国家级规划教

21世纪大学计算机系列教材

C/C++ 程序设计教程（第3版）

孙淑霞　肖阳春　魏　琴　等编著

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

内 容 简 介

本书作为C/C++程序设计课程的主教材，共由12章组成。其主要内容包括：C/C++语言程序设计概述，C语言程序设计基础（其中包括：基本数据类型、基本输入与输出函数以及运算符和表达式），控制结构，数组，指针，函数，编译预处理与变量的存储类型，文件，结构体与共用体，图形程序设计基础，C++程序设计基础，查找与排序。每章后面都有学习指导和一定量的编程练习题，书后附有习题参考答案。全书内容安排紧凑，简明扼要，由浅入深，实用性强。该书的辅教材《C/C++程序设计实验指导与测试》（第3版）中提供了其他形式的测试题及其解答，作为主教材习题的补充，将为学生编程能力的提高和课后自学提供更好的帮助。

本书可作为大专院校非计算机专业本科生、研究生的相关课程的教学用书，也可作为计算机专业学生学习C/C++程序设计的教材，同时还可供自学者参考。

图书在版编目（CIP）数据

C/C++程序设计教程/孙淑霞等编著. —3版. —北京：电子工业出版社，2009.11
（21世纪大学计算机系列教材）
ISBN 978-7-121-09839-0

Ⅰ.C… Ⅱ.孙… 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（2009）第201470号

责任编辑：胡丽华　　文字编辑：王昌铭
印　　刷：北京市天竺颖华印刷厂
装　　订：三河市鑫金马印装有限公司
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：787×1 092　1/16　印张：24.75　字数：640千字
印　　次：2009年11月第1次印刷
印　　数：4 000册　　定价：38.00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

C 语言是应用很广泛的一种语言，它的结构简单、数据类型丰富、表达能力强、使用灵活方便。C 语言既有高级语言的优点，又具有低级语言的许多特点。用 C 语言编写的程序，具有速度快、效率高、代码紧凑、可移植性好的优点。利用 C 语言，可编制各种系统软件（例如著名的 UNIX 操作系统就是用 C 语言编写的）和应用软件。

C++是一种混合语言，既有面向过程的知识，又有面向对象的理论。经过几年的教学实践，我们认为把面向过程的程序设计作为切入点，由面向过程到面向对象，由浅入深，循序渐进的教学方式比较容易被学生所接受。因此，本书在第 11 章介绍了 C++程序设计的基础知识。

本教材由 12 章组成。每一章的基本内容如下：

第 1 章 C/C++语言程序设计概述，介绍 C/C++程序的基本结构。

第 2 章 C 语言程序设计基础，介绍 C 语言的基本数据类型。

第 3 章 控制结构，介绍 C 程序的 3 种控制结构。

第 4 章 数组，介绍一维数组和二维数组的定义和使用。

第 5 章 指针，重点介绍指针变量、指针数组、指向指针的指针等的定义和使用。

第 6 章 函数，讲解函数的定义、函数的调用，函数参数的传递。

第 7 章 编译预处理与变量的存储类型，介绍编译预处理命令和变量的几种存储类型。

第 8 章 文件，介绍文件操作的方法，数据文件的读和写。

第 9 章 结构体与共用体，介绍结构体与共用体的使用，以及它们对内存的占用情况。

第 10 章 图形程序设计基础，介绍编写图形程序的基本步骤，基本图形函数。

第 11 章 C++程序设计基础，介绍 C++对 C 的扩充，以及面向对象的程序设计基础。

第 12 章 查找与排序。

本教材在编写中努力做到概念清楚、实用性强、通俗易懂。在编写中引入了大量的实例来说明相关的知识点，力求让读者尽快上手编写简单程序，激发学习兴趣。

本书在组织编写上有以下特点：

1．在内容的组织上考虑了 C 语言的特点。例如，在讲解数组后，紧接着就进行指针的讲解，使读者很容易将数组与指针联系起来，更好地理解指针。

2．文件是学生学习的一大难点。本书将文件的使用提前讲解，使读者尽早接触文件，掌握文件的基本操作，给大批量数据的处理带来方便。同时可以较好地解决学生在学习 C 语言时不能熟练地掌握文件的使用方法，而给学习 C 语言留下一大遗憾的问题。

3．全书坚持把面向过程的程序设计作为切入点，由面向过程到面向对象，由浅入深，循序渐进，使其教学内容更容易被学生接受。把 C 和 C++的内容分开，是为了教师更容易选择章节进行教学。

4．每章后面都有“本章学习指导”，共由如下三部分组成。

（1）课前思考：课前预习是必要的，课前思考中的问题可用于老师或学生检查其预习效果。

（2）本章难点：总结归纳了本章学习中的难点，以便学生了解并攻克难点。

（3）本章编程中容易出现的错误：C 程序中的错误有语法错和算法错，这里总结了一些初学者常犯的错误，以便帮助初学者避免不必要的错误。

5．本书提供了习题中的全部参考答案。所有程序均在 Turbo C/Visual C++ 6.0 环境下调试通过。由于篇幅有限，书中的程序只给出了一种参考程序，读者在学习过程中可以举一反三。

6．本书作为国家精品课程的配套使用教材，在精品课程网站上全开放地提供了大量资源，授课视频等。

与本书一起出版的《C/C++程序设计实验指导与测试》（第 3 版）是本书的配套教材，在学习过程中通过完成该配套教材中相应的习题和上机编程的练习加深对所学知识的理解，达到真正掌握 C/C++程序设计的目的。

要想学好程序设计课程，需要教师和学生的共同努力。对于学习者来说，需要多动手，多实践，多思考。一分耕耘，一分收获，坚持耕耘定会得到意想不到的收获。

本书第 1, 4 章由孙淑霞编写，第 2, 3, 7 章由肖阳春编写，第 5, 6 章由魏琴编写，第 8, 9, 12 章由李瑾坤编写，第 10, 11 章由彭舰编写。全书由孙淑霞统稿。魏琴、刘焕君为本课程制作了美观、符合授课要求的课件。丁照宇、李思明、刘焕君、鲁红英、安红岩、陈佩良参加了本精品课程的建设和本书编写过程中的部分工作。

由于作者水平有限，书中难免有错误之处，请读者批评指正。

最后要感谢为本书提出宝贵意见的老师和读者，特别要感谢电子工业出版社在本书出版过程中给予的大力支持。

该书作为国家级精品课程《C/C++程序设计》使用的教材，进行了配套的资源建设。对于使用本教材的学校，如果需要课件、例题源程序等，可以从该课程的精品课程网站 http://www.cne.cdut.edu.cn/zy/cjpkc/index.asp 上直接下载，也可以直接与我们联系（邮件地址：ssx@cdut.edu.cn）。

编著者

2009 年 10 月

目　录

第 1 章　C/C++语言程序设计概述

本章学习目标：

- 了解 C 语言的发展
- 了解 C/C++的特点
- 理解程序与程序设计
- 掌握 C 程序的基本结构
- 熟悉 C/C++程序的开发环境

1.1　引　　言

C 语言是由 B 语言和 BCPL（Basic Combined Programming Language）语言发展演化而来的。最初的 C 语言于 1972 年由贝尔实验室的 Dennis Ritchie 开发，并首次在 UNIX 操作系统的 DECPDP-11 计算机上使用，它为描述和实现 UNIX 操作系统而设计，又随 UNIX 的出名而闻名。

C 语言是国际上应用最广泛的几种计算机语言之一。它不仅可以用于编写系统软件，如操作系统、编译系统等，还可以用于编写应用软件。

随着计算机科学的发展，出现了不同版本的 C 语言，它们的差异主要体现在标准函数库中函数的种类、格式和功能上。为了有利于计算机应用技术的发展，ANSI（American National Standards Institute 美国国家标准协会）于 1983 年专门成立了定义 C 语言标准的委员会，并于 1989 年对 C 语言进行了标准化，制定出 ANSI C 的标准，又称为 C89。1995 年，经过修订的 C 语言，增加了一些库函数，出现了 C++的一些特性，使 C89 成为 C++的子集。1999 年又推出 C99，它在保留 C 语言特性的基础上，增加了面向对象的新特性。

本章简要介绍 C/C++语言的特点，C 程序的基本结构和 C 程序的调试。

1.2　C/C++语言的特点

C 和 C++是两种不同的程序设计语言，其中 C 是结构化程序设计语言，C++是面向对象的程序设计语言。

1. C 语言的特点

C 语言之所以能够广为流传，是因为它有很多不同于其他程序设计语言的特点。其主要特点有：

① 数据类型丰富。C 语言除了整型、实型、字符型等基本数据类型外，还具有数组、指针、结构、联合等高级数据类型，能够用于描述各种复杂的数据结构（如链表、栈、队列等）。指针数据类型的使用，使 C 程序结构更为简化、程序编写更为灵活、程序运行更

为高效。

② 运算符种类丰富。C语言具有数十种运算符，除了具有一般高级语言具有的运算功能外，还可以实现以二进制位为单位的位运算，直接控制计算机的硬件，还具有自增、自减和各种复合赋值运算符等。C程序编译后生成的目标代码长度短、运行速度快、效率高。

③ 符合结构化程序设计的要求。C语言提供的控制结构语句（如if-else语句、while语句、do-while语句、switch语句、for语句）使程序结构清晰，其函数结构使程序模块具有相对独立的功能，便于调试和维护，支持大型程序的多文件构成以及单个文件独立编译，有利于大型软件的协作开发。

④ 可移植性好。用C语言编写的程序几乎不做修改就可用于各种计算机和各种操作系统。

C语言的这些特点使C语言很快应用到了各计算机应用领域中的软件编写，如数据库管理、CAD、科学计算、图形图像处理、实时控制等软件。

然而，C语言也不是十全十美的，它也有缺点。主要表现在：

① 语法限制不太严格。例如，缺乏数据类型的一致性检测和不进行数组下标越界检查。正因为C语言允许编程者有较大的自由度，使C程序容易通过编译，但却难以查出运行中的错误。初学者一定不要以为编译通过了，程序就一定是正确的，就应该运行出正确结果。要想尽快找到程序中的错误，一定要掌握调试程序的方法和技术，多上机实践。

② 不适合大规模的软件开发。由于C是以数据和数据处理过程为设计核心的面向过程的程序设计语言，因此不利于提高软件开发的效率，难以适应大规模程序设计的需要。

2．C++语言的特点

C++和C是两种不同的语言。C++语言的主要特点有：

① C++是以面向对象为主要特征的语言，通过类和对象的概念把数据和对数据的操作封装在一起，通过派生、重载和多态等技术手段实现软件重用和程序自动生成，适合大规模软件的开发和维护。

② 继承了C语言的优点，兼容了C语言，因此既支持面向对象的程序设计，又支持面向过程的程序设计。用C语言编写的程序大都可以在C++环境中编译和调试。

③ 对C语言的数据类型做了扩充，使编译器可以检查出更多类型的错误，即语法检查更加严密。

1.3 程序与程序设计

计算机通过执行程序完成其工作，程序设计则是指设计、编制、调试程序的方法和过程。

1.3.1 程序

程序是计算机可以执行的一个为解决特定问题，用某种计算机语言编写的语句（指令）序列。计算机科学家沃思（Nikiklaus Wirth）把程序描述为：

程序＝数据结构＋算法

这说明了数据结构和算法对程序的重要性。设计一个合理的数据结构可以简化算法，而好的算法又可以提高程序的执行效率。

计算机可以直接执行的程序称为可执行程序（其扩展名一般为 EXE、COM），它包含的主要部分是二进制编码的机器指令和数据。机器指令直接控制计算机的每一个部件的基本动作，机器指令的表达方式称为“机器语言”。可执行程序通常以文件方式存放在磁盘上，当需要执行某程序时，必须把该程序装入内存。

1.3.2　程序设计

程序设计是根据计算机要完成的任务进行数据结构和算法的设计，并且编写其程序代码，然后进行调试，直到得出正确结果。其基本过程如下：

① 分析问题，明确要解决的问题和要实现的功能。

② 将具体问题抽象为数学问题，建立数学模型，确定合适的解决方案。

③ 确定数据结构，并根据数据结构设计相应的算法，写出算法描述。

④ 编写程序。

⑤ 调试并运行程序，直到得到正确结果。

程序设计方法经历了由传统的结构化程序设计（面向过程）到面向对象的设计。结构化程序设计采用模块分解与功能抽象和自顶向下、分而治之的方法，有效地将一个较复杂的程序设计任务分解成许多易于控制和处理的子程序（模块）。各模块之间尽量相对独立，便于开发和维护。结构化程序设计在整个 20 世纪 70 年代的软件开发中占绝对统治地位。

20 世纪 70 年代末期，随着计算机科学的发展和应用领域的不断扩大，对计算机技术的要求越来越高。结构化程序设计语言和结构化分析与设计已无法满足用户需求的变化，于是出现了面向对象的程序设计技术。面向对象的程序设计方法不仅吸收了结构化程序设计的思想，而且克服了结构化程序设计中数据与程序分离的缺点，模拟自然界认识和处理事务的方法，将数据和对数据的操作方法放在一起，形成一个对象，使对象成为程序系统的基本单位。面向对象的程序设计技术更加有利于程序的调试和维护，大大提高了程序的可重用性和修改、扩充程序的效率。

1.4　算法及其表示方法

算法与数据结构是计算机程序的两大基础，数据结构是为了研究数据运算而存在的；算法是为了实现数据运算，即实现数据的逻辑关系变化，或者是在这个结构上得到一个新的信息而存在的。数据结构与算法的实质不仅表现在两者互为依存，还体现在提高计算机效率的作用上。数据结构直接影响计算机进行数据处理的效率，而算法的好坏也直接影响计算机的效率。计算机科学家沃思对程序的描述说明了算法与数据结构在编制程序中的地位及重要性。

1.4.1　算法的特性与要求

算法是指为解决某个特定问题而采取的确定且有限的步骤。一个算法应该具有以下五个特性：

（1）确定性。算法中的每一个规则、每一个操作步骤都应当是确定的，不能有二义性，对于相同的输入应该有相同的输出结果。

（2）有穷性。一个算法必须在执行有限步骤后结束。也就是说，任何算法都必须在有限的时间内完成，而且应该在合理的时间内完成。

（3）有零个或多个输入。算法中可以没有数据输入，也可以同时输入多个需要处理的数据。

（4）有一个或多个输出。一个算法执行结束后必须有结果输出，否则该算法就没有实际意义。

（5）可执行性。算法的每一步操作都应该是可执行的。例如，当B=0时，A/B就无法执行，不符合可执行性的要求。

要设计一个好的算法通常要考虑以下要求：

（1）正确。算法的执行结果应当满足预先规定的功能和性能要求。

（2）可读。一个算法应当思路清晰、层次分明、简单明了、易读易懂。算法主要是为了人的阅读、理解和交流，其次才是机器执行。

（3）健壮。当输入数据不合法时，应能适当地作出反应或进行处理，而不会产生莫名其妙的输出结果。

（4）高效与低存储量。效率指的是算法执行的时间，存储量需求是指算法执行过程中所需的最大存储空间。同一个问题如果有多种算法可以解决，执行时间短的算法效率高，而效率与低存储量需求都与问题的规模有关。

1.4.2 算法描述

算法的描述就是用文字或图形把算法表示出来。常用的描述方法有：自然语言、流程图、N-S流程图、伪代码等。

1. 自然语言

自然语言就是人们日常使用的语言，可以是汉语、英语或其他语言。用自然语言描述算法通俗易懂，但也存在如下缺点：

① 往往要用一段较冗长的文字才能表达清楚要进行的操作；

② 容易出现“歧义性”，往往要根据上下文才能正确判断出它的含义，不太严谨；

③ 如果用自然语言描述的算法是顺序执行的，还比较容易理解，当算法中包含了判断和转移等步骤时，用自然语言描述就不容易理解。

【例1.1】 求任意3个正整数a, b, c中的最大者。

用自然语言描述的算法如下：

① 输入a, b, c；

② a和b比较，若a>b则a=>max，否则b=>max；

③ c和max比较，若c>max，则c=>max；

④ 输出max。

2. 传统流程图

传统流程图是用一些几何图形框、线条和文字来描述各种操作，是使用最早的算法和程序描述工具。美国国家标准化协会规定了一些常用流程图符号，如图1.1所示。