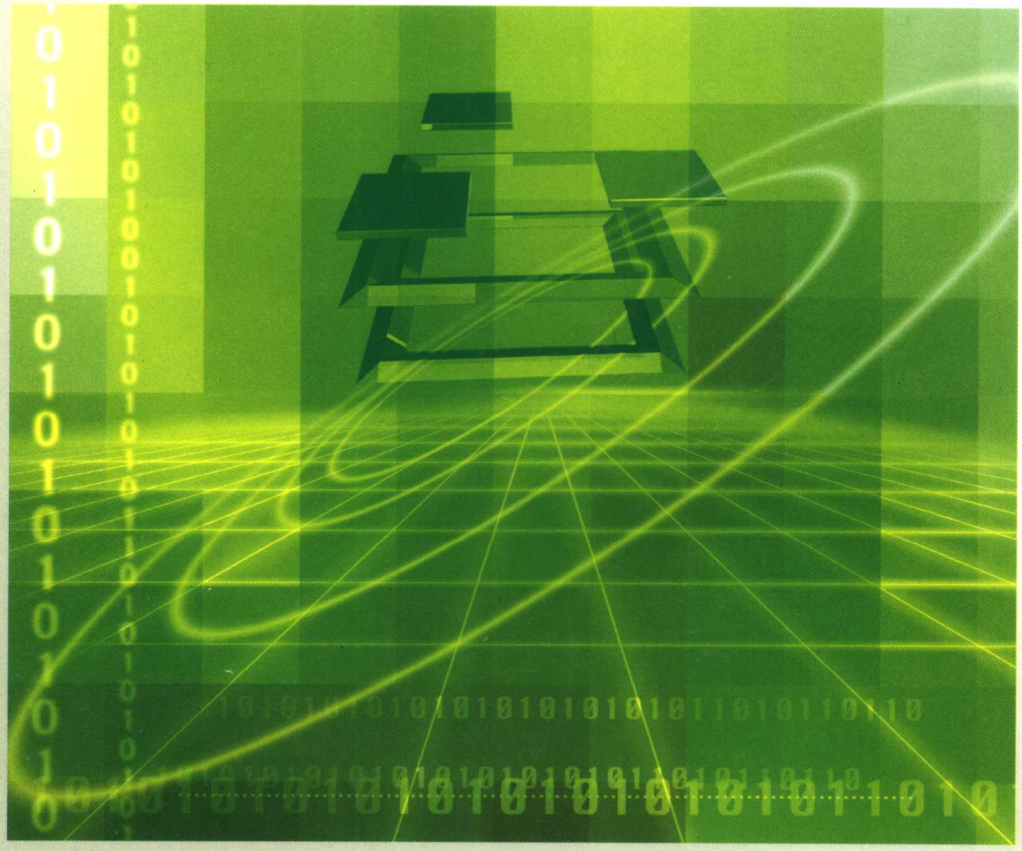
21世纪高校计算机系列规划教材

C语言程序设计教程

李清政　叶斌　雷辉　陈世强　主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21世纪高校计算机系列规划教材

C语言程序设计教程

主　编　李清政　叶　斌
雷　辉　陈世强
编　著　沈济南　周慧华
胡俊鹏　陈自根

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书对C语言的基本知识、数据类型、语法功能、使用特性、结构化程序设计方法以及常用算法及其应用作了较为系统而详细的介绍。书中还将数据结构与算法设计技术有机地结合起来，并以结构化程序设计思想贯穿全书。

本书内容充实、体系完整、思路清晰、概念准确、选材新颖、注重实用，是作者积多年教学经验编写而成的。在讲授本教材时，应结合实例分析，注意精讲多练、讲练结合。教学安排建议讲授40学时，实验24学时；也可根据教学实际选取教材的内容进行讲授。

本书可作为高等院校本科计算机程序设计课程教学用书，也可作为计算机应用开发人员的参考书或培训教材。

图书在版编目（CIP）数据

C语言程序设计教程/李清政等主编. —北京：中国铁道出版社，2006.7
（21世纪高校计算机系列规划教材）
ISBN 7-113-07286-0

Ⅰ.C… Ⅱ.李… 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（2006）第090158号

书　　名：C语言程序设计教程
作　　者：李清致　叶　斌　雷　辉　陈世强　等
出版发行：中国铁道出版社（100054，北京市宣武区右安门西街8号）
策划编辑：严晓舟　徐海英
责任编辑：苏　茜　谢立和　王慧亮
封面设计：薛　为
封面制作：白　雪
责任校对：张国成
印　　刷：河北省遵化市胶印厂
开　　本：787×1092　1/16　　**印张**：17.25　**字数**：403千
版　　本：2006年8月第1版　　2006年8月第1次印刷
印　　数：1～6 000册
书　　号：ISBN 7-113-07286-0/TP·1984
定　　价：29.00元

前　言

作为现代大学生，要提高程序设计和软件开发能力，选择C语言作为学习程序设计的基础语言仍是一个不错的主意。C语言概念简洁，提供的数据类型和运算符丰富，表达能力强，使用灵活方便，目标代码执行效率高，程序结构性强，可读性和可移植性好，能充分体现结构化程序设计的风格和特点。C语言既有高级语言的优点，又具有汇编语言的许多功能。因此，C语言既适于编写应用软件，又能用于开发系统软件，是目前功能强大，应用面广，影响力强的程序设计语言之一。

本教材在全面而系统地介绍C语言的基本概念、语法功能、使用特性以及结构化程序设计方法和技能的基础上，并通过具体实例分析，力求让学生理解和掌握数据结构、算法的概念及其应用；着重培养学生良好的程序设计思想和编程能力；为学生进一步学习、更好地从事软件开发和工程应用打下坚实的基础。

本书的作者是多年在教学一线从事C语言程序设计以及计算机课程教学的老师。根据多年来的教学心得和教改探索，本教程在编排体系、教学内容、教学目标、教学模式设计等方面，注重以现代教育理论为指导，以应用分析为基础，以多媒体教学为手段，力求体现“精讲多练”、“循序渐进”及程序设计语言教学的规律。书中所有例题和习题，也是经过精心挑选、编写和调试通过的，具有一定的代表性。

为了帮助读者学习本书，我们另编写了一本《C语言程序设计实验指导与题解》，可与本书配套使用。另外，在使用本教材时，为配合教师采用多媒体教学，也为方便读者自学，我们还将制作与本教材配套的电子教案、实例代码、实验解答以及相关电子文档，并陆续在网上发布，网址为：http//jsjjpkc.wephp.cn；在使用过程中若有问题，也可与作者交流，E-mail：yebin88@126.com。

本书由李清政、叶斌、雷辉、陈世强主编，由沈济南编写第1章、第11章；叶斌编写第2章；雷辉编写第3章；陈世强编写第4章、附录；周慧华编写第5章、第6章；胡俊鹏编写第7章、第10章；陈自根编写第8章、第9章。全书由叶斌负责统稿。

本书的编写得到了各级领导以及同行的热情关心和大力支持，在此表示感谢。

由于时间仓促，加之编者水平有限，书中错误和不当之处难免，恳请读者批评指正。

编　者

2006年6月

目　录

第 1 章　C 语言及程序设计概述

本章从程序设计语言的发展入手，介绍 C 语言的基本特点、算法及其特点与表示、C 语言程序的基本结构、语法单位以及 C 语言的集成开发环境 Microsoft Visual C++ 6.0 和 Turbo C 2.0。

1.1　程序与程序设计语言

1.1.1　程序的概念

计算机是一种具有内部存储能力的自动、高效的电子设备，它最本质的使命就是执行指令所规定的操作。如果需要计算机完成什么工作，只要将其步骤用多条指令的形式描述出来，并把这些指令存放在计算机的内部存储器中，需要执行时就向计算机发出一个简单的命令，计算机就会自动地逐条执行操作，全部指令执行完毕后即得到预期的结果。这种可以被连续执行的一条条指令的集合称为计算机的程序。也就是说，程序是计算机指令的序列，编写程序的工作就是为计算机安排指令序列。

但是，指令是由二进制编码组成，用它编写程序既难记忆，又难掌握，所以，计算机工作者就研制出了各种计算机能够识别、人们又方便使用的计算机语言，程序就是用计算机语言来编写的。因此，计算机语言通常被称为“程序语言”，一个计算机程序总是用某种程序语言编写的。

1.1.2　程序设计语言的发展

计算机系统由硬件系统和软件系统两大部分构成，硬件是物质基础，而软件则是计算机的灵魂，没有软件，计算机是一台“裸机”，什么也不能做；有了软件，才能成为一台真正的“电脑”。所有的软件，都是用计算机语言编写的。

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．机器语言

电子计算机所使用的语言是由“0”和“1”组成的二进制数，二进制数是计算机语言的基础。计算机发明之初，人们只能用计算机语言去命令计算机干这干那，一句话，就是写出一串串由“0”和“1”组成的指令序列交由计算机执行，这种语言，就是机器语言。使用机器语言的缺点是在程序有错需要修改时是非常麻烦的。而且，由于每台计算机的指令系统往往各不相同，也就是说，在一台计算机上执行的程序，要想在另一台计算机上执行，必须另编程序，这样就造成了重复工作。但由于使用的是针对特定型号计算机的语言，故而运算效率是所有语言中最高的。机器语言，是第一代计算机语言。

2．汇编语言

为了减轻使用机器语言编程的痛苦，人们对机器语言进行了一种改进：用一些简洁的英文字母、符号串（称为指令助记符）来替代一个特定指令的二进制串。例如，用“ADD”代表加法，“MOV”代表数据传递等，这样一来，人们很容易读懂并理解程序在干什么，纠错及维护都变得方便了，这种程序设计语言就称为汇编语言，即第二代计算机语言。然而计算机是识别这些符号的，这就需要一个专门的程序，专门负责将这些符号翻译成二进制数的机

器语言，这种翻译程序被称为汇编程序。

汇编语言同样依赖于机器硬件，移植性不好，但效率却十分高，针对计算机特定硬件而编制的汇编语言程序，能准确发挥计算机硬件的功能和特长，程序精炼而质量高，所以至今仍是一种被人们广泛应用的软件开发工具。

3．高级语言

在最初与计算机交流的痛苦经历中，人们意识到，应该设计一种接近于数学语言或人的自然语言，同时又不依赖于计算机硬件，编出的程序能在所有机器上通用。经过多年的研制，在 1954 年，第一个完全脱离机器硬件的高级语言——FORTRAN 问世了。目前影响较大、使用较普遍的语言有 FORTRAN、ALGOL、COBOL、BASIC、LISP、SNOBOL、PL/1、Pascal、C、PROLOG、Ada、C++、Visual C++、Visual Basic、Delphi、Java 等。

高级语言的发展也经历了从早期语言到结构化程序设计语言，从面向过程到非过程化程序语言的过程。相应地，软件的开发也由最初的个体手工作坊式的封闭式生产，发展为产业化、流水线式的工业化生产。

高级语言的下一个发展目标是面向应用，也就是说，只需要告诉程序你要干什么，程序就能自动生成算法，自动进行处理，这就是非过程化的程序语言。

1.1.3 C 语言的发展及特点

1．C 语言的发展概况

C 语言是在 20 世纪 70 年代初问世的。1978 年由美国电话电报公司（AT&T）贝尔实验室正式发表了 C 语言。同时由 B.W.Kernighan 和 D.M.Ritchit 合著了著名的《The C Programming Language》一书，通常简称为《K&R》，也有人称之为《K&R》标准。但是，在《K&R》中并没有定义一个完整的标准 C 语言，后来由美国国家标准学会在此基础上制定了一个 C 语言标准，在 1983 年发表。通常称之为 ANSI C。

2．C 语言的特点

C 语言发展如此迅速，而且成为最受欢迎的语言之一，主要因为它具有强大的功能。许多著名的系统软件，如 DBASE III PLUS、DBASE IV都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序，就更能显示 C 语言的优势了，像 PC-DOS、Wordstar 等就是用这种方法编写的。归纳起来 C 语言具有下列优点：

（1）语言简洁、紧凑，使用方便、灵活。C 语言共有 32 个关键字，9 条控制语句，源程序书写格式自由。

（2）具有结构化的控制语句，以函数作为程序模块以实现程序的模块化。

（3）数据类型丰富。C 语言除具有基本数据类型如整型（int）、实型（float 和 double）、字符型（char）外，还有各种构造类型。利用这些数据类型可以实现复杂的数据结构，如堆栈、队列和链表等。

（4）允许直接对位、字节和地址进行操作，能实现汇编语言的大部分功能。

（5）可直接操纵硬件。C 语言集高级语言和低级语言的功能于一体，既可用于系统软件的开发，也适合于应用软件的开发。

（6）生成的目标代码质量高，程序执行效率高。

（7）可移植性好（较之汇编语言）。基本可以不做任何修改，就能运行于各种型号的计算机和各种操作系统环境。

C 语言有如下不足：

① 语法限制不严格，编程者无法过多地依赖 C 编译程序去查错。

② 缺少实时检查：如数组越界等。

3．C 语言版本

目前最流行的 C 语言有以下几种：

（1）Microsoft C 或称 MS C

（2）Borland Turbo C 或称 Turbo C

（3）AT&T C

这些 C 语言版本不仅实现了 ANSI C 标准，而且在此基础上各自作了一些扩充，使之更加方便、完美。

1.2　程序设计方法

1.2.1　结构化程序设计方法

结构化程序设计方法是公认的面向过程编程应遵循的基本方法和原则。结构化程序设计方法主要包括：

（1）只采用 3 种基本的程序控制结构来编写程序，从而使程序具有良好的结构。

（2）程序设计自顶而下。

（3）用结构化程序设计流程图表示算法。有关结构化程序设计及方法有一整套不断发展和完善的理论和技术，对于初学者来说，完全掌握是比较困难的。但在学习的起步阶段就了解结构化程序设计的方法，学习好的程序设计思想，对今后的实际编程很有帮助。

1．结构化程序设计特征

结构化程序设计的特征主要有以下几点：

（1）以 3 种基本结构（顺序结构、分支结构、循环结构，详细内容参考第 3 章）的组合来描述程序。

（2）整个程序采用模块化结构。

（3）有限制地使用 goto 转移语句，在非用不可的情况下，也要十分谨慎，并且只限于在一个结构内部跳转，不允许从一个结构跳到另一个结构，这样可缩小程序的静态结构与动态结构执行过程之间的差异，使人们能正确理解程序的功能。

（4）以控制结构为单位，每个结构只有一个入口，一个出口，各单位之间接口简单，逻辑清晰。

（5）采用结构化程序设计语言书写程序，并采用一定的书写格式使程序结构清晰，易于阅读。

（6）注意程序设计风格。

2．自顶而下的设计方法

结构化程序设计的总体思想是采用模块化结构，自上而下，逐步求精。即首先把一个复

杂的大问题分解为若干个相对独立的小问题。如果小问题仍然很复杂，则可以把这些小问题又继续分解成若干个子问题，这样不断地分解，使得小问题或子问题简单到能够直接用程序的三种基本结构表达为止。然后，对应每一个小问题或子问题编写出一个在功能上相对独立的程序块，这种像积木一样的程序块被称为模块。每个模块各个击破，最后再统一组装，这样，对一个复杂问题的解决就变成了对若干个简单问题的求解。这就是自上而下，逐步求精的程序设计方法。

确切地说，模块是程序对象的集合，模块化就是把程序划分成若干个模块，每个模块完成一个确定的子功能，把这些模块集中起来组成一个整体，就可以完成对问题的求解。这种用模块组装起来的程序被称为模块化结构程序。在模块化结构程序设计中，采用自上而下，逐步求精的设计方法，便于对问题的分解和模块的划分，所以，它是结构化程序设计的基本原则。

1960 年，结构化程序设计思想诞生。C 和 Pascal 等语言都大力提倡这种程序设计的方法。结构化程序设计语言使得编写较复杂的程序变得容易。但是，一旦某个项目达到一定规模，即使使用结构化程序设计的方法，局势仍将变得不可控制。

在面向过程的程序设计方法中，问题被看作一系列将被完成的任务，如读、计算和打印，许多函数用于完成这些任务，问题的焦点集中于函数。这种程序设计的基本任务是编写计算机执行的指令序列，并把这些指令以函数的方式组织起来。

1.2.2 面向对象程序设计方法

在程序设计方法发展过程中，每一次重大突破都使程序员可以应对更大的复杂性。为了解决这个问题，面向对象程序设计方法应运而生。

提出面向对象程序设计（Orient Object Programming，OOP）方法的主要目的是弥补面向过程程序设计方法中的一些缺点。OOP 把数据看作程序开发中的基本元素，并且不允许它们在系统中自由流动。它将数据和操作这些数据的函数紧密地连接在一起，并保护数据不会被外界的函数意外改变。

面向对象程序设计在程序设计模式中是一个新的概念，对于不同的人可能意味着不同的内容。面向对象程序设计的定义是“面向对象程序设计是一种方法，这种方法为数据和函数提供共同的独立内存空间，这些数据和函数可以作为模板以便在需要时创建类似模块的拷贝。这样的程序设计方法称为面向对象程序设计。”

从以上定义可以看到，一个对象被认为是计算机内存中的一个独立区间，在这个区间中保存着数据和能够访问数据的一组操作。因为内存区间是相互独立的，所以对象可以不经修改就应用于多个不同的程序中。

1.3 算法及其表示

程序规定了计算机执行的动作和动作的顺序。如同开会的议程，每周的课程安排表一样。一个程序应包括以下两方面的内容：

（1）对数据的描述。在程序中要指定数据的类型和数据的组织形式，即数据结构。

（2）对操作的描述。即操作步骤，也就是算法。

数据是操作的对象，操作的目的是对数据进行加工处理，以得到期望的结果。作为程序设计人员，必须认真考虑和设计数据结构和操作步骤。著名的计算机科学家 Nikiklaus Wirth 提出了一个公式：

程序=数据结构+算法

开设本课程的目的是使同学知道怎样编写一个 C 程序，进行编写程序的初步训练，因此，只介绍算法的初步知识。

1.3.1　算法的概念

在日常生活中做任何一件事情，都是按照一定规则，一步一步进行的。例如在工厂中生产一台机器，先把零件按一道道工序进行加工，然后，又把各种零件按一定规则组装成一台完整机器，它们的工艺流程就是算法；在农村中种庄稼有耕地、播种、育苗、施肥、中耕、收割等各个环节，这些栽培技术也是算法。总之，在任何这些数值计算或非数值计算的过程中所采取的方法和步骤，都称之为算法。

计算机解决问题的方法和步骤，就是计算机的算法。计算机用于解决数值计算，如科学计算中的数值积分、解线性方程等计算方法，就是数值计算的算法；用于解决非数值计算如用于管理、文字处理、图形图像等排序、分类、查找方法，就是非数值计算的算法。

算法并不给出问题的精确的解，只是说明怎样才能得到解。每一个算法都是由一系列的操作指令组成的。这些操作包括加、减、乘、除、判断、置数等，由顺序结构、分支结构和循环结构组成。所以研究算法的目的就是研究怎样把各种类型的问题的求解过程分解成一些基本的操作。

算法写好之后，要检查其正确性和完整性，再根据它编写出用某种高级语言表示的程序。程序设计的关键就在于设计出一个好的算法。所以，算法是程序设计的核心。

1.3.2　算法的组成要素

1．操作

每个操作的确定不仅取决于问题的需求，还取决于它们取自哪个操作集，它与使用的工具系统有关。在高级语言中所描述的操作主要包括：算术运算（+、-、*、/）、逻辑运算（“与”、“或”、“非”等）、关系运算（==、>=、<=、>、<、!=等）、函数运算、位运算和 I/O 操作。计算机算法是由这些操作所组成的。

2．控制结构

控制结构控制组成算法的各种操作的执行顺序。结构化的程序设计要求一个程序只能由三种基本结构（顺序结构、选择结构和循环结构）或其派生出来的结构组成。1966 年 Bohm 和 Jacopini 证明出由这三种基本结构可以组成任何复杂结构的算法。

（1）顺序结构。顺序结构中的语句是按书写的顺序执行的，即语句的执行顺序与书写顺序一致。这是一种理想的结构，但只有这样的结构不可能处理复杂的问题。

（2）选择结构。最基本的选择结构是当程序执行到某一语句时，要进行一下判断，从两种路径中选择一条。例如，要在两个数 a、b 中选择一个较大的数就要经过比较判断，决定输出 a 或 b。

（3）循环结构。这种结构是将一条或多条语句重复地执行若干遍。电子计算机的优势

之一就是速度快，当把一个复杂的问题用循环结构来实现时，就能充分地发挥计算机的高速度的优势。

1.3.3 算法的特性

算法有 5 个特性：

（1）有穷性。一个算法必须总是（对任何合法的输入值）在执行有穷步骤之后结束，且每一步都可在有穷时间内完成。

（2）确定性。算法中的每一步操作的内容和顺序必须含义确切，不能有二义性。在任何条件下，算法只有唯一的一条执行路径，即对于相同的输入只能得出相同的输出。

（3）可行性。算法中的每一步操作都必须是可执行的，即算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的。

（4）输入。一个算法中有零个或多个输入。这些输入取自于某个特定的对象的集合，应在算法操作前提供。

（5）输出。一个算法中有一个或多个输出。这些输出是同输入有着某些特定关系的量。

1.3.4 算法的表示

原则上说，算法可以用任何形式的语言和符号来描述，通常有自然语言、程序语言、流程图、N-S 图、PAD 图、伪代码等。流程图、N-S 图和 PAD 图是表示算法的图形工具，其中，流程图是最早提出的用图形表示算法的工具，所以也称为传统流程图。它具有直观性强、便于阅读等特点，具有程序无法取代的作用。N-S 图和 PAD 图符合结构化程序设计要求，是软件工程中强调使用的图形工具。

因为流程图便于交流，又特别适合于初学者使用，对于一个程序设计工作者来说，会看会用传统流程图是必要的。本教材主要介绍和使用传统流程图的表示算法，只对 N-S 图作简要说明。

1. 用自然语言表示算法

用自然语言（人们日常使用的语言）表示算法，通俗易懂，但文字冗长，容易出现歧异。自然语言表示往往含义不太严格，要根据具体的语言环境才可以判断，计算机是不会对歧异进行过多的计较，只是编译的时候会报错。

【例 1-1】 求 5!。

原始方法：

Step1：求 1×2，得到结果 2。

Step2：将 Step1 中的结果乘以 3，得到新的结果 6。

Step3：将 Step2 中的结果乘以 4，得 24。

Step4：将 Step3 中的结果乘以 5 得到最后的结果 120。

改进：乘数都依据一定的规律进行递加（1～5），而且是连乘。可以设两个变量，一个代表被乘数，一个代表乘数，将每一步运算的结果保存在被乘数变量中。假设 p 为被乘数，i 为乘数。则：

Step1：使 p=1。

Step2：使 i=2。