“十二五”卓越工程技术人才培养系列教材

耿焕同 主编

C语言程序设计

江苏大学出版社
JIANGSU UNIVERSITY PRESS

“十二五”卓越工程技术人才培养系列教材

耿焕同 主编

C语言程序设计

江苏大学出版社
JIANGSU UNIVERSITY PRESS

图书在版编目(CIP)数据

C语言程序设计/耿焕同主编. —镇江：江苏大学出版社，2012.2
ISBN 978-7-81130-305-6

Ⅰ. ①C… Ⅱ. ①耿… Ⅲ. ①C语言—程序设计—高等学校—教材 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2012)第015134号

C语言程序设计

主　　编/耿焕同
责任编辑/段学庆　张小琴
出版发行/江苏大学出版社
地　　址/江苏省镇江市梦溪园巷30号(邮编:212003)
电　　话/0511-84443089
传　　真/0511-84446464
排　　版/镇江文苑制版印刷有限责任公司
印　　刷/扬中市印刷有限公司
经　　销/江苏省新华书店
开　　本/787 mm×1 092 mm　1/16
印　　张/19.75
字　　数/474千字
版　　次/2012年2月第1版　2012年2月第1次印刷
书　　号/ISBN 978-7-81130-305-6
定　　价/35.00元(含光盘)

内容简介

C 语言作为结构化程序设计语言的典型代表，非常适合作为程序设计技术的入门语言，更为重要的是，C 语言在应用上不仅适合软件开发，而且能够直接对各类可编程器件进行底层操作。

本书以 ISO C89 语言规范为蓝本，循序渐进，深入浅出，系统地讲解从语法到问题编程求解的各个环节，内容包括：程序设计理论基础篇、C 语言程序设计基础篇、C 语言程序设计能力篇。

本书紧紧结合 C 语言的学习方法和学生的学习特点，科学设计、精心组织教学内容，以浅显易懂的语言进行撰写，并配有大量的图解、例题和程序实例等，使非计算机专业人员也能快速地理解和掌握编程的技巧与精髓，熟练掌握应用 C 语言进行编程的基本技能。

本书以学生编程能力的培养为设计重点，巧妙选择典型例题，并配以问题分析及程序分析等内容，丰富了教材内涵，提高学生的学习兴趣。本书语言表达严谨、流畅，实例丰富，不仅适合各类高校举办的二级独立学院理工类本科学生学习，也非常适合其他层次学生学习，同时还可作为编程人员的自学教材和全国计算机等级考试（C 语言）的参考教材。

为便于读者学习和课程教学的需要，本书附有光盘，光盘内容包括本书所有例题的源程序和教学课件。

前　言

从1971年诞生到现在，C语言已经成为最重要和最流行的高级程序设计语言之一。C语言学习之所以经久不衰，根源在于C语言具有方便性、灵活性和通用性等特点。在其应用方面，除了适合各种类型的软件开发外，程序员还可直接对可编程硬件操作。因此，C语言不仅是计算机学科重要的核心课程，而且是其他理工科专业计算机基础知识的必修课。

随着时代的进步，掌握一种编程语言已经成为现代科技人员的一项基本技能。因此近年来，学习和掌握C语言的需求越来越迫切，特别是对于在校理工科大学生，程序设计能力越来越成为现代科技人才的一种必备能力。

虽然市面上有很多关于C语言程序设计的书籍，但是存在着诸如从抽象的语法开始学习，或是用枯燥的数学问题作为实例等不足，这在某种程度上偏离了程序设计的核心，不仅容易挫伤初学者学习程序设计的信心，而且会导致初学者对程序设计缺乏兴趣。我们根据多年从事C语言一线教学工作积累的经验和教学心得，充分借鉴现有C语言书籍的优点，采用抽象知识的学习方法，即以“先问为什么学、如何学习、最后学什么”的内容组织方式，科学设计教程和巧妙组织本书内容。

本书的主要特色和创新之处有：

（1）理念新颖。围绕现代理工科专业技术人才的培养目标，取舍恰当，精心选择内容和科学编排，力求使非计算机专业的学生系统、全面、快速地掌握C语言程序设计的方法，为后续应用开发建立坚实的语言基础。

（2）针对性强。针对二级独立院校本科学生重实践的学习特点，兼顾全国二级等级考试大纲（C语言版）的要求，采用先理论后实践的学习规律，变抽象为具体的学习策略，增强学习的目的性，合理组织C语言知识点。

（3）科学组织。全书内容包括程序设计基础、C语言基础和能力提高的知识内容，做到结构严谨，概念准确，教材内容组织合理，语言使用规范，符合教学规律。

（4）注重能力。围绕能力培养，从易于读者学习的角度出发，以实例和易理解的图形方式阐述枯燥的理论知识，并在实例讲解中详细列出问题分析、流程图、C语言程序代码、程序分析和运行结果等内容，其目的是帮助读者学会如何编程，切实做到提高应用能力。

本书循序渐进，由方法到实践，由入门到掌握，共分为3大篇。第1篇为程序设计基础篇，为初学者介绍程序设计的方法、算法、开发环境和相关的程序设计基础知识等；第2篇为C语言程序设计基础篇，从最简单的C语言程序开始认识，逐步介绍C语言程序的构成要素，使略有计算机基础的人都能容易地学会C语言编程；第3篇为C语言程序设计能力篇，主要介绍C语言中的数组、指针、函数以及文件操作等内容。

本教材由南京信息工程大学滨江学院耿焕同教授主持编写，并负责对全书进行统稿和主审。其中，耿焕同老师负责编写第 1 ~ 4 章；陈遥老师负责编写第 6 ~ 8 章；朱节中老师负责编写了第 5，11 和 12 章；李振宏老师负责编写了第 9，10 章和附录；姜青山老师负责编写了第 13 ~ 15 章。

本书的编写得到南京信息工程大学滨江学院二期教改课题（No. 2011JC0001）资助，也得到诸多专家和领导的大力支持与指导，在此表示衷心的感谢。

由于编者水平有限，书中难免有错误和不足之处，恳请专家和广大读者批评指正。

编　者

2012 年 1 月

目　录

第1篇　程序设计基础

第3篇 C语言程序设计能力

第 1 篇　程序设计基础

第 1 章　程序设计方法学

1.1　程序设计方法学简介

众所周知，随着科技和信息技术的迅猛发展，越来越多的工作和业务由程序进行支撑与管理，如数值天气预报、数字化校园以及网上购物等。那究竟什么是程序呢？通俗来讲，程序是用来控制计算机操作的代码，而程序设计的目的是利用计算机对现实问题进行求解。计算机科学家、图灵奖获得者尼克劳斯·威茨(Niklaus Wirth)教授对程序进行了经典定义：

程序＝算法＋数据结构

此公式对计算机科学的影响程度类似于物理学中爱因斯坦的 $E=mc^2$，它提示了程序的本质。

随着软件产业的迅猛发展和软件开发的工程化进程加快，程序与软件开发环境的关系越来越紧密，开发工具的选择对程序的开发效率有着重大影响，有时会获得事半功倍的效果。因此，可对程序的定义进行扩充：

程序＝算法＋数据结构＋开发环境

本书重点讨论 C 语言开发环境，并不侧重复杂的算法和数据结构，目的是使读者利用 C 语言设计简单程序，建立与计算机之间的会话交流，培养一定的编程思想和动手编程能力的软件人才。

程序设计方法学是探讨程序设计理论和方法的学科，用以指导程序设计各阶段工作的原理和原则，以及由此提出的设计技术。程序设计方法学起源于 20 世纪 70 年代，主要包括程序理论、研制技术、支持环境、工程规范和自动程序设计等课题。程序设计方法学的发展、软件的发展以及编程语言的发展三者之间有着密切的关系；通过对其研究，可不断地提高编程人员的程序设计水平，丰富程序人员的思维方法；而问题求解规模和复杂性大大地促进了程序设计技术的发展；反过来，程序设计的提高也推动了程序设计方法学这一学科的不断发展。

通常而言，程序设计方法学的概念有狭义和广义之分。狭义的程序设计方法学是指传统的有关结构化程序设计的理论、方法和技术；广义的程序设计方法学概念包括了

程序设计语言和程序设计的所有理论和方法。特别是在结构化程序设计的研究逐步衰退以后，程序设计方法学成为一个笼统的概念。随着软件产业的快速发展，又对程序设计方法提出了更高的要求，如设计过程简单化、代码跨平台化、代码重用化等，促使程序设计方法学成为一门学科。因此，首先要弄清楚程序设汁方法学的基本研究目标。

从学科定义来讲，程序设计方法学的目标是能设计出可靠、高效、易读而且代价合理的程序。更通俗地讲，程序设计方法学的最基本目标是通过对程序本质属性的研究，说明什么样的程序是一个“优秀”的程序，怎样才能设计出“优秀”的程序。

一般的程序设计过程是借助某种编程语言对求解问题的计算机算法进行编程实现，其产出是软件产品(俗称程序)，其功能是利用计算机求解问题，因此在程序设计时，最重要的是程序的正确性和程序的执行效率。程序的正确性和执行效率是由程序的结构和算法决定的，当然也与程序的易读性、可维护性有密切的关系。程序设计的一般过程应包括：分析实际问题并抽象，利用数学建模技术构建问题的数学模型，借助计算方法和数据模型构造合适数据结构，进而设计算法，最后借助计算机语言实现算法并形成程序。

程序设计方法大致经历如下主要阶段：手工作坊式、结构化、模块化、面向对象等。下面以主流的结构化程序设计和面向对象程序设计为例，分别讲解它们的设计方法。

1.2 结构化程序设计方法

1.2.1 概 述

迪克斯特拉(E. W. Dijkstra)在 1969 年提出了结构化程序设计方法，是以模块化设计为中心，将待开发的软件系统划分为若干个相互独立的模块，这样就使完成每一个模块的工作变得简单且明确，为设计一些较大的软件奠定了良好的基础。由于模块相互独立，因此在设计其中一个模块时不会受到其他模块的牵连，故可将原来较为复杂的问题化简为一系列简单模块的设计。模块的独立性还为扩充已有的系统、建立新系统带来了极大方便，因此可以充分利用现有的模块作积木式的集成与扩展。

结构化程序的概念首先是从以往编程过程中无限制地使用转移语句而提出的。转移语句可以使程序的控制流程强制性地转向程序的任一处，在传统流程图中，用“很随意”的流程线来描述转移功能。如果一个程序中多处出现这种转移情况，将会导致程序流程无序可寻，程序结构杂乱无章，这样的程序是令人难以理解和接受的，并且容易出错。尤其是在实际软件产品的开发中，更多追求的是软件的可读性和可修改性，像这种结构和风格的程序是不允许出现的，因此规定程序的 3 种基本结构。

程序的顺序、选择和循环 3 种控制流程，就是结构化程序设计方法强调使用的 3 种基本结构。算法的实现过程是由一系列操作组成的，这些操作之间的执行次序就是程序的控制结构。1966 年，计算机科学家 Bohm 和 Jacopini 证明了这样的事实：任何简单或复杂的算法都可以由顺序结构、选择结构和循环结构这 3 种基本结构组合而成。所以，这 3 种结构就被称为程序设计的 3 种基本结构，也是结构化程序设计必须采用的结构。

结构化程序设计的基本思想是采用“自顶向下，逐步求精”的程序设计方法和“单入口单出口”的控制结构。“自顶向下，逐步求精”的程序设计方法从问题本身开始，经过逐步细化，将解决问题的步骤分解为由基本程序结构模块组成的结构化程序框图；“单入口单出口”的思想认为一个复杂的程序，如果它仅是由顺序、选择和循环 3 种基本程序结构通过组合、嵌套构成，那么这个新构造的程序一定是一个单入口单出口的程序，据此就很容易编写出结构良好、易于调试的程序。

因此，结构化程序设计具有以下优点：① 整体思路清楚，目标明确；② 设计工作中阶段性非常强，有利于系统开发的总体管理和控制；③ 在系统分析时可以诊断出原系统中存在的问题和结构上的缺陷。

结构化程序设计强调对程序设计风格的要求，因为程序设计风格主要影响程序的可读性。一个具有良好风格的程序应当注意以下几点：① 语句形式化。程序语言是形式化语言，需要准确，无二意性；② 程序一致性。保持程序中的各部分风格一致，文档格式一致；③ 结构规范化。程序结构、数据结构甚至软件的体系结构要符合结构化程序设计原则；④ 适当使用注释。注释是帮助程序员理解程序，提高程序可读性的重要手段；⑤ 标识符贴近实际。程序中数据、变量和函数等的命名原则是选择有实际意义的标识符，以易于识别和理解。

如何编写程序才算符合结构化程序设计方法呢？按照 1974 年世界著名科学家 D. Gries教授的分析，结构化程序设计应包括以下几个方面内容：

- 结构化程序设计是指导人们编写程序的一般方法。
- 结构化程序设计是一种避免使用 GOTO 语句的程序设计。
- 结构化程序设计是“自顶向下，逐步求精”的程序设计。
- 结构化程序设计是一种组织和编写程序的方法，利用它编写的程序容易理解和修改。
- 结构化程序设计是控制复杂性的整个理论和训练方法。
- 结构化程序的一个主要功能是使正确性的证明容易实现。
- 结构化程序设计将任何大规模和复杂的流程图转换为一种标准形式，使它们能够用几种标准形式的控制结构通过重复和嵌套来表示。

常用的结构化程序设计语言有：C 语言、FORTRAN 语言、Pascal 语言和 Basic 语言等。

简单地说，结构化程序设计有以下特征：

1. 模块化

(1) 把一个较大的程序划分为若干个函数或子程序，每一个函数或子程序总是独立成为一个模块；

(2) 每一个模块又可继续划分为更小的子模块；

(3) 程序具有一种层次结构。

【注意】 运用这种编程方法时，必须先对问题进行整体分析，避免想到哪里写到哪里。

2. 层次化

(1) 先设计第一层(即顶层)，然后步步深入，逐层细分，逐步求精，直到整个问题可

用程序设计语言具体明确地描述出来为止。

(2) 步骤:先对问题进行仔细分析,确定其输入、输出数据,写出程序运行的主要过程和任务;然后从大的功能方面把一个问题的解决过程分成几个子问题,每个子问题形成一个模块。

(3) 特点:先整体后局部,先抽象后具体。

3. 逐步求精

逐步求精是指对于一个复杂问题,不是一步就能编成一个可执行的程序,而是分步进行,具体如下:

第一步编出的程序最为抽象;

第二步编出的程序是把第一步所编的程序(如函数、子过程等)细化,较为抽象;

……

直到最后,第 n 步编出的程序即为可执行的程序。

所谓"抽象程序",是指程序所描述的解决问题的处理规则,是由那些"做什么(What)"操作组成,而不涉及这些操作"怎样做(How)"以及解决问题的对象具有什么结构,不涉及构造的每个局部细节。

这一方法原理是:对于某一个问题(或任务),程序员应立足全局考虑如何解决这一问题的总体关系,暂不涉及每个局部细节。在确保全局的正确性之后,再分别对每一个局部进行考虑。每个局部又是一个问题或任务,因而这一方法是自顶而下的,同时也是逐步求精的。

采用逐步求精方法的优点是:

(1) 便于构造程序。由这种方法产生的程序,其结构清晰、易读、易写、易理解、易调试、易维护。

(2) 适用于大任务、多人员设计,也便于软件管理。

逐步求精方法有多种具体做法,例如流程图方法、基于函数或子过程的方法等。

1.2.2 程序设计步骤

程序设计步骤包括:

(1) 分析问题

对要解决的问题,首先必须分析清楚问题的已知条件、所求的问题等,初步确定问题的求解思路和方法。

(2) 建立数学模型

从编程的角度,遵循编程思想,列出所有已知量,找出问题的求解目标,在对实际问题进行分析之后找出它的内在规律,以建立相应的数学模型。只有建立数学模型,才有可能利用计算机解决。

(3) 选择算法

建立数学模型后还不能立即着手编程序,必须选择合适的数据结构设计解决问题的算法。一般选择算法要注意以下几点:① 算法的逻辑结构尽可能简单;② 算法所要求的存储量尽可能少;③ 避免不必要的循环,减少算法的执行时间;④ 在满足题目条件要求下,使所需的计算量最小。