普通高等教育“十二五”规划教材

C语言程序设计基础

主编　冯克鹏　雷剑刚　李涛

电子科技大学出版社

普通高等教育“十二五”规划教材

C语言程序设计基础

主　编：冯克鹏　雷剑刚　李　涛
副主编：庄景明　高宏娟　周天宏　曹桂林

电子科技大学出版社

图书在版编目(CIP)数据

C 语言程序设计基础 / 冯克鹏,雷剑刚, 李涛主编.
- 成都: 电子科技大学出版社,2014.7
ISBN 978-7-5647-2362-0

Ⅰ. ①C… Ⅱ. ①冯… ②雷… ③李… Ⅲ. ①C 语言-L
程序设计 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 092585 号

C 语言程序设计基础

冯克鹏　雷剑刚　李涛　主编

出　　版: 电子科技大学出版社(成都市一环路东一段 159 号电子信息产业大厦　邮编:610051)
策划编辑: 汤云辉
责任编辑: 汤云辉
主　　页: www.uestcp.com.cn
电子邮箱: uestcp@uestcp.com.cn
发　　行: 新华书店经销
印　　刷: 北京市彩虹印刷责任有限公司
成品尺寸: 185mm×260 mm　　**印张** 19　**字数**　468 千字
版　　次: 2014 年 7 月第一版
印　　次: 2014 年 7 月第一次印刷
书　　号: ISBN 978-7-5647-2362-0
定　　价: 36.00 元

前　言

C语言是当今最流行的程序设计语言之一，它功能丰富，表达力强，使用灵活方便，应用面广，既具有高级语言的特点，又具有低级语言的特点，适合作为系统描述语言，既可以用来编写系统软件，也可以用来编写应用软件。因此，高校的高级语言程序设计课程，主要以C语言作为程序设计语言。

本书注重教材的可读性和适用性，每章开头部分介绍本章内容，书中附有大量的图表、程序，使读者能正确、直观地理解问题；例题由浅入深，强化知识点、算法、编程方法与技巧，并给出了详细的解释；为了适合计算机等级考试，在内容安排上完全符合计算机等级考试大纲要求；另外，本书各章还配套提供题型丰富的习题。

本书内容和结构体现了教学改革成果，作者根据多年的教学经验和多项教研课题的研究成果，构建了一个程序设计概念建立和编程思想培养的框架体系，总结提炼了学习本课程的重难点和解决方法，大部分样例都经过整理和组织，以便更好地理解掌握。

本书由冯克鹏、雷剑刚、李涛任主编，高宏娟、周天宏、曹桂林、贺桂娇任副主编。全书由冯克鹏统稿审定，第1、2、3、4、5章由高宏娟编写，第6、7、8、9、10章由李涛编写，第11章由冯克鹏编写，第12、15章由雷剑刚编写，第13章由庄景明编写，第14章由曹桂林编写。

本书为普通高等教育“十二五”规划教材，可作为高等院校本、专科生专业相关课程的教材，也可以作为计算机等级考试的自学教材或参考书。

由于作者的水平有限，本书肯定会有不尽如人意之处，错漏之处在所难免，敬请读者批评指正，以便我们及时修改。

作　者

2014年3月

目 录

第5章 循环结构

第6章 函 数

第7章 地址和指针

第8章 数 组

第9章 字 符 串

第10章 对函数的进一步讨论

第11章 C 语言中用户标识符的作用域和存储类

第 1 章　程序设计基础

本章教学目标

☆ 熟悉程序和程序设计的基本概念
☆ 了解 C 语言的发展及其特点
☆ 了解结构化程序设计的基本概念
☆ 熟悉 C 语言开发工具

本章重点内容

★ 程序和程序设计基本概念
★ 算法及描述
★ C 语言的发展及特点
★ C 语言的开发环境

1.1　程序与程序设计语言

一台计算机是由硬件系统和软件系统两大部分构成的，硬件是物质基础，而软件是计算机的灵魂。没有软件，计算机是一台“裸机”，是什么也干不了的；有了软件，计算机才有了生命，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。软件是包含程序的有机集合体，程序是软件的必要元素。软件可以用以下公式来表示：软件=程序+文档=数据结构+算法+文档。

任何软件都有可运行的程序，至少一个。比如：操作系统给的工具软件，很多都只有一个可运行程序。而 Office 是一个办公软件包，却包含了很多可运行程序。软件是程序以及开发、使用和维护所需要的所有文档的总称，而程序是软件的一部分。

1.1.1　程序

程序是用计算机语言描述的某一问题的解决步骤。计算机程序是软件开发人员根据用户需求开发的、用程序设计语言描述的适合计算机执行的指令序列。计算机本身是不会做任何工作的，它是按照程序中的有序指令来完成相应的任务。

由于计算机不能理解人类的自然语言，所以不能用自然语言编写计算机程序，只能用专门的程序设计语言来编写。人们借助计算机能够处理的语言，告诉计算机要处理哪些数据以及按什么步骤来处理，这便是程序设计。

为解决某一问题编写的程序不是唯一的，不同的用户编写程序的思路也不会完全一样，因此不同的程序有不同的执行效率，这涉及到程序的优化、程序所采用的数据结构以及算法等多

方面的因素。

1.1.2 程序设计语言

1946 年世界上第一台电子计算机问世以来，计算机科学及其应用的发展十分迅猛，计算机被广泛地应用于人类生产、生活的各个领域，推动了人类社会的进步与发展。特别是随着国际互联网（Internet）日益深入千家万户，传统的信息收集、传输及交换方式正被革命性地改变，我们已经难以摆脱对计算机的依赖，计算机已将人类带入一个新的时代——信息时代。掌握计算机的基本知识和基本技能已经成为人们应该具备的基本素质，缺乏计算机知识，就是信息时代的“文盲”。

对于理工科的大学生而言，掌握一门高级语言及基本的编程技能是必需的。学习计算机语言，是一种十分有益的训练方式，而语言本身又是与计算机进行交互的有力工具。

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

机器语言，是第一代计算机语言，属于低级语言的范畴。机器语言是用 0 和 1 这样的二进制代码表示的计算机能直接识别和执行的一种机器指令的集合。

由于机器语言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。机器语言具有灵活、直接执行和速度快等特点。

但是机器语言的使用复杂、烦琐、费时、易出差错，特别是在程序有错需要修改时，更是如此。由于每台计算机的指令系统往往各不相同，所以，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，造成了重复工作。

2. 汇编语言

汇编语言（Assembly Language）是面向机器的程序设计语言。在汇编语句中，用助记符（Memoni）代替操作码，用地址符号（Symbol）或标号（Label）代替地址码。比如，用“ADD”代表加法，“MOV”代表数据传递等等。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。

使用汇编语言编写的程序，机器不能直接识别，要由一种程序将汇编语言翻译成机器语言，这种起翻译作用的程序叫汇编程序。汇编程序是语言处理系统软件，汇编程序把汇编语言翻译成机器语言的过程称为汇编。

汇编语言同样十分依赖于机器硬件，移植性不好，但效率仍十分高，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，所以至今仍是一种常用而强有力的软件开发工具。

3. 高级语言

由于汇编语言依赖于硬件体系，且助记符量大难记，于是人们又发明了更加易用的所谓高级语言。在这种语言下，其语法和结构更类似普通英文，表示方法要比低级语言更接近于待解问题的表示方法，其特点是在一定程度上与具体机器无关，易学、易用、易维护。

1954 年，第一个完全脱离机器硬件的高级语言 FORTRAN 问世了，40 多年来，共有几百种高级语言出现，有重要意义的有几十种，影响较大、使用较普遍的有 FORTRAN、ALGOL、COBOL、BASIC、LISP、SNOBOL、PL/1、Pascal、C、PROLOG、Ada、C++、VC、VB、Delphi、JAVA 等。

从早期语言到结构化程序设计语言，从面向过程的程序设计语言到非过程化的程序设计语言，高级语言的发展经历了一个漫长的进化过程。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。

20 世纪 60 年代中后期，软件越来越多，规模越来越大，而软件的生产基本上是各自为战，缺乏科学规范的系统规划与测试、评估标准，其结果是大量耗费巨资建立起来的软件系统，由于含有错误而无法使用，甚至带来巨大损失，软件给人的感觉是越来越不可靠，以致几乎没有不出错的软件。这一切，极大地震动了计算机界，史称“软件危机”。人们认识到：大型程序的编制不同于写小程序，它应该是一项新的技术，应该像处理工程一样处理软件研制的全过程。程序的设计应易于保证正确性，也便于验证正确性。1969 年，提出了结构化程序设计方法，1970 年，第一个结构化程序设计语言—Pascal 语言出现，标志着结构化程序设计时期的开始。

20 世纪 80 年代初开始，在软件设计思想上，又产生了一次革命，其成果就是面向对象的程序设计。在此之前的高级语言，几乎都是面向过程的，程序的执行是流水线似的，在一个模块被执行完成前，人们不能干别的事，也无法动态地改变程序的执行方向，这和人们日常处理事物的方式是不一致的。对人而言是希望发生一件事就处理一件事，也就是说，不能面向过程，而应是面向具体的应用功能，即对象（object）。其方法就是软件的集成化，如同硬件的集成电路一样，生产一些通用的、封装紧密的功能模块，称之为软件集成块，它与具体应用无关，但能相互组合，完成具体的应用功能，同时又能重复使用。对使用者来说，只关心它的接口（输入量、输出量）及能实现的功能，至于如何实现的，那是它内部的事，使用者完全不用关心，C++、VB、Delphi 就是典型代表。

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。

1.2　算法及其描述

一个计算机程序应该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（Data-structure）。

（2）对操作的描述。即操作步骤，也就是算法（Algorithm）。

著名计算机科学家沃思提出一个公式：数据结构+算法=程序。实际上，一个程序除了以上两个主要的要素外，还应当采用程序设计方法进行设计，并且用一种计算机语言来表示。因此，算法、数据结构、程序设计方法和语言工具这四个方面是一个程序员所应具备的知识。可见，算法在计算机科学界与计算机应用界的地位。

1.2.1　算法的概念

算法就是为解决一个具体问题而采取的方法和有限步骤，或者是指对解题方案准确而完整的描述，是一系列解决问题的清晰指令，算法代表着用系统的方法描述解决问题的策略机制。如果一个算法有缺陷，或不适合于某个问题，执行这个算法将不会解决这个问题。

一个算法应该具有以下七个重要的特征：

（1）有穷性（Finiteness）：算法的有穷性是指算法必须能在执行有限个步骤之后终止；

（2）确切性（Definiteness）：算法的每一步骤必须有确切的定义；

（3）输入项（Input）：一个算法有 0 个或多个输入，以表示运算对象的初始情况，所谓 0 个输入是指算法本身定出了初始条件；

（4）输出项（Output）：一个算法有一个或多个输出，以反映对输入数据加工后的结果，没有输出的算法是毫无意义的；

（5）可行性（Effectiveness）：算法中执行的任何计算步骤都是可以被分解为基本的可执行的操作步，即每个计算步都可以在有限时间内完成（也称之为有效性）；

（6）高效性（Highefficiency）：执行速度快，占用资源少；

（7）健壮性（Robustness）：对数据响应正确。

一个算法的质量优劣将影响到算法乃至程序的效率。不同的算法可能会用不同的时间、空间或效率来完成同样的任务。算法分析的目的在于选择合适算法和改进算法。一个算法的评价主要从时间复杂度和空间复杂度来考虑。

1. 时间复杂度

算法的时间复杂度是指执行算法所需要的时间。一般来说，计算机算法是问题规模 n 的函数 $f(n)$，算法的时间复杂度也因此记做 $T(n)=O(f(n))$。

因此，问题的规模 n 越大，算法执行时间的增长率与 $f(n)$ 的增长率正相关，称作渐进时间复杂度（AsymptoticTime Complexity）。

2. 空间复杂度

算法的空间复杂度是指算法需要消耗的内存空间。其计算和表示方法与时间复杂度类似，一般都用复杂度的渐近性来表示。同时间复杂度相比，空间复杂度的分析要简单得多。

1.2.2 算法的描述

算法可以使用自然语言、伪代码、流程图等多种不同的方法来描述，它们的优势和不足可以简单地归纳如下。

1. 自然语言

优势：自然语言描述的算法通俗易懂，不用专门训练。

不足：由于自然语言的歧义性，容易导致算法执行的不确定性；自然语言的语句一般较长，导致描述的算法太长；当一个算法中循环和分歧较多时就很难清晰地表示出来；自然语言表示的算法不便翻译成计算机程序设计语言。

2. 伪代码

伪代码方式是用介于自然语言与计算机语言之间的文字及符号来描述算法。

优势：回避了程序设计语言严格、烦琐的书写格式，书写方便；同时具备格式紧凑、易于理解、便于向计算机程序设计语言过渡的优点。

不足：由于伪代码的种类繁多，语句不容易规范，有时会产生误读。

例 1-1 用伪代码描述“输出 x 的绝对值”的算法

若 x 为正

　　输出 x

否则

　　输出 $-x$

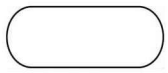
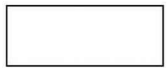
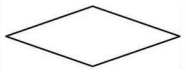
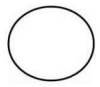
流程图

流程图是一个描述算法的控制流程和指令执行情况的有向图，是一种比较直观的描述方式。下面介绍几种常用的流程图符号，如表 1-1 所示。

优势：算法清晰简洁，容易表达选择结构，它不依赖于任何具体的计算机和计算机程序设计语言，从而有利于不同环境的程序设计。

不足：不易书写，修改起来比较费事，可以借助于专用的流程图制作软件来绘制和修改。

表 1-1　流程图中的常用符号

图形符号	符号名称	说明
	起止框	表示算法的开始或结束
	处理框	表示算法的某个处理步骤
	判断框	表示对给定条件进行判断，根据条件是否成立来决定如何执行
	输入/输出框	表明输入/输出操作
	流线	带箭头的直线，表示程序的流向
	连接圈	表示算法流向出口和入口连接点

例 1-2　用流程图描述以下算法：从键盘输入圆的半径，计算其面积和周长。

步骤：

（1）输入圆的半径 R

（2）$S=3.14*R*R$

（3）输出面积 S

说明：该算法中计算面积所需的初始数据半径 R 待定，要在程序运行后从键盘输入。算法描述如图 1-1 所示。

例 1-3　用流程图描述以下算法：求从键盘输入正方形的边长 a，求正方形的面积和周长。

步骤：

（1）输入正方形的边长

（2）$S=a*a$

（3）$c=4*a$

（4）输出面积 S，周长 C

说明：该算法中计算面积所需的初始数据边长 a 待定，要在程序运行后从键盘输入。算法描述如图 1-2 所示。

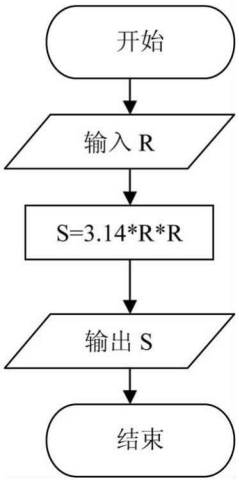


图 1-1

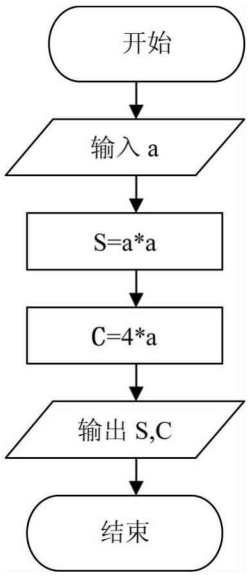


图 1-2

1.2.3 常用算法举例

1. 穷举法

穷举法又称枚举法，是一种简单而直接解决问题的方法。其基本思想是逐一列举问题所涉及的所有情形，并根据问题提出的条件检验哪些是问题的解，哪些应予排除。

2. 递归法

递归是设计和描述算法的一种有力的工具，在复杂算法的描述中被经常采用。能采用递归描述的算法通常有这样的特征：为求解规模为 N 的问题，设法将它分解成规模较小的问题，然后从这些小问题的解方便地构造出大问题的解，并且这些规模较小的问题也能采用同样的分解和综合方法，分解成规模更小的问题，并从这些更小问题的解构造出规模较大问题的解。特别地，当规模 N=1 时，能直接得解。

递归算法是一种效率比较低的算法，但是其优点是很明显的，它能够大幅度降低程序设计的复杂性，非常容易理解。但对时间复杂度要求较高的程序不适合用递归算法。

3. 回溯法

回溯法是一种选优搜索法，按选优条件向前搜索，以达到目标。但当探索到某一步时，发现原先选择并不优或达不到目标，就退回一步重新选择，这种走不通就退回再走的方法称为回溯法，而满足回溯条件的某个状态的点称为“回溯点”。

八皇后问题是一个古老而著名的经典案例。该问题由 19 世纪著名的数学家高斯在 1850 年提出：在 8 乘以 8 格的国际象棋上摆放八个皇后，使其不能互相攻击，即任意两个皇后都不能处于同一行、同一列或同一对角线上，问有多少种摆法。八皇后问题就能够用回溯法解决。

4. 贪心法

贪心算法（又称贪婪算法）是指，在对问题求解时，总是做出在当前看来是最好的选择。也就是说，不从整体最优上加以考虑，所做出的仅是在某种意义上的局部最优解。贪心算法不

是对所有问题都能得到整体最优解，但对范围相当广泛的许多问题能产生整体最优解或者是整体最优解的近似解。

5. *分治法*

在计算机科学中，分治法是一种很重要的算法。字面上的解释是“分而治之”，就是把一个复杂的问题分成两个或更多的相同或相似的子问题，再把子问题分成更小的子问题……直到最后子问题可以简单地直接求解，原问题的解即子问题的解的合并。这个技巧是很多高效算法的基础，如排序算法（快速排序，归并排序），傅立叶变换（快速傅立叶变换）等。

1.3　C 语言的发展及其特点

1.3.1　语言的发展历史

C 语言的发展颇为有趣，它的原型是 1960 年出现的 ALGOL 60 语言（也成为 A 语言）。和汇编语言相比，A 语言的可读性、可移植性较好，但离硬件远，不适合编系统程序。1963 年，剑桥大学将 ALGOL 60 语言发展成为 CPL（Combined Programming Language）语言。CPL 更接近硬件一些，但规模大。

1967 年，剑桥大学的 Matin Richards 对 CPL 语言进行了简化，于是产生了 BCPL 语言。1970 年，美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改，并为它起了一个有趣的名字“B 语言”。意思是将 CPL 语言煮干，提炼出它的精华。并且他用 B 语言写了第一个 UNIX 操作系统。但 B 语言过于简单，功能有限，并且没有数据类型。

而在 1973 年，B 语言也给人“煮”了一下，美国贝尔实验室的 D. M. RITCHIE 在 B 语言的基础上最终设计出了一种新的语言，他取了 BCPL 的第二个字母作为这种语言的名字，这就是 C 语言。C 语言非常精练，接近硬件，又克服了没有数据类型的缺点。

为了使 UNIX 操作系统推广，1977 年 Dennis M. Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本《可移植的 C 语言编译程序》。1978 年 Brian W. Kernighian 和 DennisM. Ritchie 出版了名著《The C Programming Language》，从而使 C 语言成为目前世界上流行最广泛的高级程序设计语言。

1988 年，随着微型计算机的日益普及，出现了许多 C 语言版本。由于没有统一的标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，美国国家标准研究所（ANSI）为 C 语言制定了一套 ANSI 标准，成为现行的 C 语言标准。

C 语言发展迅速，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，例如 PC-DOS、WORDSTAR 等就是用这种方法编写的。

1.3.2　语言的特点

一种语言之所以能存在和发展并具有生命力，总是有其不同于其他语言的特点。C 语言的主要特点如下：

（1）C 语言简洁、紧凑，使用方便、灵活。C 语言一共只有 32 个保留字、9 种控制语句，程序书写形式自由，主要用小写字母表示，压缩了一切不必要的成分，相对其他计算机语言而言源程序较短，因此输入程序时工作量少。