C CHENGXU SHEJI JIAOCHENG

C程序设计教程

李海华◎编著

电子科技大学出版社

C CHENGXU SHEJI JIAOCHENG

C程序设计教程

李海华◎编著

电子科技大学出版社

图书在版编目（CIP）数据

C 程序设计教程 / 李海华编著. —成都：电子科技大学出版社，2009. 10

ISBN 978-7-5647-0211-3

Ⅰ. C… Ⅱ. 李… Ⅲ. C 语言—程序设计—教材 Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2009）第 183497 号

内 容 简 介

本书介绍 C 语言的相关知识及程序设计基本技能，包括基础篇、实验篇。基础篇介绍了程序设计及 C 语言概述，数据类型、运算符与表达式，顺序结构，选择结构，循环结构，数组，函数，编译预处理，指针，结构体和共用体，位运算，文件等，每章的习题深化了读者对知识点的认识；实验篇给出了 16 个实验，以提高读者编写程序的实战能力。

本书编写的思路是：基础篇完成基本知识的学习，同时，实验篇完成基本技能的训练，体现了本书知识性、技能性、实用性的特点。

本书结构清晰，深入浅出，图文并茂，通俗易懂，可作为高等学校 C 程序设计学习的教程，也可作为各类计算机培训的教学用书以及各行各业需要进行程序设计的人员、爱好者的技术参考书。

C 程序设计教程

李海华 编著

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： 陈松明　袁　野
责任编辑： 陈松明　李述娜
主　　页： www.uestcp.com.cn
电子邮箱： uestcp@uestcp.com.cn
发　　行： 新华书店经销
印　　刷： 成都市火炬印务有限公司
成品尺寸： 185mm×260mm　　**印张** 17.25　　**字数** 430 千字
版　　次： 2009 年 10 月第一版
印　　次： 2009 年 10 月第一次印刷
书　　号： ISBN 978-7-5647-0211-3
定　　价： 29.50 元

前　言

C 程序设计教程是高校理工类专业学生必修的专业基础课，是“非计算机专业学生计算机应用能力水平考试”二级考试的主要语种之一。C 语言既具有高级语言的优点，又具有低级语言的许多特点。许多大的操作系统和编译系统都是用 C 语言来编写的，所以它是现今应用最为广泛的几种语言之一。在实际编程中，它有其他语言无法比拟的优势，是一种结构化程序设计的好语言。该课程是一门实践性很强的课程，既要掌握概念，又要动手编程，还要上机调试运行，可以培养学生丰富的表达能力和优良的编程风格，掌握几种基本的编程方法和基本算法。

➢ 主要内容

本书分为基础篇和实验篇。基础篇包括：C 语言的诞生与特点、算法与程序设计、算法的表示、结构化程序设计概要、C 语言程序的结构与函数结构；C 语言的数据类型、常量与变量、整型数据、实型数据、双精度型数据、字符型数据、运算符和表达式、逗号运算符和逗号表达式、系统函数；赋值语句、输入输出函数、注释、顺序结构、应用举例；关系运算和关系表达式、逻辑运算和逻辑表达式、if 语句、条件运算符和条件表达式、switch 语句；while 语句、do-while 循环语句（“直到”型循环）、for 循环语句、循环的嵌套、break 语句、continue 语句、goto 语句；一维数组的定义和引用、二维数组、字符数组；函数的分类、函数的概念与定义、函数的参数和函数的值、函数的调用、函数的嵌套调用及递归调用、数组作为函数参数、局部变量和全局变量、变量的存储类别、内部函数和外部函数；宏定义与符号常量、文件包含、条件编译；内存数据的指针与指针变量、指针变量的定义及指针运算、数组指针和指向数组的指针变量、字符串的指针指向字符串的针指变量、函数指针变量、指针型函数、指针数组和指向指针的指针；结构体类型的定义、结构体数组、指向结构体类型数据的指针、链表、共用体、枚举类型、类型定义符 typedef；数值在计算机中的表示、位运算符C语言提供的六种位运算符、位域（位段）；文件的基本概念、文件的基本操作、文件位置指针的有关函数。实验篇包括：Turbo C 2.0 简介和启动、Turbo C 2.0 集成开发环境、Turbo C 编辑窗口、VisualC++6.0 的基本操作等 16 个实验。

➢ 本书特色

1．实例丰富，例题新颖，针对全国大学生计算机等级考试，注重学生学习兴趣的培养。

2．知识性与技能性统一。基础篇完成基本知识的学习，实验篇完成基本技能的训练，体现了本书知识性与技能性相统一的特点。

3．以图析文，生动直观、寓教于乐。对于一些重要步骤、算法、数据结构等，都有图形示意，生动直观，易于理解和掌握。

4．基础篇中每章的习题深化了读者对知识点的认识，实验篇的 16 个实验可帮助读者提高编程能力。

本书结构清晰，深入浅出，图文并茂，通俗易懂。

由于作者水平有限，加之编写时间仓促，不足之处，希望使用本书的教师、学生及广大读者提出宝贵意见，以便进一步修订，不断提高教材编写水平。

作　者

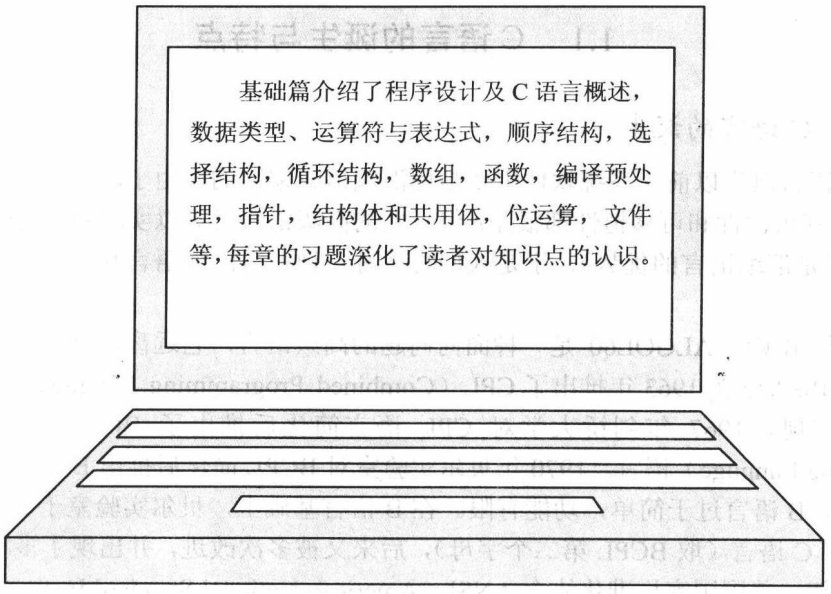
2009 年 7 月于郑州

目 录

第一篇 基 础 篇

第二篇 实 验 篇

第一篇 基 础 篇


基础篇介绍了程序设计及C语言概述，数据类型、运算符与表达式，顺序结构，选择结构，循环结构，数组，函数，编译预处理，指针，结构体和共用体，位运算，文件等，每章的习题深化了读者对知识点的认识。

第 1 章　程序设计及 C 语言概述

本章首先介绍算法和程序的概念以及程序设计的一般过程，然后介绍 C 语言的特点、C 语言程序的结构，最后介绍算法流程图和 N-S 盒图以及结构化程序设计的概念。学习本章的目的是使读者对 C 语言和程序设计有一个概略的了解。

1.1　C 语言的诞生与特点

1.1.1　C 语言的诞生

在 C 语言诞生以前，系统软件主要是用汇编语言编写的。由于汇编语言程序依赖于计算机硬件，其可读性和可移植性都很差；但一般的高级语言又难以实现对计算机硬件的直接操作（这正是汇编语言的优势），于是人们盼望有一种兼有汇编语言和高级语言特性的新语言。

1960 年出现的 ALGOL60 是一种面向问题的高级语言，它远离硬件，不易编程。在此基础上，剑桥大学在 1963 年推出了 CPL（Combined Programming Languge）语言，其规模大，不易实现。1967 年剑桥大学对 CPL 语言简化后推出了 BCPL（Basic Combined Programming Languge）语言。1970 年贝尔实验室对 BCPL 简化后推出 B 语言（取 BCPL 第一个字母），B 语言过于简单，功能有限。在 B 语言基础上，贝尔实验室于 20 世纪 70 年代初研制出来 C 语言（取 BCPL 第二个字母），后来又被多次改进，并出现了多种版本。20 世纪 80 年代初，美国国家标准化协会 ANSI（American National Standard Institute），根据 C 语言问世以来各种版本对 C 语言的发展和扩充，制定了 ANSI C 标准（1989 年再次做了修订），C 语言的发展过程如图 1-1 所示。

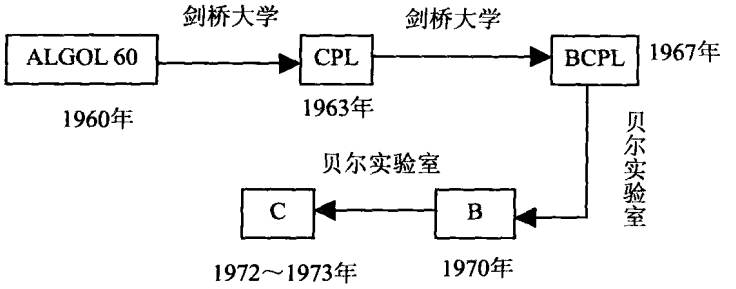


图 1-1　C 语言发展简图

1.1.2　C 语言的特点

C 语言是一种高级语言，兼有低级语言的功能，和其他高级语言相比，具有以下特点。

1．C 语言是一种模块化的程序设计语言。

2．结构化程序设计语言直接支持顺序、分支和循环三种典型的基本结构，是程序设计便于使用“自顶向下逐步求精”的结构化程序设计技术。

3．C 语言简洁、紧凑，使用方便、灵活，程序设计自由度大。

4．数据结构类型丰富。C 语言的数据类型有 13 种，特别是具有数据类型构造能力，它可以在基本类型（如字符型、整型、实型等）的基础上按层次构造各种构造类型（如数组、指针、结构体、共用体等），足以用来实现各种复杂的数据结构（如栈、链表、队列、树等）的运算。尤其是指针类型数据，功能非常强大。

5．具有结构化的控制语句。C 语言有 if else、while、do while、for、switch 等语句以适应结构化的程序设计，符合现代编程风格要求。

6．C 语言允许直接访问物理地址，能进行位（bit）操作，能实现汇编语言的大部分功能，可以直接对硬件进行操作。因此有人把它称为中级语言。

7．生成目标代码质量高，程序执行效率高。试验表明，针对同一个问题，用 C 语言描述的代码效率只比汇编语言低 10%～20%。

8．与汇编语言相比，用 C 语言写的程序可移植性好。所谓可移植是指程序从一个环境不加或稍加改动就可搬到另一个不同的环境上运行。统计资料表明，不同机器上的 C 语言编译程序 80%的代码是公共的，这就使得用 C 语言编写的程序，基本上不加修改就能用于各种型号的计算机和各种操作系统。

9．C 是 C++和 Java 的基础。但是，C 语言对程序员要求也高，程序员用 C 写程序会感到限制少、灵活性大、功能强，但在学习上较其他高级语言要困难一些。

1.2 算法与程序设计

1.2.1 算 法

1．引例

为解决某一特殊问题而采取的确定而有限的操作步骤，称为算法。请看下面几个例子。

【例 1-1】计算 1+2+3+…+100，可采取以下两种算法中的一种。

算法一：可以设两个变量（变量是指其值可以改变的量），一个变量代表和（s），一个变量代表加数（i），用循环算法表示如下：

第一步：0□s，1□i。

第二步：s+i⇒s。

第三步：i+1□i。

第四步：如果 i≤100，转第二步；否则转第五步。

第五步：输出结果 s，结束。

算法二：只有两步。

第一步：100×101/2□s。

第二步：输出 s，结束。

【例 1-2】判断一个大于等于 3 的正整数是不是素数。

所谓素数是指除了 1 和该数本身之外，不能被其他任何整数整除的数，例如 23 是素数，因为它不能被 2，3，4，…，21，22 整除。

判断素数的方法很简单，例如判断 n（n≥3）是不是素数，只需将 n 作为被除数，将 2～（n–1）各个整数轮流作除数，作除法运算，如果都不能被整除（余数不为 0），则 n 是素数。

算法表示如下：

第一步：输入 n 的值。

第二步：i 作除数，2□i。

第三步：n 除以 i，得余数 r。

第四步：如果 r=0，表示 n 能被 i 整除，则打印 n 不是素数，转第七步；否则执行第五步。

第五步：i+1□i。

第六步：如果 i≤n-1，返回第三步；否则打印 n 是素数，转第七步。

第七步：结束。

2．算法的属性

（1）有穷性

有穷性是指一个算法的操作步骤必须是有限的和合理的，即在合理的范围之内结束算法。例如求整数累加和的算法，由于整数本身是个无限集合，如果不限定其范围，会使求解步骤变成无限的。又例如，计算机执行某个算法需要几千年，虽然是有限的，但却是不合理的。当然，究竟什么算“合理”，并没有严格标准，由人们的常识和需要而定。

（2）确定性

算法中每个操作步骤都应当是明确的，而不应是含糊的、模棱两可的。在计算机算法中最忌讳的是歧义性，所谓“歧义性”是指可以被理解为两种或多种可能的含义。因为计算机至今还没有主动思维的能力，如果给定的条件不确定，计算机就无法执行。例如，“计算 3 月 1 日是一年中的第几天”，这个问题是不确定的，因为没有指明哪一年，不知道是不是闰年，闰年和平年中，2 月份的天数不一样，所以无法执行。

（3）有零个或多个输入

执行算法时需要从外界获得必要信息的操作称为输入。输入的数据个数根据算法确定。例如计算 1～100 累加和的算法不需要输入；计算 n! 的算法需要输入 n 的值；计算 m 和 n 的最大公约数和最小公倍数则需要输入 m 和 n 两个数的值。

（4）有一个或多个输出

执行算法得到的结果就是算法的输出，没有输出的算法是没有意义的。

最常见的输出形式是屏幕显示或打印机输出，但并非唯一的形式。执行算法的目的就是为了求解，“解”就是输出。

（5）有效性

算法中的每一个步骤都应当有效地执行，并得到确定的结果。例如当 b=0 时，a/b 是不能有效执行的。又例如，在 C 语言中，“a%b”中的 a 和 b 都必须是整型数据，否则也不能有效执行。

算法有优劣之分，一般希望用简单的和运算步骤少的算法。因此，为了有效地进行解题，不仅要保证算法正确，还要考虑算法的质量，选择合适的算法。

1.2.2 程序设计语言

用计算机语言描述的算法称为计算机程序，或简称程序。只有用计算机语言描述的算法才能在计算机上执行，这些语言就是程序设计语言，只有计算机程序才能在计算机上执行。人们在编写程序之前，为了直观或符合人类思维方式，常常先用其他方式描述算法，然后再

翻译成计算机程序。

程序设计语言分为如下三大类：

（1）机器语言。所有的指令都由二进制数字 0 或 1 编码组成。计算机硬件能直接执行的是机器语言程序。

（2）汇编语言。采用人们容易记忆的符号和标记来表示机器语言指令，使程序具有一定的可读性。汇编语言也称符号语言，用汇编语言编写的程序称汇编语言程序。计算机硬件不能识别和直接运行汇编语言程序，必须由“汇编程序”将其翻译成机器语言程序后才能识别和运行。

（3）高级语言。由人们容易理解的自然语言和数学语言中一些简单的符号和单词组成，语句功能强大、可读性好、编程效率最高。高级语言种类繁多（据统计有上千种），曾经引起广泛关注和使用的高级语言有 FORTRAN、BASIC、Pascal 和 C 等命令式语言（或称过程式语言）；有 LISP、PROLOG 等陈述式语言；还有当前流行的面向对象的程序设计语言，例如 C++、Java、Visual C++、Visual Basic、Delphi、PowerBuilder 等。高级语言程序也不能被计算机硬件直接识别和执行，必须把高级语言程序翻译成机器语言程序才能执行。语言处理程序就是完成这个翻译过程的，按照处理方式的不同，可以分为解释型程序和编译型程序两大类。C 语言采用编译程序，即把用 C 语言写的“源程序”编译成“目标程序”，再通过连接程序的连接，生成“可执行程序”才能运行。

1.2.3 程序设计的一般过程

简单的程序设计一般包含以下几个部分：

（1）确定数据结构。分析具体任务，确定输入数据和输出数据，确定数据的逻辑结构和存储结构。

（2）确定算法。根据确定的数据结构确定解决问题的方法，即完成任务的一步一步的步骤。

（3）编写程序。根据确定的数据结构和算法，使用选定的计算机语言编写程序代码。简称“编程”。

（4）调试程序。将编写好的程序输入计算机内存中，对程序进行测试并修正，直到程序符合任务要求。

（5）整理文档资料。根据数据结构和程序整理编写相关的文档资料。

1.3 算法的表示

常用来表示算法的方法有：自然语言、传统流程图、结构化流程图、N-S 盒图、伪代码、PAD 图等。下面介绍结构化流程图及 N-S 盒图。

1.3.1 流程图

流程图是用一组框图符号表示各种操作，也称框图。用流程图表示算法直观形象，易于理解。美国国家标准化协会 ANSI 规定的一些常用流程图符号，已为各国程序工作者普遍采用，如图 1-2 所示。

图 1-2　常用流程图符号

【例 1-3】计算 1+2+3+4+…+100 的流程图如图 1-3 所示。

【例 1-4】判断一个大于等于 3 的数是不是素数的流程图如图 1-4 所示。

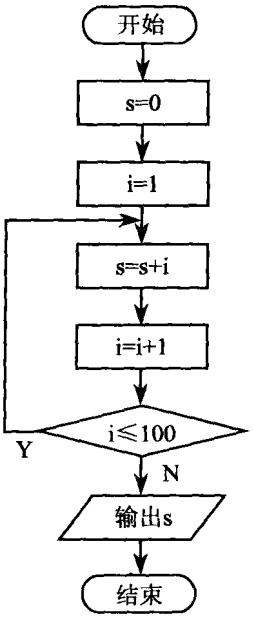


图 1-3　求 1～100 累加和的流程图

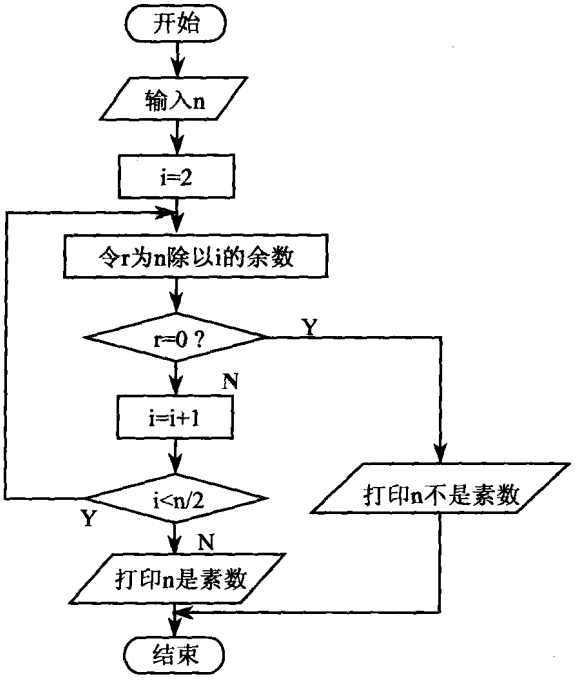
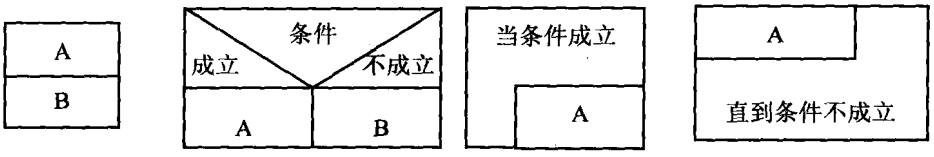


图 1-4　判断素数的流程图

1.3.2　用 N-S 盒图表示算法

N-S 图是美国学者 I.Nassi 和 B.Shneiderman 提出的一种新的流程图形式（N 和 S 是两位学者的英文姓名的首字母）。在 N-S 图中完全去掉了流程线，全部算法写在一个矩形框内，在该框内还可以包含其他的从属于它的框，即由一些基本框组成一个大框。

图 1-5 所示的符号为表示三种基本结构的 N-S 图符号。



（a）顺序结构　（b）选择结构　（c）当型循环结构　（d）直到型循环结构

图 1-5　三种基本结构的 N-S 图符号

除上述双分支选择结构以外，还有多分支的选择结构，如图 1-6 所示，当表示条件的值等于“值 i”时执行 Ai 框。虽然这种结构可以利用双分支的嵌套来实现，但 C 语言以及多数高级语言都提供了直接实现这种结构的语句。

前面的例 1-1 计算 1～100 的累加和、例 1-2 判断是否素数的算法分别如图 1-7、图 1-8 所示。