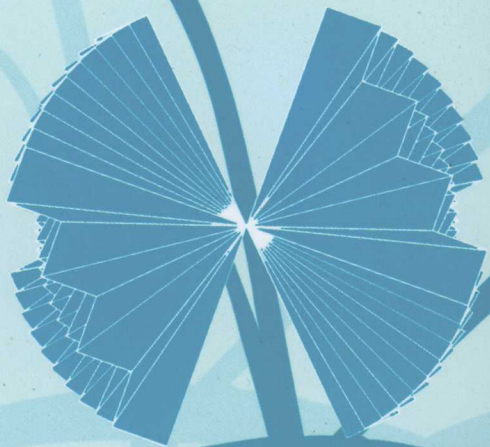
高等职业教育“十一五”规划教材

高职高专计算机专业基础课立体化教材系列

C语言程序设计

马晓晨 衡军山 ◎ 主编



科学出版社
www.sciencep.com

高等职业教育“十一五”规划教材

高职高专计算机专业基础课立体化教材系列

C语言程序设计

马晓晨　衡军山　主　编

郑阳平　苏建华　副主编

科学出版社

北　京

内 容 简 介

本书符合计算机等级考试教学大纲的要求，全书共12章，内容包括C语言基础知识、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、指针、编译预处理、位运算、结构体、共用体和用户自定义类型、文件。

本书除了作为高职高专计算机基础课程实验实训教材之外，还可以作为培训教材和自学参考书。

图书在版编目（CIP）数据

C语言程序设计/马晓晨，衡军山主编. —北京：科学出版社，2007
（高等职业教育“十一五”规划教材·高职高专计算机专业基础课立体化教材系列）
ISBN 978-7-03-020001-3

Ⅰ. C… Ⅱ. ①马…②衡 Ⅲ. C语言-程序设计-高等学校：技术学校-教材 Ⅳ. TP312

中国版本图书馆CIP数据核字（2007）第144617号

责任编辑：赖文华/责任校对：柏连海
责任印制：吕春珉/封面设计：耕者设计工作室

科学出版社出版
北京东黄城根北街16号
邮政编码：100717
http://www.sciencep.com

铭浩彩色印装有限公司印刷
科学出版社发行　各地新华书店经销
*
2007年9月第 一 版　开本：787×1092 1/16
2007年9月第一次印刷　印张：15
印数：1—3 000　字数：350 000

定价：21.00元

（如有印装质量问题，我社负责调换〈环伟〉）
销售部电话：010-62136131　编辑部电话：010-62138978-2003

前　言

“C 语言程序设计”是大部分计算机类专业的必修课程，也是很多非计算机专业开设的程序设计基础课程。作为计算机类的专业基础课，目的是使学生掌握程序设计的基本方法并逐步形成正确的程序设计思想，能够熟练地使用 C 语言进行程序设计并具备调试程序的能力，为后续课程及其他程序设计课程的学习和应用打下基础。对于非计算机专业来说，该课程有实际应用价值，为用计算机解决实际问题提供了方法，是后续理论和实践教学的基础和重要工具，同时也是计算机二级考试所要求的课程之一。

高等教育不断发展，作为重要基础课程的 C 语言程序设计也在不断更新和变化，突出实践动手能力的培养逐步成为人们的共识。作为一门计算机语言，对于初学者来说在思维方式上需要跨越心理上和思维方式上的障碍，需要逐步理解程序设计思想。如果象其他课程按照常规方法进行教学，随着课程的不断深入，大量的规则、定义、要求和机械的格式出现后，很容易使学生产生枯燥无味的感觉。而从本质上来说，计算机语言就是一个利用计算机去解决问题的工具，这就像我们在学如何使用螺丝刀时需要掌握的是如何去用螺丝刀拧螺丝，而不是学习螺丝刀的制作方法、研究螺丝刀的形状结构以及螺丝刀的制作材料等。我们培养学生的主要目的是使学生掌握 C 语言的使用方法，让学生真正具有利用 C 语言解决实际问题的能力，而不是让学生了解很多 C 语言的细节和原理，这一点对于高职高专学生尤其重要，培养学生自主学习和应用 C 语言解决实际问题比让学生掌握精通 C 语言语法重要得多。

因此，我们在编写本书时，“淡化语法，强调应用”是我们坚持的一个原则，但是考虑到本书的适用范围，很多教师多年来的传统教学思想和习惯不一定能够很快适应新的教学方法，在编写过程中充分分析了这一点，尽量适合不同风格的教学方法。所以本书在保持传统教材特点的基础上，注入了新的教学思想和方法，力争改变过去先给出定义和规则的讲授办法，从具体问题入手，努力把枯燥无味的语言讲得生动、活泼。让学生明白如何分析并解决实际问题，逐渐培养学生程序设计的正确思维模式。在每一章中都加入了“导学”，其目的除了为教师提示教学思想和方法之外，更重要的是，在学习新知识之前，引导学生认识学习的目的、学习的重点，并通过实例让他们对新知识的功能、方法和程序运行结果有一个感性认识（在阅读导学时，其中的实例只需感受不需理解和掌握），使学习目标更明确，学习更有针对性，更高效地掌握知识，培养解决问题的能力。“导学”可以很好地指导学生进行课前自学。

由于大部分学生是第一次接受计算机语言，教材注重“通俗性、可接受性”的原则。没有把重点放在语法规则的叙述上，而是放在程序设计方法上，通常由例题引出一种语法规则，通过一些求解具体问题的程序来分析算法，介绍程序设计的基本方法和技巧，既注重教材的系统性、科学性，又注重易读性和启发性。从最简单的问题入手，一开始就介绍程序，通过编写、运行程序来掌握语言的规定和程序设计的方法，然后再分析一些语法细节。在选择例题时也是由简到难逐步呈现给学生。

目　录

第 1 章　C 语言概述

导学

1. 问题的提出

C 语言是世界上最流行的程序设计语言之一。如果你想成为一名软件工程师，那么就应该学会用 C 语言来编写程序。

在计算机编程领域中，自从计算机高级语言 C 语言诞生后，它深受广大编程者的厚爱，人们把 C 语言称为程序员设计语言。第一个用 C 语言编写的 UNIX 操作系统，在世界范围内得到了广泛的应用，它的设计者因此于 1983 年获得了计算机科学的最高奖——图灵奖。正因为 C 语言是一种高效的程序设计语言，在当今应用很广泛，它既可作为系统描述语言编写系统软件，也可以编写应用软件。

2. 本章任务

1）C 语言的发展历史。

2）认识简单的 C 程序。

3）编辑、编译、连接、运行和调试一个 C 语言程序。

4）熟悉 Turbo C 2.0 集成环境，并使用其编写 C 语言程序的一般步骤。

1.1　C语言的发展历史

C 语言是国际上广泛流行的计算机高级语言之一。由于 C 语言的强大功能和各方面的优点，它既可作为系统描述语言编写系统软件，也可以编写应用软件。

C 语言编写的 UNIX 操作系统比早期汇编语言编写的 UNIX 操作系统，易于理解、修改和扩充，具有良好的移植性。作为优秀的操作系统，UNIX 在世界范围内得到广泛的应用。要使用 UNIX，就必须掌握 C 语言，因此，C 语言被越来越多的人熟知。认识到 C 语言具有的强大功能，是一门很有发展前途的计算机程序设计语言，渐渐地，C 语言成为程序员首选的标准开发语言之一。

C 语言是在 B 语言的基础上发展起来的。谈到 C 语言的历史发展，要追溯到 20 世纪 60 年代诞生的 ALGOL 语言，它是一门结构良好、逻辑严谨的面向问题的高级语言，但它距离硬件远，不易编写系统程序。1963 年英国剑桥大学推出了 CPL（Combined Programming Language）语言,它虽然在 ALGOL 语言基础上，更接近硬件，但是规模比较大，难以实现。1967 年英国剑桥大学对 CPL 语言进行了简化，推出了 BCPL（Basic Combined Programming Language）语言。1970 年美国贝尔实验室的 Ken Thompson 以 BCPL 为基础，设计出简单而且更接近硬件的 B 语言，并编写了第一个 UNIX 操作系

统。由于 B 语言过于简单，功能有限，1972 年至 1973 年间，贝尔实验室的 D·M·Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言既保持了 B 语言的精炼、接近硬件等优点，又克服了他们的缺点，这样，作为最初描述和实现 UNIX 操作系统的工作设计语言——C 语言诞生了。

1978 年贝尔实验室的 B.W.Kernighan 和 D.M.Ritchie（简称 K&R）合著了影响深远的《THE C PROGRAMMING LANGUAGE》一书，建立了 C 语言的 K&R 标准，被称为标准 C。由于 C 语言功能强大而灵活，世界各地的程序员都使用他来编写各种程序，然而，不同组织使用不同的 C 语言版本，不同程序实现之间微妙的差异，让程序员非常头痛。为了解决这种问题，美国国家标准化协会（American National Standards Institute）于 1983 年成立一个委员会，在 C 语言的 K&R 标准的基础上制定了一个 C 语言标准，通常称之为 ANSI C。现代 C 语言编译器基本都遵循该标准，本书也以 ANSI C 为标准。

20 世纪 80 年代中期，出现了面向对象程序设计的概念。贝尔实验室的 B.Stroustrup 博士，将面向对象的语言引入到 C 语言中，设计出 C++语言。随后在 2002 年微软公司推出了 C#语言，该语言与 C/C++语言有着密切的联系，并成为.NET 环境的重要编程语言。

在 C 语言得到较广泛应用的同时，C 语言的编译系统也很多。目前广泛流行的各种版本 C 语言编译系统有：Microsoft C、Turbo C、Borland C，Microsoft Visual C++等，本书使用的 C 编译系统是 Turbo C 2.0 集成开发环境。

1.2　C语言的特点

在当前的计算机编程领域中，有很多高级语言可供我们选择，如 C、Visual Basic、Java 等语言。虽然这些语言都非常卓越，适合完成大部分编程任务，但是基于以下几个原因，很多程序员认为 C 语言仍然是首选语言之一。

- C 语言具有现代化程序设计语言的特征。
- C 语言功能强大、灵活。使用 C 语言能够完成的工作只受限于您的想象力，语言本身不会带来任何约束。C 语言可以应用于操作系统、图形、电子表格、网络通信、实时控制、其他语言的编译器等领域，用途广泛。
- 数据类型和运算符非常丰富。体现在数据类型丰富，具有现代化程序设计的各种数据结构，运算符丰富，体现在 C 语言把括号、赋值号、逗号等都作为运算符处理。
- C 语言简洁、紧凑，具有结构化的控制语句，使程序模块化。虽然 C 语言中包含的关键字很少，只有 9 种控制语句，但在使用 C 语言编程时将发现，它能够完成任何任务。C 语言可以通过函数来编写 C 语言代码，在程序中或其他程序中再次调用这些函数。而且 C 语言提供了大量的库函数，供程序员使用，使程序模块化，提高编程的速度。
- C 语言允许直接访问物理地址，能够进行位（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。C 语言具有高级语言和低级语言的双重功能，所以也称 C 语言为中级语言。

- 生成目标代码质量高，程序执行效率高。由 C 语言生成的可执行代码占内存容量少，执行效率高。
- 与汇编语言相比，用 C 语言编写的程序具有良好的移植性。按照 ANSI C 标准编写的程序，无需修改，就可以移植到各种类型的计算机和不同的操作系统中。

基于以上特性，C 语言作为编程语言是初学者的首选。程序员用 C 语言编写程序会感到限制少、灵活性大、功能强。同其他高级语言相比，对操作系统、系统实用程序以及需要对硬件进行操作的场合，用 C 语言编写程序明显的优越于其他高级语言，有的大型应用软件也用 C 语言编写。

1.3　程序和程序设计语言

在开始学习程序设计时，初学者首先遇到的问题可能是：什么是程序？什么是程序设计语言？

1.3.1　程序和程序设计

程序一词来自生活，通常指完成某些事务的一种固定方式和过程。从表述方面看，对一系列动作的执行过程的描述，可以理解为程序。日常生活中也可以找到许多“程序”实例。例如，一个学生早上起床后到上学的行为可以描述为：

1）起床；
2）刷牙；
3）洗脸；
4）吃早点；
5）上学。

这是一个顺序程序，是最简单形式的程序。描述这种程序就是给出一个包含其中各个基本步骤的序列。如果按顺序实施这些步骤，就完成了该项事务。现实生活中有许多程序性活动，当我们身处其中时，通常需要按部就班地一步步完成一系列动作，对这种工作（事务、活动）过程的细节描述就是一个“程序”。这就是程序的一些直观特征。

那么，计算机如何来描述程序呢？为了使计算机能按照人们的意志进行工作，解决具体的问题，人们需要事先设计出解决问题的方案，并用计算机能够识别的语言将方案写出来，形成程序的过程称为程序设计。所谓的程序，就是一组计算机能够识别和执行的指令序列，每一条指令使计算机执行特定的操作。

实际上，计算机本身并不能解决任何问题，是人们事先进行程序设计，计算机再严格按照所设计的程序来执行各种操作，从而完成一项具体的任务。

1.3.2　程序设计语言

为了描述一个程序运行中的各种动作及其执行的顺序，就需要有某种适当的描述方式，一种描述方式就形成了一种语言。

语言通常指人们生活工作中使用的自然语言，如汉语、英语等。这些语言随着人类发展进步而自然形成，是人们互相交流信息的工具和媒介。人们用口头语言向别人传播

见闻、表达看法和想法；用书面语言写文章、书籍，以实现更大范围内的信息交流。在前面所描述的生活中的程序实例，就是用汉语作为描述程序的语言，所描述的程序是为了给人看，让人去做。

为了与计算机交流，指挥他工作，同样需要有与之交流的方式，需要一种意义清晰、人用起来比较方便、计算机也能够识别和处理的描述方式，即需要有描述程序的合适语言。可供人们编写程序用的语言就是程序设计语言，程序设计语言也常常被称为编程语言，书写程序必须要借助于程序设计语言。

程序设计语言是人与计算机交流的工具，是字、词、语法规则构成的指令系统。计算机本身可以识别并执行的，人们通过学习就可以掌握的语言。程序设计语言是人描述计算的工具，也是人与计算机交流的媒介，通过使用程序语言书写的程序，人们就能够指挥计算机完成各种特定的操作，解决现实生活中的各种各样的问题。C 语言是一种程序设计语言，也是一种高级程序设计语言。

1.4　简单的C程序介绍

为了说明C语言源程序的结构特点，先请读者看看下面两个程序。这两个程序由简到难，表现了C语言源程序在组成结构上的一些特点。虽然有关内容还未介绍，但可从这些例子中了解到一个C语言程序是什么样子，以及其基本构成和书写格式。

1.4.1　认识 C 语言程序

【例 1.1】 一个简单的 C 语言程序。

hello.c　　← 示例文件名

```
#include <stdio.h>
void main()
{
   printf("欢迎进入C语言世界!
\n");
}
```

← C 语言程序源代码

欢迎进入C语言世界!　　← 示例运行结果

用 C 语言书写的程序称为 C 语言源程序，简称 C 程序。【例 1.1】中 C 程序的文件名为 hello.c。C 程序文件名的后缀名是“.c”，C 程序是以文件名存储的。这个简单程序可以分为两部分：第一行是特殊行，说明程序用到了 C 语言系统提供的标准功能，有关内容参见标准库文件 stdio.h，作为第一部分；第二部分是程序的基本部分，从第二行到第五行，描述程序所完成的工作。该程序功能是在屏幕上输出“欢迎进入 C 语言世界!”。下面详细介绍：

第二行：　void main()

main 是主函数的函数名，表示这是一个主函数。程序在执行时，总是从 main()这个主函数开始执行的，void 表示主函数 main()没有返回值。

第三行：　{

第五行：　}

从第三行到第五行之间，属于 main() 函数的程序代码，这些代码用来完成某一种特定的操作。“{”表示程序由此开始，“}”表示程序到此结束。

第四行：　printf("欢迎进入 C 语言世界！\n");

描述程序所执行的工作，是本程序要求系统做的动作指令，即在屏幕上输出“欢迎进入 C 语言世界!”。在 C 语言中，每一条语句必须以分号(;)作为结束标志。 在这个程序中，利用缩排的方式，使得程序的层次分明，增加了可读性。其中，printf 函数的功能是把要输出的内容输出到屏幕上，printf 函数是一个由系统定义的标准函数，可在程序中直接调用。

【例 1.2】 另一个简单的 C 程序，求三个数之和。

示例文件名：sum.c

```
#include <stdio.h>
void main()                                    /* 主函数 */
{   int sum, a, b, c;                          /* 定义变量 */
    printf("Enter Two Numbers: ");
    c=100;
    scanf("%d%d",&a,&b);
    sum=a+b+c;
    printf("sum is %d\n",sum);
 }
```

程序运行的结果为：

```
Enter Two Numbers: 12  68
sum is 180
```

程序的功能是求三个数的和。/* …… */ 是表示注释部分，方便程序的阅读和理解，对程序的运行和编译不起作用。注释可以加在程序中任何位置。第 3 行是声明部分，定义变量 a、b、c 和 sum，指定 a，b，c 和 sum 为整型变量。第 4 行是标准输出函数 printf 函数，将“Enter Two Numbers:”输出在屏幕上，提示输入两个数。第 5 行是赋值语句，将整数 100 赋值给变量 c。第 6 行 scanf 是标准输入函数，表示从键盘上输入两个数。例如，输入 12 和 68，则表示将 12 和 68 分别赋值于 a 和 b，即 a 的值为 12，b 的值为 68。其中“%d”是输入输出语句的“格式说明符”，用来指定输入输出时数据的格式和类型，“%d”表示“十进制整数类型”。第 7 行计算 a，b，c 的和，然后赋值于变量 sum，那么现在变量 sum 的值就是 a+b+c 的值。第 8 行调用输出函数 printf 函数将 sum 的值输出到屏幕上。printf 函数括号中的“sum is”是普通字符串，“%d”是以十进制整数格式输出变量 sum 的值，最右端 sum 是要输出的变量，他的值为 180。

1.4.2　分析 C 语言程序的结构特点

由以上示例，可以看出 C 源程序的结构特点：

1）C 程序是由函数构成的，可以包含至少一个函数，即一个或多个函数。函数是 C 程序的基本单位，一个函数由两部分组成，即函数的首部和函数体。

函数的首部

```
{
    函数体;
}
```

函数体包括声明部分和执行部分。声明部分是对应用的变量的定义，【例 1.2】中的语句 int sum, a, b, c；就属于声明部分；执行部分是完成某一特定的操作或实现某一功能的一组代码，【例 1.2】中的第 4 行到第 8 行之间的语句。

2）一个 C 语言程序有且仅有一个 main 函数，即主函数。不论 main 函数在程序的什么位置，C 程序总是从主函数开始执行的。

3）C 程序的每一条语句必须以分号（;）作为结束标志，但主函数和花括号“}”之后不能加分号。

4）/* …… */表示注释说明，增强程序的可读性。C 语言的注释符是以“/*”开头并以“*/”结尾的串。在“/*”和“*/”之间的即为注释。程序编译时，不对注释作任何处理。注释可出现在程序中的任何位置。注释用来向用户提示或解释程序的含义。在调试程序中对暂不使用的语句也可以用注释符括起来，使程序编译和运行时跳过，该部分待调试结束后再去做相应的处理。

1.4.3　C 程序书写规则

为了书写清晰，便于阅读、理解和维护程序，在书写程序时应遵循以下规则：

1）C 程序没有行号，一般一条语句占一行。

2）利用缩排格式，表明 C 程序的层次关系，以便看起来更加清晰，增加程序的可读性。

3）C 程序习惯上使用小写字母书写，但在一些宏定义中，将常量名用大写字母表示。需要注意的是，C 语言区分大小写字母。

4）在程序中，适当的加上注释说明，方便程序的阅读和理解。

在编程时应力求遵循这些规则，以养成良好的编程风格。

1.4.4　理解 C 程序的编译和运行

C 语言是高级语言，用高级语言编写的程序称为“源程序”。而计算机只能识别和执行“0”和“1”组成的二进制指令。为了让计算机能够识别和执行高级语言编写的指令，必须使用一种“编译程序”的软件，把源程序“翻译”为由二进制组成的“目标程序”，然后将该“目标程序”与系统的库函数和其他“目标程序”连接起来，形成“可执行程序”。

编写好一个 C 语言源程序后，如何在计算机上运行呢？一般要通过以下四个步骤：

1）编写源程序，并保存。

2）编译源程序，生成目标程序。

3）连接各个目标代码、库函数，生成可执行程序。

4）运行可执行程序。

运行过程如图 1.1 所示。

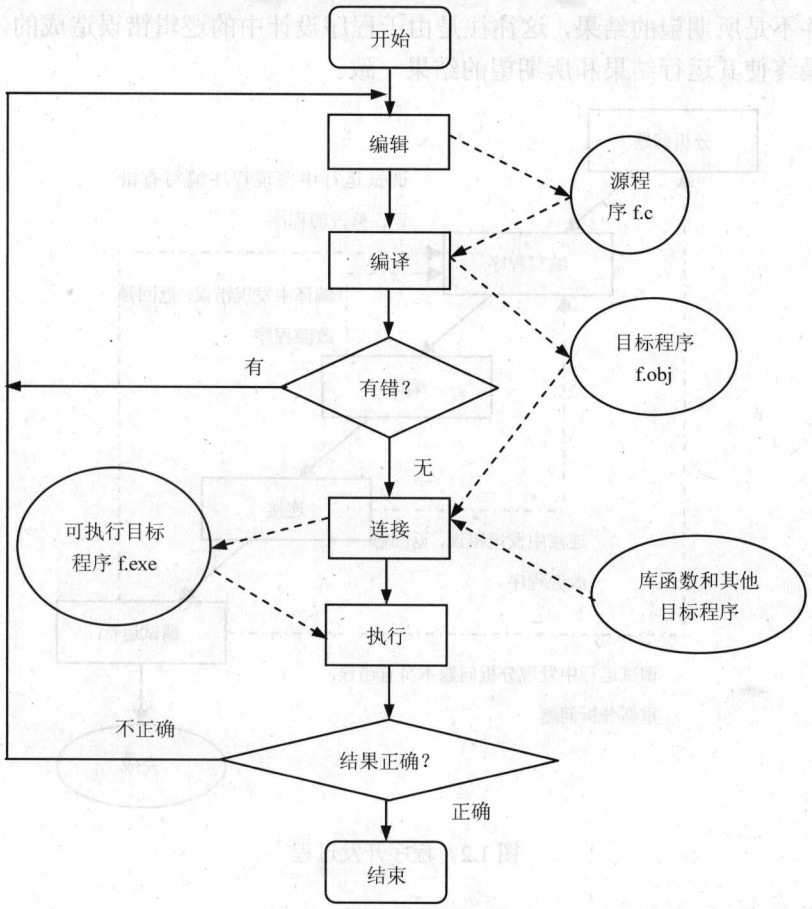


图 1.1　C 语言的上机步骤

其中实线表示操作流程，虚线表示文件的输入输出。例如编写的源程序文件，保存为 f.c，在进行编译时再将源程序 f.c 输入，经过编译后得到目标程序文件 f.obj，再将目标程序文件 f.obj 输入内存，与系统提供的库函数等连接，得到可执行的目标程序文件 f.exe，最后 f.exe 调入内存并使之运行，得到程序运行的结果。在编译、连接和运行时，如果产生了错误，则反复的排错与调试，直到没有错误，运行结果正确为止。

1.4.5　掌握开发程序的过程

为了让计算机能够解决具体的问题，可以按照下面的几个步骤去完成，如图 1.2 所示。

1）分析问题，设计出一种解决问题的途径与方法。

2）根据自己设想的解决方案，使用开发环境，编写程序，保存，如编写 C 程序。

3）使用编译环境对编写的程序进行编译，反复地修改程序中出现的错误，直到编译正确通过。

4）正常连接后生成了可执行程序，运行程序，查看结果。

5）调试，直到运行结果和实际问题一致。有时编译、连接完成后，程序能够运行，