“十三五”普通高等教育本科规划教材

C 语言程序设计

（第二版）

郑　玲　主　编
魏振华　石　敏　陈　菲　副主编

中国电力出版社
CHINA ELECTRIC POWER PRESS

“十三五”普通高等教育本科规划教材

C语言程序设计

（第二版）

主　编　郑　玲
副主编　魏振华　石　敏　陈　菲
参　编　姜力争　彭　文　胡海涛　葛　红

中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书为“十三五”普通高等教育本科规划教材，全书共 13 章，内容包括 C 语言程序设计初步、C 语言程序基础、选择结构、循环结构、数据类型和表达式、数组、函数、指针、结构体、共用体与枚举、指针的高级应用、文件、编译预处理。

书中全部例题均在 Visual C++ 6.0 环境下调试通过。并免费提供全部例题的程序源代码和教学用多媒体电子课件。

本书可作为高等学校本科、高职高专软件专业及相关专业程序设计的入门教材，也可作为全国计算机等级考试的辅导教材，也可供相关领域的工程技术人员学习参考。

图书在版编目（CIP）数据

C 语言程序设计/郑玲主编. —2 版. —北京：中国电力出版社，2018.8

“十三五”普通高等教育本科规划教材

ISBN 978-7-5198-1556-1

Ⅰ. ①C… Ⅱ. ①郑… Ⅲ. ①C 语言－程序设计－高等学校－教材 Ⅳ. ①TP312.8

中国版本图书馆 CIP 数据核字（2017）第 308778 号

出版发行：中国电力出版社
地　　址：北京市东城区北京站西街 19 号（邮政编码 100005）
网　　址：http://www.cepp.sgcc.com.cn
责任编辑：张　旻
责任校对：太兴华
装帧设计：张　娟
责任印制：吴　迪

印　　刷：北京天宇星印刷厂
版　　次：2009 年 9 月第一版　2018 年 8 月第二版
印　　次：2018 年 8 月北京第九次印刷
开　　本：787 毫米×1092 毫米　16 开本
印　　张：21.5
字　　数：517 千字
定　　价：55.00 元

前　言

“C 语言程序设计”是我国工科院校的一门必修课程，为了更好地体现高等学校培养人才的要求，围绕着高等技术应用型人才培养的主线，课程内容的改革本着突出基础理论知识的应用和实践能力培养的原则，按照突出应用性、实践性的原则重组课程结构，更新教学内容，为了适应教学要求，我们重新编写了新教材。该教材符合教学大纲要求，以培养学生程序设计能力为目标。通过该课程的学习，学生不仅要掌握高级程序设计语言的知识，更重要的是在实践中逐步掌握程序设计的思想和方法，培养问题求解和语言的应用能力。

C 语言是一门功能强大、灵活性好、可移植性高的程序设计语言，它具有自由的书写格式、良好的表达能力、丰富的数据结构、清晰的程序结构等优势。但由于 C 语言涉及的概念较多，语法规则比较繁杂且使用灵活，对于缺乏计算机基础知识的初学者来说，容易引起混乱。这也是造成 C 语言“难学”的主要原因之一。

虽然目前有关 C 语言程序设计的教材很多，但现有的教材一般围绕语言本身的体系展开内容，以讲解语言知识为主，特别注重语法知识讲解，书中大多数例题也是围绕的语法知识展开，很容易使学生陷入繁杂的语法记忆和理解中，对 C 语言的学习产生畏难情绪。

本教材是作者多年教学经验和应用 C 语言体会的结晶，在内容选择和结构组织上，体现以培养程序设计能力为核心，以 C 语言基础知识、算法基本概念和程序基本结构为重点的教学理念。本教材具有以下几个特点：

在结构组织上本着学以致用的原则，内容安排循序渐进，每个知识点的介绍都以引起读者的学习热情和兴趣为出发点，每一章都通过案例和问题引入内容，以解决问题为目的介绍相关的语言知识，重点讲解程序设计的思想和方法。为了避免过多地罗列 C 语言的语法规则使学生难以掌握，我们将难点分散到各个章节。教材从第 1 章开始就教学生学写简单的应用程序，在第 2 章我们只是简单介绍 C 语言的基本语法知识、简单的数据类型和输入/输出语句，利用这些知识学生就能实现简单的程序设计了，有关数据类型、表达式、数据类型转换等烦琐的运算规则我们放在第 5 章，这样既便于学生理解和掌握又不乏对知识的总结和提高。同样我们将数组、函数和指针也分解成两部分，即基础部分和提高部分，在第 6 章介绍了数组的基本概念和编程方法，在第 7 章函数中介绍了函数的基本概念，重点让学生掌握模块化的程序设计思想，在第 8 章指针中介绍了指针的基本概念，重点让学生掌握间接寻址的概念和方法。同时将数组和函数的概念进一步延伸，介绍如何利用指针解决函数设计中的问题，介绍了指针与数组、指针与字符串等典型的应用问题。在学习完结构体后，我们进一步地学习和理解指针，在第 11 章中给出了指针的高级应用。

在写作的风格上，注重教材的可读性和可用性。在每一章都以学习目标开始，让学生首先了解本章学习的关键内容；每一章的知识点都是由一个实例引入，以语言基础知识、算法基本概念和程序基本结构为重点，以应用为主线，引入了大量应用实例，侧重实例分析，在实例选取上，既考虑了实例的典型性，又考虑了实用性和趣味性，实例分析的重点也放在了程序设计的思想方法上，力求做到引人入胜，不断增加读者的编程兴趣。对于书中的每一个

例题，我们都给出了 Visual C++6.0 下的运行结果；在每一章的结尾我们都给出了小结，旨在对本章的内容做系统的概括和总结；为了便于学生学习和掌握，对于一些常用语法规则和常见错误提示我们都用简短精辟的语言进行了总结，并以醒目的方式（用边框括起来）提醒读者加强记忆。

第一版教材于 2009 年出版后，一直应用于大学本科教学。作者根据教学的应用情况，与 2017 年对原教材进行了改编和修订。增加和替换了部分实例。对原有的内容也进行了改编使之更能满足本科教学的要求。

在每一章的后面我们都给出习题，题型丰富，包括填空题、选择题、判断题和编程题。

我们还编写了与本书配套的《C 程序设计语言实验与习题指导》，提供了全部的习题解答和实验指导。

另外我们还提供了与本教材配套的多媒体教学课件、全部的例题和习题源代码都可以从 http://jc.cepp.sgcc.com.cn 上下载。

全书共分为 13 章，第 1 章和附录由葛红老师编写，第 2 章和第 3 章由郑玲老师编写，第 4 章和第 13 章由胡海涛老师编写，第 5 章和第 7 章由彭文老师编写，第 6 章由魏振华老师编写，第 8 章由姜力争老师编写，第 9 章、第 10 章和第 12 章由石敏老师编写，第 11 章由陈菲老师编写。全书由郑玲老师统稿，陈菲老师对全书进行了校对，李为老师对全书进行了审阅。

限于作者水平，书中难免存在不足之处，竭诚希望得到广大读者和同行的批评指正。

编　者

2018 年 5 月

目　录

第1章　C语言程序设计初步

学习目标

掌握程序设计语言的基本概念，理解机器语言、汇编语言、高级语言的区别及特点；
通过阅读简单的C程序，了解C语言的结构特点；
熟悉Visual C++6.0编程环境，掌握C语言程序在Visual C++6.0环境下的开发过程。

C语言是一种高级程序设计语言，它是由贝尔实验室在20世纪70年代开发出来的。它具有高效性、灵活性以及高可移植性等特点，经过多年的发展，它已经成为在许多领域具有广泛应用的、流行的编程语言。

在我们深入学习C语言之前，我们先学习计算机程序设计语言的概念、分类及特点，了解C语言的起源和发展。通过阅读简单的C程序，掌握C语言的特点及开发过程。通过本章的学习，读者可以对C语言有一个大概的了解。

1.1　C语言概述

什么是C语言？什么是程序？怎样设计程序？这往往是计算机语言的初学者首先遇到的问题。

1.1.1　程序设计语言

什么是计算机程序设计语言？我们知道人与人交流要用人们所理解的语言，人与计算机交互，让计算机按照人的命令完成指定的工作，就必须使用计算机所能理解的语言。因此，计算机程序设计语言是计算机能够理解和识别的、具有一定格式的语言，是人与计算机交互的媒介。

什么是计算机程序？要让计算机按照人的意志完成某项任务，就必须首先制定好完成该任务的执行方案，再将其分解成计算机所能识别的并可以执行的指令序列。将该指令序列存放在内存中，当人发出执行命令后，计算机自动地依次执行该指令序列，完成人所规定的任务。因此，计算机程序就是完成某一指定任务的一组有序的指令集合。

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

计算机是一个电子设备，它直接能读懂的语言是机器语言，是由“0”和“1”组成的二进制指令。使用机器语言编写程序是十分困难的，因为它难以记忆、容易出错，只有计算机专业人士才能掌握，而且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，必须重新编写程序。但是由于机器语言都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

因为机器语言难记、难读、容易出错，人们便用一些简洁的英文字母、符号串来替代一

个特定的二进制指令，比如，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。这种面向机器的符号语言就称为汇编语言，即第二代计算机语言。然而计算机是不认识这些符号的，这就需要一个专门的程序负责将这些符号翻译成二进制的机器语言，这种翻译程序被称为汇编程序。

汇编语言同样十分依赖于计算机硬件，移植性不好，但代码效率高。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，程序精炼而且质量高，所以至今仍是一种常用且强有力的软件开发工具。

3. 高级语言

因为机器语言和汇编语言都依赖于计算机硬件系统，具有难以掌握和可移植性差等特点，我们将它们统称为低级语言。

为了解决低级语言所面临的问题，人们意识到，应该设计一种这样的语言，这种语言接近于数学语言或人类的自然语言，同时又不依赖于计算机硬件，编写出的程序能在所有机器上运行，使得程序易读、易维护且编程效率高。这种语言就称为高级语言，即第三代计算机语言。从 1954 年第一个高级语言问世以来，高级语言发展很快，至今已达数百种，影响较大、使用较普遍的有 FORTRAN、BASIC、Pascal、C、C++、Visual C、Visual B、Delphi、Java 等。

用高级语言编写的程序称为源程序。源程序是不能在计算机中直接执行的，必须将其翻译成机器指令才能在计算机中执行。将源程序翻译成机器指令的方式有两种：编译方式和解释方式。

编译方式是通过编译程序将源程序全部翻译成机器语言程序（一般称为目标程序），然后通过连接程序将用户程序与系统提供的库函数连接在一起，形成可执行程序。可执行程序是可以在计算机中直接运行的程序。

解释方式是通过解释程序逐句翻译执行的，即翻译一条执行一条，不产生目标程序，每次执行都要重新翻译。

1.1.2 C 语言简介

C 语言是在 20 世纪 70 年代初问世的。1978 年，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 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》，也称为“K&R 标准”。但是，《K&R》中并没有定义一个完整的标准 C 语言。后来，美国国家标准学会在此基础上制定了一个 C 语言标准，于 1983 年发表，通常称之为 ANSI C，或者称为“标准 C”。

早期的 C 语言主要是用于 UNIX 系统。由于 C 语言的强大功能和各方面的优点逐渐为人们认识，到了 20 世纪 80 年代，C 语言开始进入其他操作系统，并很快在各类大、中、小和微型计算机上得到了广泛的使用，成为当代最优秀的程序设计语言之一。

目前，在微机上广泛使用的 C 语言编译系统有 Microsoft C、GCC、Turbo C、Borland C 等。虽然它们的基本部分都是相同的，但还是有一些差异，所以请大家注意自己所使用的 C 编译系统的特点和规定（参阅相应的手册）。

1.1.3 C++语言简介

在 C 语言的基础上，1983 年，贝尔实验室的 Bjarne Stroustrup 推出了 C++语言。C++语言进一步扩充和完善了 C 语言，成为一种面向对象的程序设计语言。C++语言目前流行的最新版本是 Borland C++、Symantec C++和 Microsoft Visual C++。

C++语言支持面向对象的程序设计方法，为程序员提供了一种与传统结构化程序设计不同的思维方式和编程方法，同时也增加了整个语言的复杂性，掌握起来有一定难度。

C 语言是 C++语言的基础，C++语言和 C 语言在很多方面是兼容的。因此，在学习 C++语言之前，最好先精通 C 语言，再进一步学习 C++语言，就能以一种熟悉的语法来学习面向对象的语言，从而达到事半功倍的目的。

1.1.4　C 语言的主要特点

C 语言之所以成为国际上广为流行的语言，是因为它有许多不同于其他语言的特点。其主要特点如下：

（1）C 语言是一种结构化语言，它层次清晰，便于按模块化方式组织程序，易于调试和维护。

（2）C 语言简洁、紧凑，使用方便、灵活，只有 32 个关键字和 9 种控制语句。

（3）C 语言的表现能力和处理能力极强。具有丰富的运算符和数据类型，便于实现各类复杂的数据结构。

（4）C 语言的库函数十分丰富，包含了数百个函数。这些函数可以用于输入/输出、数学计算、字符处理、存储分配等多种操作。

（5）C 语言可以直接对硬件进行操作，能实现汇编语言所能实现的大部分功能，还可以直接访问内存的物理地址，进行位运算。它集高级语言和低级语言的功能于一体，因此有人把它称为中级语言，既可用于系统软件的开发，也适合于应用软件的开发。

（6）C 语言生成的目标代码质量高，程序执行效率高。一般 C 语言生成的目标代码只比汇编语言低 10%～20%，是各类高级语言中最快的。

（7）C 语言的可移植性强。虽然 C 语言具有低级语言的功能，但与汇编语言相比，它不依赖于计算机硬件，在硬件结构不同的各种计算机之间不做修改或稍作修改即可实现程序的移植。

（8）C 语言语法限制不太严格，程序设计的自由度大。例如，对数组下标越界不做检查，对变量类型的使用比较灵活，整型和字符型数据在一定范围内可以通用等。大多数高级语言的编译程序对语法检查都比较严格，但 C 语言放宽了语法检查，允许程序员有较大的自由度，使程序员能编出高效灵活的程序。但同时也给初学者带来了难度，使得 C 语言难以掌握，编出来的程序容易出错。

1.2　C 语言程序简介

为了说明 C 语言源程序结构的特点，我们先学习两个简单的 C 语言程序，虽然有关内容还未介绍，但是我们可从这些例子中了解到组成一个 C 语言源程序的基本部分和书写格式。

1.2.1　简单的 C 语言程序

【例 1-1】 在屏幕上显示“This is a C program.”。

源程序：

```
/* 在屏幕上显示"This is a C program  */
#include<stdio.h>                    /*编译预处理命令*/
void main( )                         /*主函数*/
```

```
{
   printf("This is a C program.\n");    /*printf 输出函数*/
}
```

运行结果：

```
This is a C program.
Press any key to continue
```

程序说明：

（1）在程序的第 1、2、3、5 行都有用/*和*/括起来的内容，这是程序的注释部分。程序不执行注释部分，它用来说明程序的功能，帮助程序员阅读和理解程序。注释部分可以写在程序的任意地方，但是要注意，注释是不能嵌套的。

（2）#include 是编译预处理命令[1]，其意义是把尖括号< >（也可以使用双引号""）内指定的文件（如本例中的 stdio.h）包含到本程序来，成为本程序的一部分。被包含的文件通常是由系统提供的，其扩展名为.h，因此也称为头文件。C 语言的头文件中包括了各个标准库函数的函数原型[2]。因此，凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。本程序中调用了 printf( )库函数，该函数由 C 语言的标准输入/输出库提供，因此必须用 include 命令将 stdio.h 头文件包含到该程序中。“stdio”即“standard input and output”的缩写。此外，要注意的是，C 语句是以分号结束的，例如 printf( )函数语句后面要用分号做结尾，而预处理命令并不是 C 语句，所以不需要以分号做结尾。

（3）main( )是主函数，任何 C 语言程序都必须有且只有一个主函数，程序的运行都是从主函数开始的。主函数的函数体由一对大括号{}括起来，函数体可以由多个语句组成，这一对{}不能省略。main 前面的 void 是函数返回值类型[3]，void 表示该函数不返回任何值。

（4）printf( )是 C 语言提供的标准输入/输出函数库中的格式输出函数[4]，其功能是在屏幕上输出字符信息，输出的信息必须用双引号" "号括起来，其中的'\n'是转义字符[5]，代表换行符。

（5）输出屏幕的最后一行一般会给出类似“Press any key to continue”的提示，表示程序运行结束，按任意键将退出输出屏幕。不同的开发工具会给出不同的提示。

（6）C 语言是区分大小写的，main 和 Main 不是同一个标识符[6]，所以 Printf( )也不是由 C 语言提供的库函数，无法执行输出。

【例 1-2】 输入圆的半径，求圆的周长及面积。

源程序：

```
/*  程序的功能：求圆的周长与面积          */
#include<stdio.h>
#define  PI  3.1415926                  /*宏定义：定义 PI 为圆周率,PI 为符号常量*/
/* 主函数*/
void main( )
{    float r,circum,area;               /* r 圆的半径、circum 圆的周长、area 圆的面积*/
     float get_circum(float r);         /*  声明函数 get_ circum  */
```

[1] 见 13.3 文件包含
[2] 见附录 C
[3] 见 7.1 函数的概述
[4] 见 2.5 C 语言的格式输出 printf 函数
[5] 见 5.2.3 字符型
[6] 见 2.2.2 标识符

```
    float get_area(float r);          /*  声明函数 get_area */
    printf("请输入圆的半径:");          /*  在显示器上输出提示信息*/
    scanf("%f",&r);                   /* 从键盘上输入 r 存圆的半径*/
    circum=get_circum(r);             /* 调用函数 get_ circum 求圆的周长*/
    area=get_area(r);                 /* 调用函数 get_area 求圆的面积*/
    printf("圆的周长为%.2f,圆的面积为 %.2f\n",circum,area);   /*输出结果*/
}
/*  函数 get_ circum(r)计算圆的周长     */
float get_circum(float r)
{
    return(2*PI*r);
}
/*  函数 get_area(r)计算圆的面积        */
float get_area(float r)
{
    return(PI*r*r);
}
```

运行结果：

```
请输入圆的半径:1.5
圆的周长为 9.42,圆的面积为 7.07
Press any key to continue
```

程序说明：

（1）“#define　PI　3.1415926”为宏定义[1]，也是一个编译预处理命令，用来定义一个符号常量 PI。PI 代表圆周率 3.1415926。由于 π 在程序中不能作为标识符名[2]，所以我们用 PI 来表示。在宏定义之后，编译器在编译时会将所有的“PI”都替换成后面的字符串“3.1415926”，因此程序中的“`return(2*PI*r);`”实际上执行的是“`return(2*3.1415926*r);`”。要注意的是，这是直接的字符串替换，而并非给 PI 赋值，编译后，程序中是不存在 PI 的。因此，PI 并不是变量，由于它的值已经固定了，因此我们将其视为符号常量。而替换 PI 的“3.1415926”只是个字符串，而并非数值，原则上讲，它可以是任意字符串，只要替换后的程序不会产生编译错误，都是正确的。要注意的是，宏定义和“#include”命令一样，只是个编译预处理命令，而并非 C 语言语句，因此不需要以分号结束。如果宏定义的末尾写了分号，会视为替换字符串的一部分，会一同替换符号常量。所以，在宏定义末尾写分号，经常会引发编译错误。

（2）C 语言中的函数体一般分为两部分，一部分为声明部分，另一部分为执行部分。C 语言规定程序中所有用到的变量[3]都必须先声明后使用。如果程序中未使用任何变量，则无声明部分，如［例 1-1］。本例中使用了三个变量：r、circum 和 area，分别用来表示圆的半径、周长和面积。float 代表浮点类型，C 语言还有 int（整型）和 char（字符型）等数据类型。

（3）`float get_circum(float r);`和 `float get_area(float r);`是函数声明语句，也属于声明部分。函数 get_circum( )、get_area( )为用户自定义的子函数，这些子函数定义在主函数的后面，所以调用之前必须对它们进行声明。

[1] 见 13.2 宏定义
[2] 见 2.2.2 标识符
[3] 见 2.3 变量