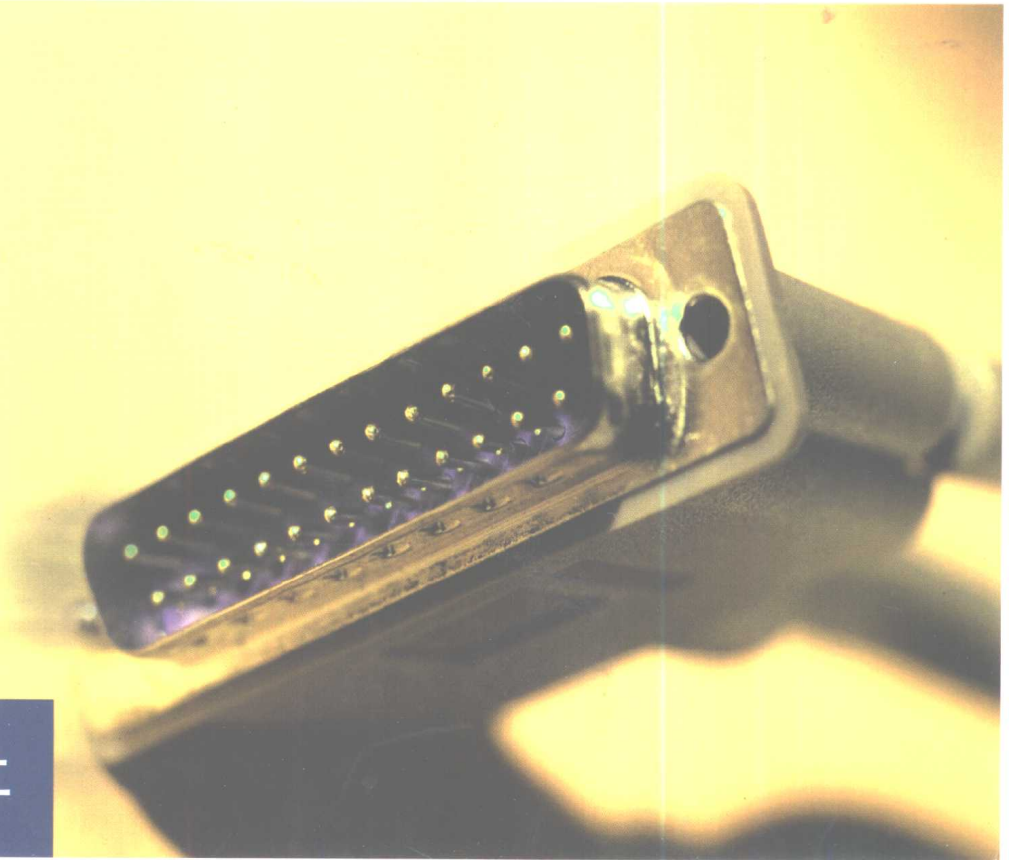
计.算.机.系.列.教.材

COMPUTER

C语言程序设计

王锐 王铠 王立宏 编著



人民邮电出版社

计算机系列教材

C语言程序设计

王锐 王铠 王立宏 编著

人民邮电出版社

内容提要

本书主要介绍 C 语言的程序设计方法。内容包括 C 语言基本概念、程序结构、数据类型、文件操作、编译预处理和上机实验等。在每一章后面附有习题可供读者练习使用。

通过本书的学习可以使读者了解 C 语言的最基本特征，掌握一定的程序设计方法，最终编写出简单的 C 语言程序。

本书可作为大学本、专科学生的学习教材，也可供计算机培训班和自学者使用。

计算机系列教材

C 语言程序设计

◆ 编　著　王　锐　王　铠　王立宏

◆ 人民邮电出版社出版发行　北京市崇文区夕照寺街 14 号

邮编　100061　　电子函件　315@ pptph.com.cn

网址　http://www.pptph.com.cn

北京顺义振华印刷厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092　1/16

印张:15

字数:368 千字　　1999 年 7 月第 1 版

印数:10 101－15 100 册　　2000 年 7 月北京第 2 次印刷

ISBN 7-115-07949-8/TP·1201

定价:19.80 元

编委会名单

序　言

为了适应“逐步实现教材多样化，增加不同品种、不同档次、不同风格、不同改革试验的教材”要求，我们组织编写了这套《计算机系列教材》，以适应大、中专计算机教学的需要。

本套教材的基本任务是系统地阐述计算机的基本概念和基本操作，这些基本概念和基本操作将是未来掌握计算机知识的基础。因此，本套教材的构架并不打算定位在不断变化的、十分活跃的研究和应用领域，而是立足于对基础知识的介绍上。本套教材共包括以下10本，计划1999年出版前5本，2000年出齐。

1. 《微型计算机应用基础》
2. 《中文 Windows 98 实用教程》
3. 《操作系统概论》
4. 《C 语言程序设计》
5. 《PASCAL 语言程序设计》
6. 《数据结构》
7. 《计算机实用软件》
8. 《计算机网络基础》
9. 《微机系统原理与维修》
10. 《汇编语言程序设计》

本套教材的书名基本上沿袭了其他版本，但无论在章节划分还是在内容选材方面几乎都是新的，因此完全可以说是一套新作。在写作方面，力求深入浅出、通俗易懂，并特别注重实例的选择和说明。为了加深对基本概念的掌握，各章的末尾均给出大量习题供学员课外练习。同时，每本教材的后面都附有实验题配合学员上机实习使用。

由于编写时间紧促，而且限于水平和经验的不足，书中肯定存在不少错误和遗漏，我们诚心希望使用本套教材的广大教师和同学们提出宝贵的批评建议。

教材编委会

1999年6月

前　言

软件界有这样一句话，“真正的程序员是用 C 的”。是的，C 语言有着其它语言无以伦比的优点，它的功能强大而灵活，几乎可以控制到计算机的任何角落，编写的程序具有良好的可读性和可移植性，并且最终生成的目标代码运行效率非常高。因此，C 语言常被用作大型系统软件的开发平台，比如，著名的 Unix 操作系统就是完全用 C 语言编写的。可以说，使用 C 语言进行程序设计已成为每个计算机程序员都应掌握的基本技能。

但是，人们通常认为 C 语言比较难，不容易被初学者掌握，尤其是非计算机专业的读者在学习时可能感觉更困难一些。因此本书在内容上作了精心安排，尽量做到循序渐进、重点突出并且通俗易懂。

因为计算机最终处理的都是数据，所以说，学习一门计算机语言的主要任务就在于了解它的两个功能：一是它如何来表示和描述数据；二是它如何来操作并处理数据。

C 语言采用数据类型来描述数据，通过各种程序结构来实现对数据的操作。因此，本书首先介绍整数、实数和字符等基本数据类型，然后以此为基础引出了 C 语言的程序结构，最后给出 C 语言的其它数据类型。

本书共 11 章，可分为 6 部分。第 1 章概要地介绍 C 语言的基本知识，为后面的学习作铺垫。第 2 章主要介绍 C 语言的数据类型、语句格式等。第 3、4 章介绍 C 语言的程序结构。第 5~9 章进一步介绍 C 语言的各种数据类型及程序设计方法与技巧。第 10 章讲述 C 语言特有的编译预处理功能。最后一章是实验，集中介绍了 C 语言的上机环境及各部分的上机习题。对于初学者易犯的错误，在这里也作了较为详细地讲解。

本教材列出了大量例题，对不同专业，可作适当取舍。

本书作者虽有多年的 C 语言教学和编程经验，但由于计算机技术在不断发展，加之作者水平有限，书中的错漏在所难免，敬请广大读者批评指正。

作　者

1999 年 5 月

目　　录

第1章　概论

计算机是 20 世纪最伟大的发明之一，它的诞生给人类的思想和行为带来了巨大的冲击。时至今日，计算机已经成为人们生活中不可缺少的一部分。比如，工厂在用它管理生产；银行在用它存款转帐；办公室在用它处理日常事务；家庭在用它学习和娱乐。尤其是近年来国际互联网（Internet）的迅速普及，使人们坐在家里就可遨游世界，与天涯海角的人自由交谈。计算机仿佛已无处不在、无所不能。

其实，计算机的这种能力是来自人的智慧，即人们为计算机设计的程序。

每个计算机内都运行着一系列由程序设计者编写的、能实现某种特定任务的程序。它使计算机可以根据人的意志而动作，从而服务于人。因此可以说，计算机若离开它内部运行的程序，只能是一堆无用的零件与电路。正是人们编写的程序，才给计算机赋予了“生命”。

为了能向计算机发出指令，使它按人的要求执行任务或做出反应，人们发明了程序设计语言。程序设计语言实际上是一系列对计算机可进行操作的规则，按这些规则人可以编写程序与计算机进行信息交流。因此，程序设计语言是人与计算机进行信息交流的工具。为了达到各种目的，人们设计了上千种程序设计语言，常用的也有几十种。众多的程序设计语言如何分类，目前众说纷纭。一般认为可把它们划分为 3 大类。

1. 面向机器的语言

最早出现的程序设计语言是机器语言。机器语言规定了若干由 0 和 1 组成的指令序列，可用它们来对计算机发布命令。比如，可用指令“10000000”表示加法运算；用指令“10010000”表示减法运算等。这些指令的优点是可以被计算机直接理解和执行，因此速度极快。但是，它们的缺点也很明显，即非常难记，也难以理解，更难以用它们来编写复杂程序。为了克服这些缺点，人们引进一些符号来表示这些指令。如将上面加法和减法运算的指令表示为“ADD”和“SUB”。

这些符号指令比单纯的 0 和 1 指令容易记忆，所以弥补了机器语言难记、难理解、难编写的缺陷。用它们来编写程序也使程序设计的效率和质量都得到了提高。人们把这种改进的、用符号来描述的指令系统称为汇编语言。

机器语言和汇编语言统称为面向机器的语言。它们的共同特点是针对特定计算机而设计，其指令序列依据不同的中央处理器（简称 CPU）而异，因此它们无法独立于机器而存在。也就是说，设计程序时不但要考虑到如何解决问题，而且要熟悉所用机器的内部结构和相应的指令才能具体实现程序。这样一来，在一台计算机上编制的程序，在另一台上可能根本无法运行，从而不能保证程序的通用性和可移植性，造成大量的重复劳动。

2. 面向过程的语言

面向过程语言的出现使程序在通用性和可移植性上得到了加强。它独立于机器，不因机

器的不同而不同，能用于各种计算机并解决各类问题。这使程序设计者不再需要了解计算机内部具体的结构，就可编写程序。因而相对面向机器的语言来说是一个进步。目前最流行的程序设计语言基本都是面向过程语言。如：Fortran 语言、Pascal 语言、C 语言和 Cobol 语言等。

使用面向过程的语言，用户要告诉计算机“做什么”，而且要分析解题的过程以便通知它“怎么做”，计算机才能做出相应的动作。因此这种语言也称为过程化语言或算法语言。

3. 面向问题的语言

使用过程化的语言解题，编程者的精力大多集中在“怎么做”这一解题过程上。而用面向问题的语言编程，人们却无须考虑这么多，只需要告诉计算机“做什么”，计算机便会替你完成具体解题过程。面向问题语言的代表有 SQL 语言、报表语言和判定语言等。

面向问题语言是程序设计语言发展的目标，它使用简单，程序编写方便，但是实现起来难度较大，运行效率及灵活性都不如面向过程的语言，并且因为当前大多数程序是用过程化语言编写的，因此，学习程序设计仍要以过程化的程序设计语言为基础。

C 语言是世界上最为流行的程序设计语言之一。它功能强大，在结构上具有模块化、结构化的特征，既可以用来设计系统软件，也可以用来编写应用软件，是一种典型的面向过程的语言。

本章主要介绍 C 语言的发展历史、特点、符号及程序设计、程序结构等问题。

1.1 C 语言概论

计算机技术从 20 世纪中期问世以来，在短短的几十年里，发展速度是惊人的。无论是软件和硬件，更新的周期都很短，每一次更新都有其时代特征，推动着计算机技术的不断进步。其中，C 语言的出现在计算机技术史上被一致认为是具有划时代意义的。

C 语言作为一种模块化、结构化、面向过程的通用程序设计语言，其特征在于紧紧把握了现代计算机系统结构的公共特征，能完成各种各样的编程任务，从而适用于从微型机、小型机到大型机等各类计算机。它的发展是一个不断演变、扩充和改进的漫长过程，凝聚了许多人的心血。

1.1.1 C 语言发展史

C 语言是 1972 年由美国贝尔实验室的 Dennis Ritchie 首先设计的。当时设计的目的是为了编写 UNIX 操作系统，而其根源可以追溯到 ALGOL60。

ALGOL60 是一种面向问题的程序设计语言，在 1960 年由国际委员会设计产生。因它与硬件联系比较少，不宜用来编写需要大量硬件支持的系统程序，于是，1963 年剑桥大学和伦敦大学在 ALGOL60 的基础上推出了复合程序设计语言 CPL（Combined Programming Language）。CPL 语言加强了与硬件的联系，但由于规模比较大、难以实现，因此，1967 年剑桥大学的 Matin Richards 对 CPL 语言进行了简化，研制了 BCPL（Basic Combined

Programming Language）语言。紧接着，为了编写 UNIX 操作系统，在 1970 年，贝尔实验室的 Ken Thompson 又对 BCPL 语言作了进一步简化，设计出了一种很简单而且很接近硬件的语言，借用 BCPL 的第一个字母“B”，命名为 B 语言。B 语言实现了历史上的第一个 UNIX 系统。但 B 语言又过于简单，因此功能有限。1972 年，贝尔实验室的 Dennis Ritchie 对它的进一步改进就导致了 C 语言的诞生（取 BCPL 语言的第二个字母“C”而得名）。Dennis Ritchie 对 B 语言从总体设计上进行了清理和改进，保留了 B 语言的优点，如：精练、接近机器等，并增强了语言的控制能力，丰富了数据类型，从而使 C 语言成为一种功能强大的通用程序设计语言。

Dennis Ritchie 和 K • Thompson 于 1973 年用 C 语言设计了 UNIX 操作系统第五版，进而扩大了 UNIX 的功能。但当时 C 语言还只被作为贝尔实验室内部使用的一种工作语言，并没有为人们所注意和了解。直到 1975 年，利用 C 语言编写的 UNIX 操作系统第六版公布后，C 语言才引起人们的广泛注意，并迅速风靡世界。

很显然，C 语言的发展和它在编写 UNIX 操作系统中的应用是紧密相关的，而 UNIX 操作系统能有今天的扩展面，C 语言也起了很大的作用。

C 语言的高效、实用和灵活，受到了人们的青睐，几乎各类计算机上都有 C 语言的影子。许多著名的应用软件，如微软公司的大部分产品都是用 C 语言写成的。C 语言已成为 IBM PC 系列微机上最主要的程序设计语言之一。不少软件生产厂家都有自己编写的 C 语言版本，这些版本在基本设计思想上都是一致的，但在具体的细节上却不尽相同。比较典型的有 Borland 公司出品的 C 语言编译系统和 Microsoft 公司出品的 C 语言编译系统。为了对 C 语言进行规范，在 1983 年和 1987 年，美国国家标准化协会（ANSI）根据 C 语言问世以来的各种版本，相继公布了两次 C 语言的标准——83 ANSI C 和 87 ANSI C。当前流行的 C 编译系统都是以它们为基础的。

1.1.2 C 语言的特点

C 语言在短短的几十年发展中能成为当今计算机界公认的一种优秀的程序设计语言，究竟是由于什么原因？又有什么特点呢？

- 正如前面所提到的，C 语言是一种结构化、模块化的程序设计语言。结构化或模块化程序设计的思想就是：按照系统论的观点，一个大型的复杂问题，可以分成若干个简单的小问题。C 语言将这些小问题用函数（或者叫作“模块”）的方式来逐一解决，把这些不同的函数或模块组装起来，就可以最终完成对复杂性问题的求解。正是基于这种思想，C 语言可以编制出结构简洁、思路清晰的程序。
- C 语言独立于计算机，对机器的依赖性很小，即具有良好的可移植性。C 语言最初是根据在 PDP-11 小型机上定义的小型语言来移植的，其代码很容易被改变到一个新的主机上。今天，无论是微机、工作站、大、中、小型机还是巨型机，都配有 C 语言编译系统，这是 C 语言能得到广泛应用的又一个主要原因。
- C 语言在程序设计上是一种优美的小型语言，它的关键字比典型的结构化程序

设计语言 PASCAL 少一些，而功能却要强大得多。因为它不但具有一般高级语言所具有的功能，而且还能够面向计算机硬件，具有汇编语言的某些功能。例如：它可以直接处理字符、位加工、地址及指针运算等，而这些通常需要汇编语言来完成。这一特征可以使 C 语言大大提高运算效率，从而使 C 语言程序有很高的运行效率。

- C 语言具有强大的数据控制功能，这包括数据处理能力和数据描述能力两个内容。数据处理能力是指 C 语言提供了大量的运算符（34 种），具有丰富的运算功能。数据描述能力是指 C 语言提供了多种数据类型，用以描述所处理的对象。
- C 语言具有预处理程序的功能，独树一帜，别具一格，是其它语言所没有的。
- C 语言的编程风格自由，语法限制少。
- C 语言是面向对象程序设计的基础。C 语言中几乎所有的功能和特点，都已被 C++所吸收，而 C++经过改进和补充，已发展为用以实现面向对象技术的程序设计语言的代表。所以我们掌握了 C 语言，就为今后的学习做好了准备。

1.1.3 C 语言的符号

要编写程序，必须使用各种符号将程序在计算机内表示出来。任何一种语言都有它自己规定的一系列语言符号，利用这些符号就可以定义语法，并组成不同的语句。

C 语言的基本符号有 4 类：

1. 字母。程序中可使用的英文大小写字母各有 26 个，共计 52 个。
2. 阿拉伯数字。程序中可使用的有 0，1，2，……，9，共 10 个。
3. 下划线符。
4. 特殊字符。程序中可使用的有：

(1) 算术运算符：加（+）、减（-）、乘（*）、除（/）、取模（%）。
(2) 自增自减运算符：自增加 1（++）、自减少 1（--）。
(3) 关系运算符号：小于（<）、小于等于（<=）、大于（>）、大于等于（>=）、不等于(!=)、全等比较（==）。
(4) 逻辑运算符：逻辑与（&&）、逻辑或（||）、逻辑非（!）。
(5) 位运算符：与（&）、或（|）、加（∧）、求反（~）、左移（<<）、右移（>>）。
(6) 赋值运算符：（=）。
(7) 组合运算符：加等（+=）、减等（-=）、乘等（*=）、除等（/=）、取模等（%=）、左移等（<<=）、右移等（>>=）、与等（&=）、或等(|=)、按位加等（∧=）。
(8) 逗号运算符：（,）。
(9) 条件运算符：（？）。
(10) 地址运算符：取地址（&）、取内容（*）。

(11) 其它运算符：对于不同的 C 语言版本，各有不同。如：数组下标定界（[]）、结构和联合体成员（.）、函数定义及引用（()）、结构指针（->）等。

1.1.4 C 语言的程序结构

下面通过几个最简单的例子来说明 C 语言的程序结构和最基本的组成部分，以使大家在学习 C 语言具体的语法规则前，对它有一个感性的了解。

【例 1-1】在屏幕上打印单词“Hello!”。

```
main()                  /*在屏幕上打印“Hello!”*/
{
   printf("Hello! ");
}
```

这是一个非常简单的 C 语言程序，它的主要功能就是在屏幕上打印一个英文单词“Hello”。当将这段程序输入计算机并编译执行后，屏幕上会出现如下结果：

```
Hello!
```

现在我们根据 C 语言的一些规定，来对【例 1-1】作一些解释。

1. 程序第一行中的“main（）”表示该程序是一个名称为 main 的函数。main 是 C 语言编译系统用的专有名字。每一个 C 语言程序都必须有一个以 main 作为开始的函数，被称为主函数，它标志程序该从这里开始执行。main 后面的一对圆括号‘()’，是函数的标志。C 语言用它来存放函数所使用的参数。通常，一个 C 语言函数可以有参数，也可无参数，但这对圆括号必须要有，不能省略。

2. “/*”与“*/”之间的内容是程序的注释部分，注释部分不参与程序的执行，对程序的运行结果没有影响，只是用来给编程人员作为一种提示或助记。注释的使用可以增强程序的可读性。

3. 程序从第二行的“{”符号到第四行的“}”之间的内容被称作函数体，也就是函数的可执行部分。‘{’和‘}’在这里的主要作用就是作为函数体开始和结束的标志，相当于 Pascal 中的“Begin”，“end”。

4. 【例 1-1】的函数体只有一条 printf 语句。printf()也是一个函数，是 C 语言的标准库函数。printf 的功能是在输出设备，如屏幕上输出圆括号中参数的内容。目前 printf 的参数为“Hello!”，也就是要在屏幕上打印“Hello!”这一字符串。

5. printf 语句最后是一个分号，这是 C 语言中每条语句的结束标志。它告诉编译系统当前语句到此结束，是必不可少的。

现在将【例 1-1】稍作修改，使它在屏幕上再打印单词“WORLD!”。

【例 1-2】在屏幕上打印“Hello!”和“WORLD!”。

```
main()                  /*在屏幕上打印“Hello!”和“WORLD!”*/
{
   printf("Hello! \n");  /*打印“Hello!”*/
```

```
    printf("WORLD! ");    /*打印“WORLD!“*/
}
```

程序编译运行后屏幕的输出结果为：

```
Hello!
WORLD!
```

仔细检查程序我们会发现程序中的函数体不但增加了一条 printf 语句，而且还在第一条 printf 语句的参数后增加了“\n”这个符号。\n 在 C 语言中是输出换行标志。它表示在出现在\n 以后的所有输出信息均要输出到下一行去。因此在【例 1-2】中“WORLD!”被显示到屏幕的下一行。

【例 1-3】求 3 个整型数 i，j，k 之和。

```
main ()                                          /*求三个整型数 i，j，k 之和*/
{
   int  i，j，k，sum;                             /*变量说明*/
   i=2; j=4; k=6;                                /*赋值并计算*/
   sum=i+j+k;                                    /*赋值并计算*/
   printf("The sum of i+j+k is%d\n", sum);  /*输出计算结果*/
}
```

在这个例题中，函数体由 6 条语句组成。

- int i，j，k，sum;

是说明语句，它说明 i、j、k、sum 四个变量为整型数。

- i=2; j=4; k = 6;

是 3 条（注意，是 3 条，而不是 1 条）赋值语句。其中“=”号为赋值运算符，表示将数值 2 赋给变量 i，4 赋给变量 j，6 赋给变量 k。这时，i、j、k 中的值分别就是 2、3、6。

- sum=i + j + k;

也是一条赋值语句，表示将 i、j、k 的值之和赋给 sum。这时 sum 的值为 12。

- 程序的最后一行是输出语句，它输出符号序列：The sum of i + j + k is，然后按 %d 指示符的规定（%d 规定输出十进制数）输出后面的变量 sum 的值。这个例子的输出结果是：

```
The sum of i+j+k is 12
```

为了进一步了解 C 语言的程序结构，下面介绍多函数的程序结构。【例 1-4】就是一个多函数的程序结构，main 函数中调用一个自定义函数 calculate。

【例 1-4】多函数程序结构例子。

```
main ()                                          /* 多于一个函数的结构*/
{
   calculate();                                  /*调用自定义函数 caluate()*/
}
calculate ()                                     /*自定义函数/*
{
```