教育科学“十五”国家规划课题研究成果

C 程序设计

王柏盛　　主　编
李万庆　贺洪江　副主编

高 等 教 育 出 版 社

内 容 提 要

本书全面介绍了 Turbo C 语言的基本概念，常量、变量、运算符和表达式，程序控制语句，函数，指针，结构、联合、枚举和定义类型，编译预处理命令，文件，字符屏幕和图形函数以及实用编程技术等内容。全书共分为 10 章。每章附有习题和实验，并精选了一部分全国计算机等级考试（二级 C 语言程序设计）的练习题，通过大量实例介绍 C 程序设计的思想、方法和技巧。

作者根据多年教学和科研积累的丰富经验，吸取当前一些 C 语言教材中的优点，增加了实用编程技术方面的内容，力求使本书体系合理、结构严谨、概念清晰、例题丰富、通俗易懂。

本书可作为高等院校程序设计课程的教材，也可供自学者使用或作为教师教学参考书。

与本书配套的辅助教材《C 程序设计习题题解》也同时出版。

前　言

C 语言是在国内外得到广泛使用的结构化程序设计语言，许多大型开发工具都以 C 语言作为基础语言。C 语言功能丰富、表达力强，使用方便灵活，目标程序效率高，可移植性好，既有结构化高级程序设计语言的优点，又有低级语言的绝大部分功能，它可以取代汇编语言，用来编写大型系统软件、工具软件和控制软件。

现在，C 语言已不仅为计算机专业工作者所使用，而且为广大工程技术人员所喜爱，许多人已经用它编写出很多优秀的应用软件。高等院校在非计算机专业普遍开设了 C 语言程序设计课程，大部分学生都选择 C 语言参加全国计算机等级考试。

本书全面介绍了 Turbo C 语言的基本概念，常量、变量、运算符和表达式，程序控制语句，函数，指针，结构、联合、枚举和定义类型，编译预处理命令，文件，字符屏幕和图形函数以及实用编程技术等内容。全书共 10 章，每章均附有习题和实验，并精选了一部分全国计算机等级考试（二级 C 语言程序设计）的练习题，通过大量实例介绍了 C 程序设计的思想、方法和技巧。

作者根据多年教学和科研积累的丰富经验，吸取当前一些 C 语言教材中的优点，大篇幅增加了字符屏幕、图形函数和实用编程技术方面的内容，力求使本书集教材、参考资料为一体，具有较强的实用性。本书的主要特点可以概括为以下几个方面：

（1）体系合理、结构严谨、概念清晰、例题丰富，在章节安排上，更符合结构化程序设计语言的特点。

（2）在程序设计技巧上有所创新，充分运用了 C 语言各种运算符和表达式的功能。各章设计了很多例程序并提供多种解法，以加强读者阅读能力、编程能力、调试能力和创新能力的培养。

（3）增加了字符屏幕、图形函数及实用编程技术等章节，使本书在深度和广度上都得到了加强。

（4）指出了 Turbo C 在表达式中存在的严重不一致性。

讲解本书约需 70 学时，其中上机实验 20 学时。不讲第九章和第十章需要 50 学时，其中包括 14 学时上机实验。有些章节，如第三章，有大量的例题，不少例题属于阅读、理解、加深题，学时不充分时可以跳过不讲。

由于程序设计语言是个有机的整体，各部分内容之间相互联系、渗透，因此，读者在使用本书时，应该在学习后面章节的同时经常复习前面的内容，以便加深理解。

与本书配套的辅助教材《C 程序设计题解》也同时出版。

本书第二、三、四、九、十章由王柏盛编写，第五、八章由李万庆编写，第一、六、七章和全部附录由贺洪江编写。全书由王柏盛统稿、整理。

本书在编写、出版过程中得到了河北工程大学有关领导的关心和支持，在此深致谢意。

由于作者的水平有限、经验不足，编写时间仓促，书中难免存在许多缺点不足之处，恳请广大读者和同行批评指正。

王柏盛

2003 年 12 月

目　录

第一章　C 语言概述

1.1　C 语言的起源

C 语言是 1972 年由美国人丹尼斯·里奇(Dennis Ritchie)设计发明的,并首次在 UNIX 操作系统的 DECPDP－11 计算机上使用。它是由早期的编程语言 BCPL(Basic Combined Programming Language)发展演变而来的。在 1970 年,AT&T 贝尔实验室的肯·汤普森(Ken Tompson)根据 BCPL 语言设计出较先进的语言,即 B 语言,在此基础上促进了 70 年代 C 语言的问世。

随着微型计算机的日益普及,C 语言出现了多种版本。为了改变这些版本不一致的情况,1983 年美国国家标准学会为 C 语言制定了一套 ANSI 标准。Turbo C 完全是按照 ANSI 的 C 语言标准实施的。它是一种快速、高效的编译程序。Turbo C 不仅提供一个集成开发环境,而且还按传统方式提供了一个命令行编译程序版本,以满足不同用户的需要。

1.2　C 语言的特点

1. 结构化语言

结构化语言的一个显著特点是代码和数据的分隔化,即程序的各部分除了必要的信息交流外彼此互不影响,相互隔离。C 语言的主要结构成分是函数,函数是 C 语言的基本结构模块,所有的程序活动内容都包含在其中,函数在程序中被定义完成独立的任务,独立地编译成目标代码,这样可以实现程序的模块化。C 语言中,另一个实现程序结构化和分离化的方法是使用复合语句,复合语句是作为一个语句对待的,且具有逻辑联系的程序语句的组合,它是一个逻辑单元。严格地说,C 语言并不是一种真正的结构化语言,因为它不允许在子程序或函数中再定义子程序或函数,但由于它的结构类似于 ALGOL、Pascal 和 Modula－2,通常还是把 C 语言称为结构化语言。因为,C 语言有现代程序设计语言的各种数据结构,它的数据类型有整型、实型、字符型、数组、结构、联合及指针和无值型,能用来实现各种复杂

的数据结构的运算,如队列、链表、树、图、堆栈等。另外,C 语言具有结构化的控制语句,如 if...else、while、do...while、for、switch 等语句。C 语言用函数作为程序模块,以实现程序的模块化,是结构化的编程语言,符合现代编程风格。

2. 简洁、紧凑、灵活

Turbo C 共有 43 个关键字,9 种控制语句,程序书写自由,主要以小写字母表示,压缩了一切不必要的成分。C 语言语法限制不太严格,程序设计自由度大,例如对数组边界不做检查,整型、字符型、逻辑型数据可以通用。

3. 运算符丰富

C 语言共有 44 种运算符,把括号、赋值、强制类型转换等都作为运算符处理,从而使 C 语言的运算类型极其丰富,表达式类型多样化,灵活使用各种运算符,可以实现在其他高级语言中难以实现的运算。

4. 中级语言

C 语言把高级语言的基本结构与低级语言的实用性结合起来。它可以像汇编语言一样对位、字节和地址进行操作,这三者是计算机最基本的工作单元,C 语言可实现汇编语言的绝大部分功能。

5. 移植性好

用 C 语言编写的程序移植性好,生成的目标代码质量高,程序执行速度快。

6. 功能强大

C 语言有丰富的库函数、强大的图形功能,有预处理能力,和其他语言如汇编语言、Pascal 语言、数据库语言等的接口容易实现,C 程序中还可以直接调用 DOS 命令。因此,C 语言适合于编写各种系统软件、工具软件等大型软件。目前,在工业计算机控制系统开发中,越来越多的人都使用 C 语言来编写控制软件。

7. 编译语言

C 语言是编译语言,用 C 语言写的源程序必须经过编译后才能运行。

1.3 C 语言的词法

1.3.1 字符集

C 语言的字符集包括:大小写英文字母、数字、下划线"_"及 +、-、*、/、=、,、;、?、!、''、%、.、&、^、#、'、<、>、|、(,)、{}、[,]、~、\等。

1.3.2 关键字

关键字又称为保留字。C语言编译系统对关键字赋有专门的含义。Turbo C共有43个关键字,分别包括以下几种类型:

(1) 描述存储属性的有:auto,extern,static,register;

(2) 描述数据类型的有:char,int,float,double,void,struct,union,enum,long,short,signed,unsigned;

(3) 描述语句的有:goto,if,else,switch,case,default,break,for,do,while,continue,return;

(4) 描述访问方式的有:const(常量修饰符),volatile(易变量修饰符);

(5) 描述编译状态的有:sizeof;

(6) 描述数据类型定义的有:typedef;

(7) Turbo C扩展的关键字有:asm,_cs,_ds,_es,_ss,cdecl,pascal,far,near,huge,interrupt。

1.3.3 标识符

C语句中以标识符命名程序中的对象名,如函数、变量、符号常量、数组、结构、联合、指针、数据类型、存储属性、标号及宏等。

标识符是以字母、数字和下划线组成的,但第一个字符必须是字母或下划线。Turbo C规定标识符的有效字符长度为1~32个字符。在C语言中字母大小写是有区别的,COUNT、Count和count为3个不同的标识符。习惯上,符号常量、宏名等用大写字母;变量、函数名等用小写字母;系统变量由下划线开头。

标识符不能和Turbo C中的关键字相同,虽然Turbo C并不排斥标识符和库函数中的函数名相同,但程序中应该尽量避免用库函数名做标识符。另外,main虽然不是ANSI标准的关键字,但各种C语言版本都把它作为主函数名,因此也不应该把它作为标识符。C源程序名不属于C语句,属于操作系统,其命名要求符合操作系统文件名的规定,其扩展名通常为.C。

标识符的选择应该做到"见名知义""常用取简""专用取繁"。例如:count、name、year、month、student_number、display、screen_format等,使之一目了然,以增加程序的可读性。

例:

正确的标识符名	不正确的标识符名	
test_123	100	数字打头

_label_100	interger !	包含 !字符
rectangle_area	chinese word	包含空格字符
circle_radius_r	for	是关键字
students	printf	虽然可用,但与库函数中的输出函数同名

1.4 C 程序的组成和结构特点

C 语言的 43 个关键字再加上语法规则,就构成了 C 编程语言。所有的 C 语言关键字都是小写字母。

所有的 C 语言程序都包含一个或多个函数。惟一不可缺少的是 main()函数,它是程序开始运行时第一个被调用的函数。一个好的 C 语言程序中,main()函数提纲性地列出程序所要做的事情,而这一提纲由一系列函数调用所组成。main 虽然不是 C 语言的组成部分,但还是应该把它作为关键字来对待,不要把它用作变量名,否则有可能使编译程序产生混乱。

1.4.1 程序举例

【例 1.1】

```
main()
  {
     printf("This is a C Program.\n");
  }
```

本程序的作用是输出以下一行信息:

This is a C Program.

其中 main 表示"主函数",每一个 C 程序都必须有一个 main 函数。函数体由一对大括号 {} 括起来。本例中主函数的函数体只包括一条输出语句,printf 是 C 语言中的输出函数,双引号中的字符串原样输出。"\n"是换行符,即在输出信息"This is a C Program."后回车换行。语句的最后有一个分号,分号是语句的组成部分。

【例 1.2】

```
main()                    /* 求两数之和 */
  {
     int a,b,sum;         /* 定义变量 */
     a = 123; b = 456;
     sum = a + b;
```

```
    printf("sum is %d \n" ,sum) ;
}
```

本程序的作用是求两个整数 a 和 b 之和 sum。/* ... */表示注释部分,注释只是为了方便阅读而加的,对编译和运行不起作用。第二行是变量定义部分,说明 a、b 和 sum 是整型变量(int)。第三行是两条赋值语句,使 a 和 b 的值分别是 123 和 456。第四行计算 a + b 的和并赋给 sum。第五行的'%d'表示按十进制整数格式输出 sum 的值。本例编译运行后输出的信息为

```
sum is 579
```

【例 1.3】

```
main()                              /* 主函数 */
  {
    int a,b,c ;                     /* 定义变量 */
    scanf("%d,%d" ,&a,&b) ;         /* 输入 a 和 b 的值 */
    c = max(a,b) ;                  /* 调用函数 max,将得到的值赋给 c */
    printf("max = %d" ,c) ;         /* 输出 c 的值 */
 }
int max(x,y)          /* 定义 max()函数,函数值为整型,x、y 为形式参数 */
int x,y ;                           /* 定义形参 x、y 为整型 */
 {
    int z ;                         /* 定义 max()函数中的 z 为整型变量 */
    if(x > y) z = x ;else z = y ;   /* 求 x、y 的最大值并赋给 z */
    return(z) ;                     /* 从子函数返回调用处,z 为返回的函数值 */
}
```

本程序包含两个函数:主调函数 main()和被调函数 max()。max()的作用是将 x 和 y 中的较大值赋给 z。return 语句将 z 的值返回给主调函数 main()。返回值是通过函数名 max()带回到 main()的调用处。main()函数中的 scanf 是'输入函数'的名字,其作用是通过键盘输入 a 和 b 的值。&a 和 &b 中的'&'的含义是'取变量的地址';"%d,%d"的含义按十进制整数形式输入。main()函数中的第三行为调用 max()函数,在调用时将实际参数 a 和 b 的值分别传送给 max()函数中的形式参数 x 和 y,执行函数 max 后,得到一个返回值赋给主函数中的变量 c。本程序编译运行结果为:

```
输入:8,5 < Enter >        输入 8 和 5 给 a 和 b, < Enter > 为回车键
输出:max = 8              输出 c 的值
```

本例中用到了函数调用、实参和形参等概念,将在函数一章中再详细介绍。

1.4.2 结构特点

通过上述几个简单的例子,可以看出以下几点:

(1) C 程序是由函数构成的。一个 C 程序中必须有一个名为 main 的主函数和若干个子函数(可以没有)。函数是 C 程序的基本单位。被调函数可以是系统提供的库函数,如 printf 和 scanf 函数,也可以是用户自己编写的函数,如例 1.3 中的 max 函数。C 函数相当于其他语言中的子程序。C 语言用函数来实现特定功能,可以说 C 语言是函数式的语言,程序的全部工作都是由函数来完成的。C 语言函数库十分丰富,Turbo C 提供三百多个库函数。这种特点使得 C 语言容易实现程序模块化。

当一个程序较大、包括很多函数时,可以分成若干个文件,每个文件包括几个函数。

(2) 一个函数由两部分组成:① 函数说明部分,包括函数名、函数类型、函数属性、函数参数和参数类型,函数名后必须跟一对圆括号,可以没有函数参数。② 函数体,即大括号内的部分,如果函数内有多对大括号,则最外层的一对大括号为函数体的范围。函数体一般包括变量定义和执行部分,也可以没有变量定义部分,甚至连执行部分也没有,这是一个空函数,它什么也不干。

(3) 一个 C 程序总是从 main 函数开始执行,与 main 函数在程序中的位置无关,即 main 函数可以在其他函数的前面或后面。

(4) C 程序书写格式自由,一行可以写一条语句,也可以写多条语句,一条语句也可以分成几行写,没有行号,也不像 FORTRAN 和 COBOL 那样严格规定书写格式。C 语言规定可以在任何一个分隔符和空格处换行。

(5) 每条语句以分号结束,分号是语句的组成部分,即使是最后一条语句分号也不可缺少。

(6) C 语言没有输入/输出语句,输入/输出是由库函数完成的,C 对输入/输出实行"函数化"。

(7) 可以用/* … */对程序中的任何部分做注释,以增加程序的可读性,注释对编译和运行不起作用。/* 和 */必须成对出现,且/与 * 、* 与/之间不允许有其他字符(如空格)出现。注释不能插在关键字或标识符的中间,C 语言规定有分隔符的地方都可以插入注释,或者说能换行的地方都可以插入注释。通常注释可以出现在程序段前面,以对该程序段加以说明,或在语句行的后面出现,以对该语句加以说明。

【注意】 语句 a = b/*p 的含义是变量 b 除以 *p 的商赋给变量 a,*p 是指针变量 p 指向的变量,这种写法会引起编译错误,因为 C 编译系统把/* 看成是注释的起点,找到 */才认为注释结束。修改的办法是将语句 a = b/*p 改为 a = b/(*p);。