国家信息产业部电子人才交流中心参与规划
“十一五”高职高专规划教材

C语言
程序设计教程
——基于Turbo C

科海策划
主编　李　莉　陈　哲　谢金达
副主编　严仲兴　张　捷　陈功文

- 知识讲解
- 要点回顾
- 课堂练习
- 上机实验
- 综合实例
- 网络资源

走实践应用案例教学之路·培养技能型紧缺人才

科学出版社

C 语言程序设计教程
——基于 Turbo C

李 莉 陈 哲 谢金达 主 编
严仲兴 张 捷 陈功文 副主编

科 学 出 版 社

内 容 提 要

本书循序渐进地讲解了C语言的基本语法和程序设计的基本方法。

全书共13章，分别介绍了C语言的集成环境、数据类型、运算符及表达式、输入输出语句、选择结构和循环结构、函数、数组、指针、编译预处理、结构体与共用体、枚举与位运算、文件等内容。每章除讲解知识点外，还穿插了较多的小程序，同时提供了要点回顾和习题。为加强学生的理解，最后给出一个综合程序设计实例，并在附录中给出了10个上机实验指导。

本书可作为大中专、高职高专学校的教材，也可以作为计算机等级考试的参考资料和自学用书。

图书在版编目（CIP）数据

C语言程序设计教程——基于Turbo C /李莉，陈哲，谢金达编著.
—北京：科学出版社，2007

ISBN 978-7-03-018537-2

I. C… II. ①李… ②陈… ③谢… III. C语言—程序设计—教材
IV. TP312

中国版本图书馆CIP数据核字（2007）第017249号

责任编辑：刘秀青 张 楠 / 责任校对：科 海
责任印刷：科 海 / 封面设计：林 陶

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

北京科普瑞印刷有限责任公司印刷
科学出版社发行 各地新华书店经销

*

2007年3月第一版 开本：16开
2007年3月第一次印刷 印张：19.5
印数：1-4000 字数：475千字

定价：29.00元

（如有印装质量问题，我社负责调换）

本书编委会

主　编： 李　莉　陈　哲　谢金达

副主编： 严仲兴　张　捷　陈功文

编　委： 安丰彩　石文华　高力勋

周爱霞　张瑞坤　宋玉璞

徐玉莲　彭　斌　刘冰洁

王　倩　张媛媛　冯焕婷

刘　燕　毕春华　李寒松

前　言

C 语言功能丰富、使用灵活、目标程序效率高、可移植性好，可用于开发系统软件和应用软件，是近年来被广泛应用的一种计算机语言。许多高校都开设了 C 语言课程，全国计算机等级考试也将其列为二级考试课程。作者在多年讲授 C 语言课程的基础上，结合全国计算机等级考试大纲，编写了本书，目的是使读者在学习程序设计语言的同时，培养自己的优良编程风格，掌握基本的编程方法和典型算法。

C 语言的数据类型多，运算规则复杂，语句形式灵活，初学者不易掌握。在本书编写过程中，力求做到语言简明、概念清晰、通俗易懂，同时将重点放在基本概念、基本方法上。讲解时，注意循序渐进，难点分散，对用到的基础知识都进行了简单介绍，对编程容易出错的地方给出提示，使读者学习时少走弯路。读者可以在没有太多计算机基础知识的情况下使用本书。

本书参照 ANSI 标准 C 编写，以 Turbo C 2.0 为程序运行环境，主要介绍 C 的数据类型、C 的结构化程序设计方法、函数间数据传递、C 的数组和指针的概念及应用、常用库函数及内存动态分配、结构体与共用体、位运算、文件的用法等。在内容安排上，注重知识的系统性和实用性，根据各章节的知识点，结合实用案例逐步给出新的编程方法并与原编程方法进行比较，并通过一些小型实例介绍典型算法，还给出每章的学习要点回顾和习题。为加强学生的理解，最后给出一个综合程序设计实例，并附有 10 个上机实验指导（根据课时安排，有些章节和例题可以留给学生自学）。

在本书出版过程中，得到科学出版社有关领导的大力支持和成洁编辑的热情帮助。在此表示衷心的感谢。

书中难免存在疏忽错误之处，诚挚地希望读者批评指正。

编　者

2007 年 1 月

目　录

第 1 章　C 语言基础知识

本章的学习目标：

了解 C 语言的发展及特点，掌握 C 语言程序的构成，了解 C 语言风格和源程序书写格式，了解 C 语言程序的编译和执行步骤，熟悉 Turbo C 集成环境，熟练操作独立完成一个 C 程序的输入、编译、连接、运行的操作过程。

1.1　C 语言的发展及特点

C 语言是目前最为流行的计算机高级程序设计语言之一。它设计精巧，功能齐全，不仅是开发系统软件的理想工具，也是开发应用软件的理想程序设计语言，因此深受广大程序设计者的欢迎。

C 语言是在 1972 年由美国贝尔实验室的 D. M. Ritchie 为描述和实现 UNIX 操作系统而设计的，现在已经成为一个成熟的通用编程语言，并广泛应用于多种机型（如个人计算机、工作站和大型机）和操作系统（如 Windows、DOS 和 UNIX）。C 语言既可以处理数据库、网络、图形、图像等，又适合在工业控制、自动检测等方面应用，比如现在控制和检测领域的很多软件都是用 C 语言编写的。

C 语言的特点可大致归纳如下：

（1）C 语言短小精悍，基本组成部分紧凑、简洁。C 语言一般只用 32 个标准关键字、45 个标准运算符以及 9 种控制语句，不但语言组成精练、简洁，而且使用方便、灵活。

（2）C 语言运算符丰富，表达能力强。C 语言具有“高级语言”和“低级语言”双重特点，其运算符包含的内容广泛，所生成的表达式简练、灵活，有利于提高编译效率和目标代码的质量。

（3）C 语言数据结构丰富，程序结构化好。C 语言提供了编写结构化程序所需要的各种数据结构和控制结构，并且具有以函数调用为主的程序设计风格，所以利用 C 语言所编写的程序具有良好的结构。

（4）C 语言提供了某些接近汇编语言的功能，有利于编写系统软件。C 语言提供的一些运算和操作，能够实现汇编语言的功能，比如它可以直接访问物理地址，并能进行二进制位运算等，这为编写系统软件提供了方便。

（5）C 语言程序所生成的目标代码质量高。C 语言程序所生成目标代码的效率仅比用汇编语言描述同一个问题低 20%左右，因此用 C 语言编写的程序执行效率高。

（6）C 语言程序的可移植性好。在 C 语言所提供的语句中，没有直接依赖于硬件的语句，与硬件有关的操作，如数据输入、输出等都是通过调用系统提供的库函数来实现的。因此，用 C 语言编写的程序能够很容易地从一种计算机环境移植到另一种计算机环境中。

当然，C 语言本身也有弱点：一是运算符的优先级较多，不容易记忆；二是由于 C 语言语法限制不太严格，在增强了程序设计灵活性的同时，一定程度上也降低了某些安全性，这对程序设计人员提出了更高的要求。

有些 C 语言的不足之处，在 C++里已经得到了改进。1983 年，贝尔实验室的 Bjarne Strou-strup 在 C 的基础上，推出了 C++。C++进一步扩充和完善了 C 语言，目前流行的版本有 Borland C++和 Microsoft Visual C++。C++语言支持面向对象技术，易于将问题空间直接映射到程序空间，为程序员提供了一种新的思维方式和编程方法。

C 是 C++的基础，C++语言和 C 语言在很多方面是兼容的；在 C++的环境中，许多 C 代码可以直接使用。因此，掌握了 C 语言后再学习 C++，就能以一种熟悉的语法来学习面向对象的语言，从而收到事半功倍的效果。另外，广泛应用于 Internet 的 Java 语言具有安全、跨平台、面向对象、简单、适用于网络等显著特点，它也采用了 C 语言中的大部分语法，所以学好 C 语言还有助于学习 Java 语言。

1.2 C 程序的构成

C 语言是函数式语言，一个 C 程序由一个或多个函数组成，其中必须有且只能有一个名为 main 的主函数。在程序中，主函数 main()的位置没有特殊要求，但程序执行是从主函数开始，随主函数的结束而结束。主函数是整个程序的控制部分。在主函数执行过程中，可以调用 C 语言库函数或用户自定义函数。C 语言有丰富的库函数，可以供用户直接调用。

例 1.1 C 程序基本结构举例。

```
#include <stdio.h>
main()
{
   printf("This is an example.");
}
```

本程序运行时输出一个字符串“This is an example.”。

本程序有一个主函数 main()，花括号{}中的内容是 main 的函数体。一般函数体由若干条语句组成，包括说明语句和执行语句。说明语句用来定义或声明程序中出现的变量名称和类型，执行语句进行计算和输出。

程序中的第一条语句是文件包含语句，其作用是将标准输入输出头文件包含到当前文件中来。头文件又称为标题文件或包含文件，扩展名一般为“.h”。头文件可以看作源文件之间的接口，程序设计时需要用文件包含命令 include 将头文件包含到程序文件中来。头文件内容一般为：类型声明、函数声明、常量定义、数据声明、枚举定义、包含指令、宏定义、注释等。

上例文件中要用到输出函数 printf，而该函数原型在标准输入输出头文件 stdio.h 中定义，所以程序开始要添加#include <stdio.h>语句，否则在编译时候会出现 printf 没有定义的错误信息。

程序的开始执行点是 main()。如果程序有多个函数，且在 main 函数的执行过程中调用其他函数，被调函数执行结束后会返回到主函数中对应的调用点；主函数执行到 return 语

句或右花括号“}”时，程序结束。

C 语言程序是用函数驱动的，函数要先定义或声明后才可使用。有关函数的更详细内容将在后续章节中学习，先通过下面例题了解 C 程序的一般组成。

例 1.2 两个函数组成的程序，功能是计算这两个整数的乘积。

```
#include <stdio.h>                          /* 文件包含 */
main()                                      /* 主函数名 */
{                                           /* 函数体开始 */
   int product(int ,int );                  /* 函数声明*/
   int x,y,z;                               /* 局部变量类型定义 */
   printf("enter the value of x,y:");       /* 屏幕提示信息 */
   scanf("%d%d",&x,&y);                     /* 输入变量值 */
   z=product(x,y);                          /* 调用计算乘积的函数 */
   printf("x=%d,y=%d\n",x,y);               /* 打印 x 和 y 的值 */
   printf("x*y=%d\n",z);                    /* 打印乘积值 */
}                                           /* 函数体结束 */
int product(int a,int b)                    /* 被调函数及其形式参数 */
{
   int c;                                   /* 定义局部变量 */
   c=a*b;                                   /* 计算乘积 */
   return(c);                               /* 返回值 */
}
```

这是一个简单的含多个函数的 C 程序，它含有一个主函数 main()和一个子函数 product()。主函数负责输入两个变量的值，然后调用子函数，最后打印结果。子函数负责接收从主函数传送过来的两个数，并计算其乘积，最后将该乘积返回主函数。

程序运行时，屏幕首先显示提示信息：enter the value of x,y:_。

提示信息末尾的“_”是闪烁的光标，用户若从键盘输入 5 和 8（5 和 8 用空格隔开），按回车键后，程序输出为：

```
x=5,y=8
x*y=40
```

1.3 C 语言风格和源程序书写格式

C 程序风格灵活，书写格式自由，一般有以下特点：

（1）如果程序源文件由多个函数组成，每个函数在整个程序文件中的位置可以任意。但不管主函数位于程序中的何处，程序总是从主函数开始运行。

（2）标识符的命名应该“见名知意”，例如：sum，average，student_no；如果用多个单词表示一个标识符，中间可以用下划线分隔。标识符一般采用小写字母，大写字母常用作符号常数。

（3）每个语句以分号结束。既允许一行内写多条语句，也允许一条语句分多行书写。但为了看起来清楚，建议每行写一个语句。

（4）分级缩进格式。同一级别的语句采用同样的缩进格式，使程序看起来条理清晰，

有层次感，语句的执行顺序与书写时的缩进格式无关。

（5）适当使用注释，帮助理解程序。注释写在“/*　　*/”内，可以出现在程序中的任何位置，对源程序编译时将跳过注释语句，也就是注释语句对目标程序和可执行程序都没有任何影响。

例 1.3　程序书写格式举例。

下面程序可实现计算 $S=1+\frac{1}{2!}+\frac{1}{3!}+...+\frac{1}{100!}$（暂时不讨论程序的设计思路，只是了解程序书写格式）。

```
#include <stdio.h>
main()
{
    double sum=0.0,t,s;              /*定义变量类型和名称 */
    int i,j;
    for(i=1;i<=100;i++)              /*外重循环开始 */
    {
        t=1;
        for(j=1;j<=i;j++)            /*内重循环，计算阶乘 */
        t=t*j;
        printf("%e\n",t);            /*输出 t 的值 */
        s=1/t;                       /*计算阶乘的倒数 */
        sum=sum+s;                   /*求阶乘倒数的和 */
    }                                /*外重循环结束 */
    printf("\n%e\n",sum);            /*输出结果 */
}
```

该程序是一个双重循环，每重循环体内的语句均采用分级缩进格式书写。程序的缩进格式仅使程序看起来有层次感，对程序的逻辑结构和编译执行没有影响。

1.4　C 程序的编译和执行

C 语言是一种编译型的程序设计语言。一个 C 程序要经过编辑、编译、连接和运行 4 个步骤，才能得到运行结果。

1. 编辑

所谓编辑是指对 C 语言源程序进行输入和修改。编辑所使用的软件是各种编辑程序，如 DOS 系统提供的全屏幕编辑程序 edit，Windows 系统提供的记事本程序 notepad 等，都可以用来编辑 C 源程序。编辑结束后，将 C 源程序以文本文件的形式存放在磁盘上，文件名由用户自己选定，扩展名一般为“.c”，例如 f1.c，example.c 等。C 语言源程序是以 ASCII 代码形式输入和存储的，不能直接被计算机执行。

2. 编译

编译是把已编辑好的 C 源程序翻译成可重定位的二进制目标程序，编译过程由编译程

序完成。在编译时，编译程序自动对源程序进行句法和语法检查，当发现错误时，就将错误的类型和在程序中的位置显示出来，以帮助用户修改源程序中的错误。此时应重新进入编辑状态，对源程序进行修改后再重新编译，直到通过编译为止。编译完成后，编译程序自动生成二进制目标代码并对目标代码进行优化后生成目标文件。目标文件的名称由编译系统自动规定为和源文件同名，也可以由用户另外指定；目标文件的扩展名为“.obj”。

3. 连接

经编译后得到的二进制代码还不能直接执行，必须把经过编译的各个模块的目标代码与系统提供的标准模块（如 C 语言中的标准库函数）连接后才能运行。

连接也称链接或装配，是用连接程序将编译过的目标程序和程序中用到的库函数连接装配在一起，得到具有绝对地址的可执行文件，即计算机能直接执行的文件。此文件的扩展名由系统自动指定为“.exe”（例如 f1.exe）。

4. 运行

运行是将可执行文件投入运行，以获取程序的运行结果。运行可执行文件时，在 DOS 操作系统下是在提示符后直接键入可执行文件名，在 Windows 操作系统下是用鼠标双击可执行文件名。

C 程序的编译和执行过程如图 1-1 所示。

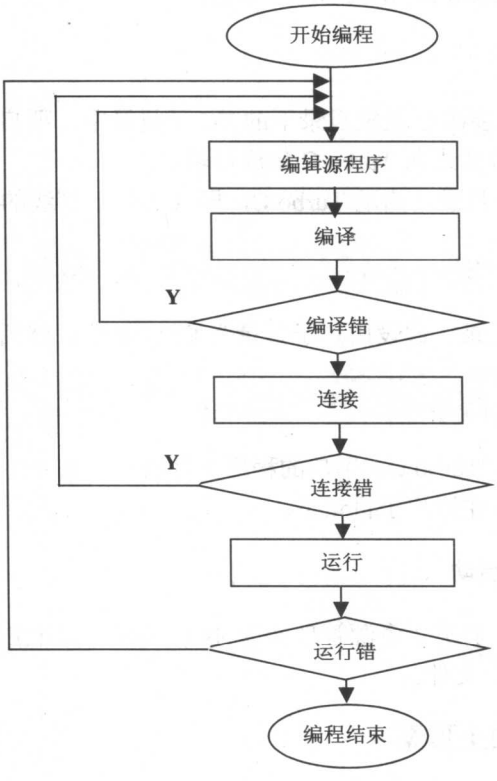

图 1-1　C 程序的编译和执行过程