21 世纪计算机应用基础课程教材

C 语言程序设计教程

祝胜林　主编

孙爱东　张明武　王　琴　参编

严尚维　主审

华南理工大学出版社

·广州·

内 容 简 介

C 语言是一种编译型程序设计语言，也是应用最为广泛的计算机语言之一，是当代大学生从事计算机应用应该熟练掌握的一种程序设计工具，同时又是学习面向对象 C ++ 、JAVA 等语言的基础。

本书以 ISO C(ISO 9899—1990)为基础，结合当前个人计算机流行的 C 编译系统，尤其是 Turbo C 2.0的编译环境，理论与实践相结合，全面介绍了 C 语言及其结构化程序设计的概念和技术。该书结构严谨，内容编排紧凑且具新颖性、独创性。书中给出了大量的程序举例及详细分析，每章罗列了相关的学习目标、重点与难点、学习提示并配有相当数量的习题，帮助读者总结、提高。

本书可作为大专院校、成人教育、函授和计算机应用培训班的教材，供从事计算机软件开发的程序设计人员和计算机应用的科技人员自学和参考，也可作为参加计算机等级或水平考试的参考书。

图书在版编目(CIP)数据

C 语言程序设计教程/祝胜林主编. —广州：华南理工大学出版社，2004.8
(21 世纪计算机应用基础课程教材)
ISBN 7-5623-2105-1

Ⅰ.C… Ⅱ.祝… Ⅲ.C 语言-程序设计-高等学校-教材 Ⅳ.TP312

中国版本图书馆 CIP 数据核字(2004)第 055169 号

总 发 行：华南理工大学出版社（广州五山华南理工大学 17 号楼，邮编 510640）
发行部电话：020-87113487 87111048（传真）
E-mail：scut202@scut.edu.cn http：//www.scutpress.com
责任编辑：胡 元
印 刷 者：广东省农垦总局印刷厂
开 本：787×1092 1/16 **印张**：15.25 **字数**：381 千
版 次：2004 年 8 月第 1 版第 1 次印刷
印 数：1～5 000 册
定 价：23.50 元

版权所有 盗版必究

前　言

C 语言是一种编译型程序设计语言，具有如下主要特点：语言简洁、紧凑，使用方便、灵活；语言表达能力强，可以表达各种数据结构；是一种结构化程序设计语言；兼有高级语言和低级语言的特点；具有较高的移植性。自 20 世纪 70 年代初推出以来，C 语言在国内外得到迅速推广和广泛应用，熟练掌握 C 语言程序设计是当代大学生应该具备的基本素质，也是学习面向对象的 C ++、JAVA、PHP 等语言的必备基础。目前，“C 语言程序设计”不但是计算机及其相关专业的专业基础课程，而且在不少高校也是非计算机专业公共基础课程。

本书由长期从事 C 语言程序设计教学一线工作的 4 位教师祝胜林、孙爱东、张明武和王琴共同编写完成。在编写过程中，我们参照了国内外有关 C 语言的著作和国内各种与 C 语言有关的等级或水平考试大纲，同时融合了自身的教学经验。因此，本书具有如下特点：

（1）系统性：可以满足学生系统地学习 C 语言和结构化程序的需要。

（2）针对性：本书的举例经过精心挑选，程序分析深入浅出，可以满足学生参加各种考试复习的需求。

（3）扩充性：对传统教材中的内容进行了扩充，增加了图形、系统调用、接口通信等方面的知识。

（4）新颖性：对传统的章节结构进行了重新编排，使内容更加紧凑，同时也更加突出程序设计的重点。

（5）实验配套：“C 语言程序设计”是一门实践性非常强的课程，为此，我们还编写了与之配套的实验教程，以提高学生的实际操作能力。

全书共分 10 章。第 1 章介绍 C 语言发展历史、C 程序的格式和特点以及 C 程序上机步骤。第 2 章介绍 C 语言的基本数据类型。第 3 章介绍运算与运算表达式的基本知识。第 4 章从算法实现的角度，介绍结构化程序的 3 种结构。第 5 章介绍数组及其应用。第 6 章从函数定义、调用和编译预处理等方面讨论函数的使用方法。第 7 章讨论 C 语言的精华部分——指针的定义和引用。第 8 章讨论 C 语言的构造数据类型的基本概念，以及链表的应用。第 9 章介绍 C 语言的文件概念和文件操作。第 10 章介绍图形、系统调用及汇编接口函数的用法。全书写作分工如下：祝胜林和王琴编写第 1、2 章，祝胜林编写第 3、4 章，孙爱东编写第 5、6 章，张明武编写第 7、8、9 章，王琴编写第 10 章和附录内容，由祝胜林负责统稿，严尚维主审。

本书在编撰过程中，得到了华南理工大学出版社、华南农业大学信息学院的大力支持和帮助，华南农业大学信息学院的肖德琴副院长、刘财兴副院长对本书提出了中肯的意见和建议，在此表示感谢！我们在编写过程中借鉴了国内外有关书籍，谨向这些书籍的作者表示真诚的感谢！

由于编者能力所限，书中如有不妥之处，敬请广大读者和专家批评指正。

编　者

2004 年 6 月

使 用 说 明

一、使用约定

为了帮助读者更好地阅读本书，对书中采用的一些约定或惯例介绍如下：

1. 按键约定

单个键：回车键（↙），制表键（< Tab >），取消键（< Esc >），功能键（F1 ~ F10）。

组合键：Ctrl + F1（方法：按住 Ctrl 键不放，再按功能键 F1），Ctrl + OI（方法：按住 Ctrl 键不放，再依次按字母 O 和 I 键），Alt + F5（方法：按住 Alt 键不放，再按功能键 F5）。

2. 输入形式

为了区分输入和输出形式，约定输入以下划线形式表示。如：year：1998↙表示数字 1998 通过键盘输入，以回车键结束输入。

二、教材配套

为了提高“C 语言程序设计”课程的理论教学与实践教学的效果，我们还编写了《C 语言程序设计实验教程》与本书配套。《C 语言程序设计实验教程》内容丰富，结构合理，层次清楚。一方面针对主教材各章节的知识要点进行总结归纳，详细分析了各种考试题型的解题方法与技巧，另一方面从读者角度设计了阅读程序、程序改错、分析程序、自行设计程序以及综合性、设计性实验等项目，由浅入深，帮助读者提高程序设计水平。

三、客户支持

对本书列举的所有源程序文件和电子教案，我们将它上载到华南理工大学出版社网站（http://www.scutpress.com），如有需要可自行下载，欢迎提出宝贵意见。联系方式：E-mail:zhushenglin@21cn.com。

目　录

第 1 章　概　　述

【学习目标】

◇ 了解 C 语言的发展历史和特点；
◇ 了解 C 程序的基本结构与特点；
◇ 初步掌握 C 程序的上机编写、调试程序的方法与步骤；
◇ 了解本章所介绍的常用数学库函数的功能与用法。

【重点与难点】

重点在于了解 C 程序的基本结构与特点；难点在于掌握 C 程序的上机调试方法与步骤。

1.1　C 语言的发展历史和特点

1.1.1　C 语言的发展历史

C 语言是一种编译型程序设计语言，也是应用最为广泛的计算机语言之一，是从事计算机应用、科学研究需熟练掌握的一种程序设计工具，同时又是学习面向对象 C++、JAVA等语言的基础。它产生的根源可以追溯到 ALGOL 60。1960 年出现的 ALGOL 60 是一种面向问题的高级语言，但它离硬件比较远，不适合编写系统程序。1963 年，英国剑桥大学推出了 CPL(Combined Programming Language,CPL)语言，CPL 语言在 ALGOL 60 的基础上与硬件接近了一些，但规模仍然比较宏大，难以实现。1967 年，剑桥大学的 Martin Richards 对 CPL 语言做了简化，推出了 BCPL(Basic Combined Programming Language, BCPL)语言。BCPL 语言是计算机软件人员在开发系统软件时作为记述语言使用的一种结构化程序设计语言，它能够直接处理与机器本身数据类型相近的数据，具有与内存地址对应的指针处理方式。1970 年，Ken Thompson（美国，贝尔实验室）以 BCPL 语言为基础，又做了进一步的简化，设计出比较简单而且很接近硬件的 B 语言（取 BCPL 的第一个字母）。但是，B 语言过于简单，数据没有类型，功能也有限。1972 ~ 1973 年间，Dennis M. Ritchie 和 Brian W. Kernighan（美国，贝尔实验室）在 B 语言的基础上设计出了 C 语言（取 BCPL 的第二个字母），又称 K&R C 语言(C 语言的起源顺序表示:ALGOL 60→CPL→BCPL→B→C)。

在 C 语言出现之后的 10 多年中，适用于不同计算机和不同操作系统的 C 语言编译系统相继问世，从而把 C 语言的应用推向一个更加广泛、普及的阶段。1983 年，美国国家标准化协会(American National Standards Institute, ANSI)根据 C 语言问世以来各种版本对 C 语言的发展与扩充，制定了新的标准，称为 ANSI C。1987 年，ANSI 又公布了新标准 87

ANSI C。1990 年，国际标准化组织(International Standardization Organization，ISO)接受了87ANSI C 为 ISO C 的标准(ISO 9899—1990)。目前流行的 C 语言编译系统都是以它为基础，如在微型计算机上使用的 Microsoft C、Turbo C、Quick C、Borland C 等，它们的语法和语句功能是一致的，差异主要体现在各自的标准函数库中收纳的函数种类、格式和功能上。本书以 87 ANSI C 为基础，同时兼顾现代 C 语言编译系统不同版本的通用性、一致性。

进入 20 世纪八九十年代，随着面向对象程序设计思想和可视化程序设计模式概念的推广、普及，C 语言也朝着支持面向对象、可视化程序设计语言方向发展，出现如 C++、Visual C++、C++Builder 等语言。C 语言已经成为编写系统软件、应用软件和进行程序设计教学的重要编程语言而风靡世界，成为世界上应用最广泛的计算机语言之一。

1.1.2 C 语言的特点

C 语言之所以成为当前软件开发的主流程序设计语言，取决于其不同于其他语言的独特优势，即自身的特点。C 语言的主要特点如下：

(1) 语言简洁、紧凑，使用方便、灵活。C 语言一共只有 32 个关键字（参考附录 2)、9 种控制语句，程序书写形式自由，压缩了不必要的成分。

(2) 语言表达能力强。C 语言运算符丰富，共有 34 种（参考附录 3)，且应用范围广；数据类型丰富，可以表达各种数据结构。

(3) 属于一种结构化程序设计语言。C 语言具有表达 3 种基本结构（顺序、选择和循环）的流程控制语句和实现模块化的函数结构、函数调用，符合现代编程风格的要求。

(4) 兼有高级语言和低级语言的特点；一方面，C 语言具有高级语言面向用户、容易理解、便于阅读和书写的优点；另一方面，C 语言具有与内存地址对应的指针处理方式，可以直接处理内存中的各种类型数据，能进行位运算，可实现汇编语言的大部分功能，可直接对硬件进行操作，具有接近汇编语言程序执行的高效率。

(5) 具有较高的移植性。C 语言程序本身并不依赖于计算机硬件系统和操作系统，从而便于在不同的硬件结构和运行不同操作系统的计算机间基本上不做修改就可实现程序的移植。C 语言提供的预处理命令可以提高软件开发效率，并为程序的组织和编译提供了便利，也提高了程序的可移植性。

1.2 C 语言程序的基本结构

C 语言和其他高级语言一样，按照特定的语法规定和相应的表现形式来构成程序的基本结构。遵循 C 语言程序基本结构的相关规定来编写程序，不仅可以方便程序设计人员和程序使用人员相互沟通，而且有利于提高程序调试效率。

1.2.1 C 程序结构

著名计算机科学家 Nikiklaus Wirth 曾经提出下面的公式：

数据结构 + 算法 = 程序

其中，数据结构(Data Structure)是指对数据的描述，即在程序中指定数据的类型和数据的组织形式；算法(Algorithm)是指数据操作的步骤。C 语言中各种数据类型表示了对数据的描述，各种语句按一定的顺序、步骤实现了对数据的操作。

一个 C 程序可以由若干个源程序文件组成，一个源程序文件又可以由若干个函数和预处理命令以及全局变量声明部分组成，其中函数由数据说明部分和执行语句组成。一个源程序文件构成的 C 程序结构可以用图 1.1 表示。在后续部分将逐步具体介绍相关内容。

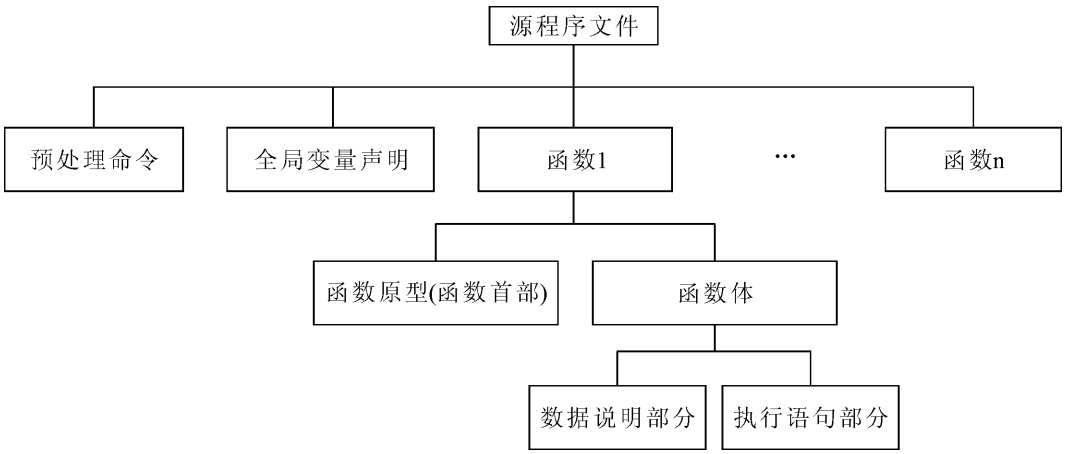

图 1.1　C 程序结构

1.2.2　简单的程序举例

本节首先介绍几个简单的 C 程序（只有一个源文件），然后归纳总结 C 程序的结构特点和格式约定。为了更好地理解举例的程序，先介绍一些必要的基本知识。

1. main 函数

每一个 C 程序必须有一个 main 函数，并且只能有一个 main 函数，它是程序执行的入口和出口。也就是说，C 程序的执行从 main 函数开始，调用其他函数后流程返回到 main 函数，在 main 函数中结束整个程序的运行。main 函数是由系统定义的。

2. 格式输入/输出函数

输入输出是以计算机主机为主体而言。从计算机向外部输出设备（如屏幕、打印机等）输出数据称为“输出”，从外部输入设备（如键盘、鼠标等）向计算机输入数据称为“输入”。C 语言的输入/输出是通过函数来实现的。在头文件 stdio.h 中定义了多种输入/输出函数，其中格式输入/输出函数 scanf/printf 就是标准的库函数，下面对 scanf/printf 函数作一个简单的介绍。

（1）printf 函数。printf 函数的一般格式：

```
printf（输出格式，输出系列）
```

它的功能是输出系列各项按照输出格式所指定的格式向外部输出设备输出数据。

①输出格式。是用双引号("")界定的字符串，也称转换控制字符串。它包含两种信

息，一种是格式说明，由%和格式字符组成，如十进制整数%d，浮点小数（实型数）%f等。格式说明的作用是将输出的数据转换为指定的格式输出。

注意：格式说明总是由%开始，并且%与格式字符之间不能插入空格。

另一种是普通字符，即需要原样输出的字符。如 printf("Circle area = %.2f",s)；中的“Circle area =”是普通字符，“%.2f”是格式说明，即指定输出项 s 的值以小数形式且保留两位小数显示,%f 默认以 6 位小数形式输出。另外，可以没有输出系列，譬如只输出字符串 printf("\nHello!");。

②输出系列。是需要输出的数据。如果有多个输出项，则输出项与输出项之间用逗号(,)分隔，输出格式中的格式说明与输出系列的输出项个数必须相同，并且它们按照各自的先后顺序一一对应。譬如：

printf("...%d...%x...%f...",a, b, c)；中 a 与%d、b 与%x、c 与%f 一一对应。

（2）scanf 函数。scanf 函数的一般格式：

scanf（输入格式，输入系列）

它的功能是输入系列各项按照输入格式所指定的格式向计算机输入数据。

①输入格式。它与输出格式相同，只不过普通字符必须原样输入。

②输入系列。它与输出系列基本一致，但输入项必须为地址量，如 &r 就是一个地址量，其中'&'是取地址运算符。在使用多个输入项的 scanf 函数时，键盘输入的各项数据之间要使用正确的分隔符，譬如：

scanf("%d%d", &a, &b);在输入时可以用空格、Tab 或回车键作为分隔符：3 4↙。

scanf("%d,%d", &a, &b);在输入时普通字符(,)必须原样输入：3, 4↙。

3. 标识符(Identifier)

标识符是用来标识变量名、符号常量名、函数名、数组名、类型名、文件名的有效字符序列。C 语言规定标识符只能由字母(A ~ Z 或 a ~ z)、数字(0 ~ 9)和下划线 3 种字符组成，且第一个字符不能为数字。如：变量名 a、b、r，符号常量名 PI，函数名 min。

注意：标识符是大小写敏感的，即大写字母和小写字母被认为是两个不同的字符，如 sum 与 Sum 表示不同的标识符。

4. 用户自定义函数

从用户使用的角度看，函数分为标准函数（即库函数）和用户自定义函数两种。

（1）标准函数：是由系统提供的，用户不必自己定义这些函数，可以直接使用它们，如：printf/scanf 函数。有些库函数必须在文件开始处用#include 预处理命令包含头文件，如：#include <stdio.h>。

（2）用户自定义函数：用以解决用户的专门需要。例如，int min(int x, int y)函数用来返回 x、y 中的最小值。函数的概念、内容比较多，本书第 6 章专门讲解，在此仅简单介绍函数定义的一般形式：

```
返回值的类型　函数名（形式参数说明）
{
    数据说明部分;
    执行语句部分;
}
```

其中，返回值的类型和形式参数说明可以没有，但函数名后的括号（）不能省略。形式参数有多个的时候，需要分开说明，并且参数之间用逗号进行分隔。譬如，int min(int x, int y)中 int x 与 int y 就是分开说明并且用逗号分隔。函数体通过 {} 括起来，由两个部分按前后顺序构成。数据说明部分在执行语句部分之前，顺序不能颠倒也不能交叉。例如，下面的函数定义违反了这一约束而出现错误：

```
main()
{
    int a,b;      /*定义变量 a、b 为整型(整数)*/
    a=5;b=8;      /*将整数 5 存入变量 a,将整数 8 存入变量 b*/
    int c;        /*定义变量 c 为整型*/
    c=a+b;
    printf("\n a+b=%d",c);      /*\n 回车换行,光标回到下一行开始处。向
                                  屏幕输出 a+b 的和*/
}
```

其中 a=5；b=8；执行语句部分插入到数据说明部分（变量定义部分）而出错。

5. 算术表达式

（1）基本的算术运算符：加法（+）、减法（-）、乘法（*）、除法（/）、求余数或称求模运算（%）。加、减、乘的用法与数学用法相同，除法和求余数运算在以后遇到时再详细讲解。

（2）算术表达式：用算术运算符、括号将运算对象（也称操作数）连接起来的、符合 C 语法规则的式子，称为 C 算术表达式。运算对象可以是常量、变量、函数等。譬如 1+2、PI*r*r、min(a,b)，其中 PI 是代表 π 的符号常量，min 是函数。

（3）优先级和结合性：优先级是指在表达式求值时，先按运算符的优先级别高低顺序执行。算术运算符的优先级是“先乘除后加减”，与数学用法相同。结合性是指各种运算符的结合方向。算术运算符的结合方向是“自左向右”（或“先左后右”），也与数学规定相同。

6. 注释

由“/*”开始，由“*/”结束，在“/*”和“*/”中间放置注释的文本内容。如：“/* The first C program. */”。注释的作用是增加程序的可读性，主要用来说明程序的功能、用途、符号的含义等内容。由于在编译程序时系统会将注释信息剔除，不包含在目标程序中，所以它并不影响程序的执行效率。

7. 程序举例

例 1.1　在输出屏幕上输出一行文本：The first C program.。（源程序名：eg0101.c）

```
1:  /* The first C program. */
2:  main( )
3:  {
4:      printf( " \n The first C program. " ) ;
5:  }
```

说明：

请务必注意每行的数字和冒号，如 1：表示行号，并不属于程序本身。

第 1 行：注释行。

第 4 行：向屏幕输出一行文本。\n 表示新的一行(new line)，即光标回到下一行的开始处。

程序运行情况：

```
The first C program.
```

例 1.2 从键盘输入半径，计算圆的面积和周长并输出到屏幕上，保留两位小数。(源程序名：eg0102. c)

```
1:  #define PI 3. 14
2:  main( )
3:  {
4:      float r, s;
5:      printf( " \nr = " ) ;
6:      scanf( " %f" , &r) ;
7:      s = PI * r * r;
8:      printf( " Circle area = %. 2f" ,s) ;
9:  }
```

说明：

第 1 行：预处理命令，定义符号常量，因为 π 在输入源程序时不能输入。

第 4 行：定义半径 r、面积 s 为浮点型 (实数)。

第 5 行：提示输入半径值。

第 6 行：按照浮点型格式输入半径值，并存入变量 r 中。

第 7 行：将所求圆面积公式 πr^2 的值存入变量 s。

第 8 行：按照浮点型格式向屏幕输出圆的面积，.2 表示保留两位小数。

程序运行情况：

```
r = 1
Circle area = 3. 14
```

例 1.3 从键盘上输入两个整数，求其中最小值并将结果输出到屏幕上。(源程序名:eg0103. c)

```
1:  main( )
2:  {
3:      int a, b, c;
4:      printf( " \n a, b:" ) ;
```

```
5:        scanf("%d, %d", &a, &b);
6:        c = min(a,b);
7:        printf("\n min value = %d", c);
8:  }
9:  int min(int x, int y)
10: {
11:       if(x > y) return (y);
12:       else return(x);
13: }
```

说明：

第 3 行：定义变量 a、b、c 为整型。

第 4 行：提示输入 a、b 的形式。

第 5 行：输入两个整数，分别存入变量 a、b。

第 6 行：将 min 函数的返回值赋值给（存入）变量 c。

第 7 行：按照整数形式向屏幕输出结果。

第 9 行：定义 min 函数，形式参数 x、y 为整型，返回值为整型。

第 11 行：如果 x > y，返回 y 的值。

第 12 行：否则（即 x≤y），返回 x 的值。

程序运行情况：

```
a,b:12,13
min value = 12
```

8. 程序书写习惯

根据上面 3 个程序举例，归纳程序的书写习惯如下：

（1）习惯上，C 程序一般使用英文小写字母（增加易读性），符号常量或特殊意义的符号通过大写字母形式与其区分开来。

（2）C 程序由一个或多个语句构成，每个语句以分号(;)结束。

（3）C 程序书写格式自由，一行内可以写多个语句，一个语句也可以分别写成多行。一般情况下，每个语句占用一行。

（4）缩进编排(Indent)。它是指不同结构层次的语句，从不同的起始位置开始，即在同一结构层次中的语句，缩进同样的字数。

（5）为了增强程序可读性，程序中可以适当添加空格和空行。

（6）注释。恰当的注释可以增加源程序的可读性。

1.3　C 程序的上机调试过程

C 语言是一种编译型程序语言。调试一个 C 程序需要经历 4 个基本步骤：编辑、编译、连接和执行。下面首先解释、比较 4 个步骤的实质含义和上机流程，再介绍上机环境，重点介绍 Turbo C 2.0 的集成环境(Integrated Developing Environment,IDE)。

1.3.1 源程序的编辑、编译、连接与执行

（1）编辑(Edit)。编程人员把程序代码输入计算机的过程和修改已经存在的代码的过程就是编辑。编辑生成的文件叫源程序（源文件），源文件名的后缀一般要求为 .c，如 ex0101.c。用来建立源文件的程序软件叫编辑器(Editor)，如 Unix 下的 vi、ed，MS-DOS 下的 Edit、Wps、Wordstar 等，Windows 下的记事本等，或者程序编写的集成环境，如 Turbo C 2.0、Borland C++3.1 等的集成环境。

（2）编译(Compile/Make)。对源程序的语法和逻辑结构等进行检查以生成目标文件(Object)的过程就是编译。编译过程是通过编译程序（或称编译器 Compiler）进行的。不同版本的 C 编译系统使用方法也不一样，使用时需要参考相应的资料，Turbo C 2.0 的编译程序(Tcc.exe)可以方便地通过集成环境的菜单命令执行。如果没有语法错误的源文件经过编译后生成目标文件，目标文件名的后缀在 Unix 下为 .o，在 MS-DOS 下为 .obj。如果源文件中存在不符合 C 语言规则的格式或语句时，会出现编译错误，在屏幕上显示错误的位置和种类，方便编程员通过编辑器修改源程序。

（3）连接(Link)。源文件经过编译后产生的目标文件是浮动的程序模块，不能直接运行。连接的作用是使用系统提供的连接程序（或称连接器 Linker）把目标文件、其他目标程序模块与系统提供的标准库函数有机组合起来，生成可以运行的可执行文件。可执行文件名的后缀在 Unix 下为 .out，在 MS-DOS 下为 .exe。当程序调用了不存在的函数或函数名时，就会出现连接错误，这时同样需要重新修改源程序。

（4）执行(Run)。可执行文件生成后，可以直接运行，在 MS-DOS 下通过直接键入主文件名后按回车键来运行，如：ex0101↙，Windows 下可通过双击可执行文件图标运行。执行程序并输入相应的测试数据，如果得到正确的结果，表示程序上机调试成功；如果得到的结果不符合要求，表示程序存在逻辑错误，这种情况需要重新修改源程序以剔除逻辑错误。

（5）上机调试程序的流程。完成 C 程序编写后，还需要进行上机调试。调试程序要经过以下几个步骤：

①上机输入与修改源程序。

②对源程序进行编译，如果有错误（语法错误），需转到第①步修改程序；如果没有错误，则进入第③步。

③与库函数进行连接，如果指定的函数不存在而出现错误（连接错误），需转到第①步修改程序，没有错误则进入第④步。

④执行程序，如果运行的结果不正确而出现错误（逻辑错误），需转到第①步修改程序。运行结果正确则结束调试。

上机调试程序的流程如图 1.2 所示。