高等学校计算机教材

王载新　曾大亮
杨有安　崔珂梅　编

程序设计基础

（C语言）

清华大学出版社

高等学校计算机教材

程序设计基础（C 语言）

王载新　曾大亮
编
杨有安　崔珂梅

清华大学出版社
北　京

内 容 简 介

C 语言是现代最流行的通用程序设计语言之一，它的简洁、紧凑、灵活、实用、高效、可移植性好等优点深受广大用户欢迎。C 语言的数据类型丰富，它既具有高级程序设计语言的优点，又具有低级程序设计语言的特点：既可以用来编写系统程序，又可以用来编写应用程序。因此，C 语言正在被迅速地推广和普及。

本书从计算机语言和程序设计的基本知识、C 语言的发展与特点出发，全面、系统地介绍 C 语言程序设计中的变量、运算符号、表达式、数据类型、存储类别、语句、函数等，还由浅入深地介绍程序设计的基本方法和算法。

本书可以作为高等院校非计算机专业的程序设计和 C 语言的教材，也可以作为初次学习 C 语言程序设计的读者的参考书。

图书在版编目（CIP）数据

程序设计基础（C 语言）/王载新，曾大亮，杨有安，崔珂梅编. —北京：清华大学出版社，2003
ISBN 7-302-07876-9

Ⅰ. 程… Ⅱ.①王… ②曾… ③杨… ④崔… 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（2003）第 123918 号

出 版 者：清华大学出版社　　地　　址：北京清华大学学研大厦
http://www.tup.com.cn　　邮　　编：100084
社 总 机：010-62770175　　客户服务：010-62776969
责任编辑：马　丽
封面设计：钱　诚
版式设计：张红英
印 刷 者：北京嘉实印刷有限公司印刷
装 订 者：三河市化甲屯小学装订二厂
发 行 者：新华书店总店北京发行所
开　　本：185×260　印张：25　字数：560 千字
版　　次：2004 年 1 月第 1 版　2004 年 1 月第 1 次印刷
书　　号：ISBN 7-302-07876-9/TP・5723
印　　数：1～12000
定　　价：29.00 元

前　言

C 语言是近年来在国内外都得到迅速推广的一种现代通用的程序设计语言。它具有丰富的数据类型、灵活方便的多种运算符、新颖的控制流程和数据结构以及简洁的表达式。它的处理能力强、运算速度快、目标效率高，具有完善的结构化、模块化程序结构。它既具有高级语言的优点，又具有低级语言的许多特点；它既适合于编写系统软件，也适合于编写应用软件。由于 C 语言的通用性及可移植性，所以可以在不同的机器上移植 C 语言程序。学习和使用 C 语言已成为许多程序员的迫切要求。

在高等院校，非计算机专业选修 C 语言课程的学生越来越多，由于总学时的限制，许多专业往往只安排学习一门计算机语言课程。为此，我们编写了本书，以满足程序设计初学者的需要。我们力求语言通俗易懂，并由浅入深，循序渐进地介绍 C 语言及其程序设计。由于本书针对的是计算机程序设计的入门者，为此，在介绍 C 语言之前简要地介绍了计算机语言方面的基本知识。本书在介绍如何使用 C 语言的同时，对程序设计中的基本概念和算法都做了详细介绍。

为了让读者了解程序设计的基本要求与技巧，本书选用了大量不同类型的、易于理解的例题和各种习题。我们认为，C 语言虽然适合于编写系统软件，但也适合于编写应用软件，多数非计算机专业的读者可能主要是编写应用软件。编写系统软件需要有一定的硬件知识。本书介绍的例题不涉及太多的硬件知识，以便一般的读者也能理解和掌握。有了这个基础，今后实践中需要时再扩展有关的知识是不困难的。

本书介绍的是 C 语言的最基本部分。

本书由华中科技大学计算机科学与工程学院教师编写。全书共分 11 章，其中第 1、9、10、11 章和附录由王载新编写，第 2、3、4 章由曾大亮编写，第 5、6 章由杨有安编写，第 7、8 章由崔珂梅编写。由于时间仓促及编者水平所限，书中错误和不当之处在所难免，敬请读者批评指正。

编　者

2003 年 10 月于华中科技大学

目　录

第 1 章　C 语言概述

C 语言是一种非常流行和深受程序设计者欢迎的通用程序设计语言。为了适应初次学习程序设计的读者要求，本章先简要介绍 C 语言的历史、C 语言的主要特点以及编写 C 语言程序的说明；然后介绍简单的 C 语言程序、C 语言程序的开发过程和上机操作。

1.1　计算机语言

语言是人们交换思想的工具，我们日常生活中使用的汉语、英语等称为自然语言。计算机诞生以后，人们要指挥计算机工作就产生了计算机语言。计算机诞生的初期，人们使用的计算机语言仅由 0 和 1 代码组成，被称为机器语言。指令是人们指挥计算机进行某种操作的命令。指令的集合称为程序。用机器语言编写的程序难写、难读和难修改，使计算机的推广使用受到了极大的限制，在计算机诞生后的一段时间里只有少数专业人员能使用计算机。随后人们使用便于记忆的符号代替 0 和 1 组成的指令，便产生了符号语言（或称汇编语言）。由汇编语言编写的程序要经过汇编程序将其翻译成机器语言程序，计算机才能执行。用机器语言或用汇编语言编写程序（称程序设计）时都离不开具体的计算机指令系统，用它们编写程序在技术上过于复杂，效率不高，故被称为低级语言。

随着计算机的发展，20 世纪 50 年代中期诞生了计算机高级语言，用高级语言编写的程序有易写、易读、易修改的优点，高级语言的出现使计算机的使用得到迅速普及。到目前为止，世界上有数百种高级语言，但常用的不过几十种（如 FORTRAN、PASCAL、C、LISP、COBOL 等）。用汇编语言或高级语言编写的程序称为源程序，高级语言源程序必须由相应的编译程序将它翻译成相应的汇编语言程序或机器语言程序，经翻译得到的程序称为目标程序。

1.2　C 语言简介

1.2.1　C 语言的历史

C 语言的历史是从 BCPL 语言和 B 语言演化而来的。BCPL 是 1967 年 Martin Richards 为编写操作系统软件和编译器而开发的语言。Ken Thompson 在模拟了 BCPL 语言的许多特

点的基础上开发了 B 语言，并于 1970 年在贝尔实验室用 B 语言在一台 DEC PDP-7 计算机上实现了第一个 UNIX 操作系统。BCPL 和 B 语言都是“数据无类型”语言，即每一个数据项都占用内存中的一个字，处理数据项的责任落在了程序员的身上。

C 语言是贝尔实验室的 dennis Ritchie 在 B 语言的基础上开发出来的，1972 年在一台 DEC PDP-11 计算机上实现了最初的 C 语言。C 作为 UNIX 操作系统的开发语言而开始广为人们认识。实际上，当今许多新的重要的操作系统都是用 C 或 C++编写的。在过去 20 年内，C 语言已经能够用在绝大多数计算机上了。C 语言是与硬件无关的。由于 C 语言的设计严谨，把用 C 语言编写的程序移植到大多数计算机上是可能的。在继承 BCPL 和 B 语言的许多重要概念的同时，C 语言增加了数据类型和其他功能强大的特点。

C 语言在各种计算机（有时称为“硬件平台”）上快速推广并因此产生了许多 C 语言版本。这些版本虽然是类似的，但通常是不兼容的。对希望开发出的代码能够在多种平台上运行的开发者来说，这是他们面临的一个严重的问题。显然，人们还需要一种标准的 C 语言版本。1983 年，美国国家标准化协会（ANSI）根据 C 语言诞生以来各种版本对 C 语言的发展和扩充，制定了新的标准，称为 ANSI C。1987 年，ANSI 又公布了新标准——87 ANSI C。目前流行的 C 编译程序都是以它为基础的。本书的叙述也基本上以 87 ANSI C 为基础。目前广泛流行的各种版本的 C 语言编译系统虽然基本部分是相同的，但有些部分仍然不同。在微型机上使用的 Microsoft C、Turbo C、Quick C 等，它们的不同版本又略有差异，因此读者在使用具体 C 编译系统时，还应通过阅读有关手册了解它的具体规定。

1.2.2 C 语言编程说明

C 语言共有 32 个关键词，又称保留字，它们一般由英语单词或其缩写组成。C 语言的语句有严格的语法要求，不能任意改变语句格式，不过控制语句仅有 9 种。C 语言程序中字母的大小写是区分的，其中保留字和主函数名必须用小写字母表示，标识符也习惯用小写字母表示。C 语言程序书写形式自由，但是，为了便于阅读习惯采用分层缩进的格式。C 语言简洁、紧凑，使用方便、灵活。相对其他语言源程序，C 语言源程序输入的工作量较少。

C 语言的数据类型丰富，有整型、实型、字符型、数组型、指针型、结构型、联合型和枚举型等，能用来实现各种复杂的数据结构。C 语言运算符丰富，共有 34 种，其中包括其他语言没有的一类位（bit）运算，C 语言把括号、赋值、强制类型转换等都作为运算符处理，从而使 C 语言的运算符类型极其丰富，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。C 语言具有很强的数据处理能力。这些既是 C 语言的优点，也是学习 C 语言的难点，我们必须深刻理解和区分它们的含义和用法，在编程中才能正确灵活地使用。

C 语言程序中可以使用如#define、#include 等编译预处理，能进行字符串或特定参数的宏定义，以及实现对外部文本文件的读取和合并，同时还具有#if、#else 等条件编译预处理

语句。这些功能的使用有利于提高程序质量和软件开发的工作效率。C 语言是一种结构化程序设计语言，它具有结构化控制语句（如 if else、while、do while、switch、for 等语句）。C 语言用函数作为程序模块，以实现程序的模块化。因此，在程序设计中应该采用结构化、模块化程序设计方法。

C 语言既具有高级语言的特点，又具有低级语言的一些功能。C 语言程序的可移植性好，用 C 语言编写的程序只需很少的改动或不作任何改动就可以在不同的计算机上运行。C 语言表达力强，生成的代码质量高，C 语言代码效率要比其他高级语言代码效率高，所以 C 语言既可以用来编写系统软件，也可以用来编写应用软件。但是，C 语言也有不足之处，C 语言的编译程序对语法检查不太严格。例如，对数组下标越界不作检查，由程序编写者自己保证程序的正确。C 语言对变量的类型使用比较灵活。例如，整型与字符型和逻辑型数据可以通用。C 语言允许程序编写者有较大的自由度，放宽了对语法的检查。因此，我们在编写程序时应当仔细检查程序，保证其正确性，而不要过分依赖 C 语言编译程序去查错。

1.3　简单的 C 语言程序介绍

用 C 语言编写的程序，称为 C 语言源程序，简称 C 程序。下面介绍几个简单的 C 程序。

【例 1-1】输出一行信息的 C 程序。

```
main()
{
   printf ("Hello,good morning!\n");
}
```

该程序的作用是输出以下一行信息：

```
Hello,good morning!
```

其中，main 表示“主函数”。C 程序是由一个或多个具有相对独立功能的程序模块组合而成，这样的模块称为函数，每个 C 程序必须有一个 main 函数。函数体由大括弧{ }括起来。本例中主函数内只有一个输出函数调用语句，printf 是 C 语言中的输出函数（详见第 9 章）。双引号内的字符串原样输出。“\\n”是换行符，即在输出“Hello,good morning!”后回车换行。语句最后有一分号。

【例 1-2】计算两数之和的 C 程序。

```
main()                    /*求两数之和*/
{ int a,b,sum;            /*变量说明*/
```

```
    a=50; b=45;
    sum=a+b;
    printf("sum=%d \n",sum);
}
```

该程序的作用是求两个整数 50 和 45 之和。程序中的第 1、2 行的“/*……*/”表示注释部分。注释是为了提高程序的可读性而增加的，对程序的编译和运行不起作用。注释可以加在程序的任何位置。

第 2 行为说明语句。说明变量 a、b 和 sum 的值为整型数（int）。int 必须小写，它是 integer（整数）的缩写。说明该句的作用是让编译程序根据变量类型为变量分配存储单元，变量的名字是为存储单元取的名字。在这里 a、b、sum 各单元存放的数据分别为：50、45、95，如图 1-1（a）所示。需要说明的是这里用到的三个变量名所表示的三个内存单元不一定是相邻的，之所以称做变量名，是因为在这种状态下重新赋值后，所存内容会发生变化。例如：

```
a=123;b=456;sum=a+b;
```

则 a、b、sum 的内容便成为如图 1-1（b）所示的情况，所以把它们叫做变量。第 3、4 行中出现的“=”是赋值运算符，表示把赋值运算符右边的数值或运算结果（如第 4 行）赋值给左边的变量。第 5 行中“%d”是输入、输出的“格式字符串”，用来指定输入、输出时的数据类型和格式，“%d”表示十进制整数类型，在执行输出时此位置上代以一个十进制整数值。printf 函数中括号内最右端的 sum 是要输出的变量，现在它的值为 95（即 50+45 之和），因此输出一行信息为：

```
sum=95
```

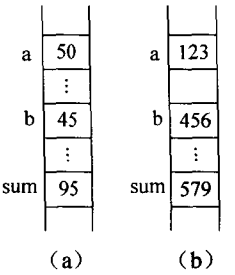

图 1-1　变量的存储形式

【例 1-3】包含函数调用的 C 程序。

```
main()                          /*主函数*/
{ int a,b;                      /*变量说明*/
  scanf("%d",&a);               /*输入变量 a 的值*/
  b=abs(a);                     /*调用 abs 函数，将得到的值赋给变量 b*/
  printf("/%d/=%d\n",a,b);      /*输出 a 和 b 的值*/
}
```

```
int abs(x)                          /*定义 abs 函数，函数返回值为整型，x 为形式参数*/
int x;                              /*形参说明*/
{ int y;                            /*abs 函数中的变量说明*/
  if (x>0) y=x;
  else y=-x;
  return(y);                        /*将 y 的值返回调用处*/
 }
```

该程序包括两个函数：主函数 main 和被调用函数 abs。abs 函数的作用是将 x 的绝对值赋给变量 y。return 语句将 y 的值返回给调用函数 main 的调用处。main 函数中的 scanf 是“输入函数”的名字（注：scanf 和 printf 都是 C 语言提供的标准输入、输出函数）。scanf 函数括号中变量 a 前面的“&”的含义是“取地址”，这里 scanf 函数的作用是给以变量 a 的地址所标志的单元输入数据。关于 scanf 函数详见第 10 章。

main 函数中第 4 行为调用 abs 函数，调用时将实际参数 a 的值传送给 abs 函数中的形式参数 x。经过执行 abs 函数得到一个返回值（即 abs 函数中变量 y 的值），把这个值赋给变量 b。然后输出 a 和 b 的值。printf 函数中双引号内的“/%d/=%d\n”，在输出时，其两个“%d”将先后由 a 和 b 的值取而代之，“/ /=”原样输出。程序运行结果如下：

```
-123 ↓          （输入-123 给 a）
/-123/=123      （输出 a 和 b 的值）
```

例中用到了函数调用、实参和形参等概念，这里只作了简单的解释。读者如对此不大理解，可以先不予深究，在以后有关章节时就会理解。在此介绍的这个例子是让读者对 C 程序的组成和形式有一个初步的了解。

通过以上几个例子，可以看到以下几点。

（1）C 程序是由函数构成的。一个 C 程序至少包含 1 个 main 函数，也可以包含一个 main 函数和多个其他函数。被调用的函数可以是系统提供的库函数（例如 printf 和 scanf 函数），也可以是用户根据需要自己编制的函数（如例 1-3 中的 abs 函数）。

（2）一个函数由两部分组成：

① 函数的定义部分。包括函数名、函数类型、函数属性、函数形参名、形式参数类型。一个函数名后面必须跟一对圆括号，函数参数可以没有，如 main()。

② 函数体，即函数说明部分下面的大括号{}内的部分。如果一个函数内有多对大括号，则最外层的一对{}为函数体的范围。

函数体一般包括：

- 变量说明。如例 1-2 中 main 函数中的“int a,b,sum”；
- 执行部分。由若干个语句组成。

当然，在某些情况下也可以没有变量说明（如例 1-1）。甚至可以既无变量说明，也无执行部分。如：

```
main()
{}
```

它是最小的合法的 C 程序但它不执行任何操作。

（3）一个 C 程序总是以 main 函数开始执行的，而不论 main 函数在整个程序中的位置如何，即 main 函数可以放在程序的最前，也可以放在程序的最后，或是放在程序的中间。当然不能放在其他函数中间，因为 C 语言函数不能嵌套定义。

（4）C 程序书写格式自由，语句可以从任一列开始书写，一行内可以写多个语句，一个语句可以分写在多行上。

（5）C 程序中无论是执行语句还是说明语句，每一个语句最后必须有一个分号，即使是程序的最后一个语句也必须有分号，分号是语句结束的标志。

（6）C 语言本身没有输入、输出语句。输入和输出的操作是由库函数 scanf 和 printf 等函数来完成的。

（7）C 程序中可以用/*…*/对任何部分作注释，以增加程序的可读性。注意，注释不能嵌套，如/*…/*…*/…*/是错误的。

1.4 C 语言程序的开发过程

用 C 语言编制程序到完成运行，一般要经过编辑、编译、连接、运行几个阶段，下面对在 Turbo C 环境下运行 C 程序做简单介绍。

Turbo C 集成开发环境（Turbo C 中的 TC）用起来十分方便，因为它集编辑、编译和调试于一体，无需独立的编辑、编译、连接程序就能建立并运行调试 C 程序。当磁盘上已安装了 Turbo C 时，用其运行 C 程序的步骤如下。

1．编辑 C 源文件

有两种进入编辑状态的方法。

（1）设要编辑的源文件名为 file.c，可输入命令：

```
TC file ↓
```

以上未输入文件的扩展名，系统自动给文件加上扩展名“.c”。如果它是一个新文件，则 Turbo C 屏幕窗口空白等待输入源文件；如果它是一个已存在的文件，则系统将其调入内存并在屏幕窗口上显示其内容，这时用户可以根据需要进行修改。

（2）输入：

```
TC ↓
```

屏幕显示 Turbo C 版本等信息，按回车键后该信息消失，屏幕顶部留下一排“命令”