普通高等教育“十一五”国家级规划教材　计算机系列教材

C程序设计教程

（第3版）

林小茶　陈昕　编著

清华大学出版社

普通高等教育“十一五”国家级规划教材　计算机系列教材

林小茶　陈昕　编著

C程序设计教程

（第3版）

清华大学出版社
北京

内 容 简 介

尽管随着计算机技术的飞速发展，高级程序设计语言的种类越来越多，但是C语言仍然是最适合作为学习程序设计思想的入门语言。本书在内容的编排上，更多地考虑了初学者的需求。

本书主要内容包括：C语言的基础知识、结构化程序设计、模块化程序设计、数组、指针、结构和文件。

全书的内容从易到难，循序渐进，列举了大量的能够解决实际问题的实例，并有一个贯穿始终的例子，将一个小程序逐渐扩充成一个比较大的程序。同时，特意安排了一些与信息安全方向有关的小例子，增加趣味性。

最后一章还讨论了两个实例，帮助读者了解和掌握编写实用的能解决实际问题的C程序的方法。

本书主要是为初学程序设计语言的高校学生量身定做的，也可作为C语言自学者的教材或参考书。

图书在版编目(CIP)数据

C程序设计教程/林小茶等编著. —3版. —北京：清华大学出版社，2018
(计算机系列教材)
ISBN 978-7-302-48948-1

Ⅰ. ①C…　Ⅱ. ①林…　Ⅲ. ①C语言－程序设计－教材　Ⅳ. ①TP312.8

中国版本图书馆CIP数据核字(2017)第285929号

责任编辑：张　民
封面设计：常雪影
责任校对：时翠兰
责任印制：丛怀宇

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦A座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn，010-62795954

印 装 者：三河市铭诚印务有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：22.25　　**字　　数**：513千字
版　　次：2005年10月第1版　2018年6月第3版　　**印　　次**：2018年6月第1次印刷
印　　数：1～1500
定　　价：49.80元

产品编号：072649-01

前　言

本教材是在前两版的基础上，总结了教学过程中的经验，征询了部分专家的意见，并考虑了读者和学生的需求，经过进一步修订而成的。

主要的修订内容包括：

(1) 选用了一些新的案例。这些案例不但更适合初学者，也更有实际意义。例如，增加了一些与信息安全概念相关的小程序，加密、解密和信息隐藏等；又例如，将所缴税的计算方法已经修改为最新的，即 2013 年开始实施的计算方法；增加了一些有关奥运会的例子，将前一版有关奥运会的案例中的相关数据全部改为里约奥运会的数据。希望读者学习起来更感兴趣。

(2) 将“函数说明”的提法修改为“函数声明”，全局变量等存储类别的“变量说明”修改为“变量声明”，对结构体类型的“说明”也修改为对结构体类型的“声明”。

(3) 摒弃了“字符串变量”的提法，直接采用更准确的术语“字符数组”。

(4) 将大部分程序的架构改为“int main( ) {… return 0; }”。

(5) 尽量符合 C99 的标准，例如注释符全部改为“//”。

本书由林小茶和陈昕共同编写，除了共同讨论全部章节的写作思想和内容，陈昕主要负责每章典型错误分析和第 8 章部分程序的编写。

最后，借此次本书再版的机会，向使用本书作为教材和学习参考书的教师和读者表示衷心的感谢，并殷切希望您对本书的内容和编写方法提出宝贵的意见和建议。

由于编者水平有限，疏漏在所难免，请广大读者批评指正。

作　者

目　录

第1章　C语言概述

C语言是一种通用的程序设计语言，由于其功能非常强大，因此可以用来完成一些非常复杂的工作。很多操作系统都是用C编写的，例如UNIX，MS-DOS，Microsoft Windows及Linux等。C语言具有高效、灵活、功能丰富、表达力强和移植性好等特点。

1.1　程序设计语言

程序在日常生活中是一个被经常使用的词汇。例如，2016年里约奥运会开幕式在里约热内卢马拉卡纳体育场举行，程序如下：

07:04　演员们开始了表演，狂欢正式开始。
07:55　运动员的入场仪式。
10:14:38　国际奥委会主席巴赫先生致辞。
10:29:15　巴西代总统特梅尔宣布：里约奥运会开幕。
10:38:09　运动员代表宣誓。
10:38:22　代表裁判员宣誓的田径裁判宣誓。
10:38:35　篮球教练员桑托斯代表教练员宣誓。
10:50:18　升奥运会会旗。
10:54:47　点燃圣火的仪式。

上述里约奥运会开幕式的程序是用自然语言描述的。这是我们每个人都能懂的语言，是人际交流使用的最常见的一种方式。

下面是一段C语言程序，其功能是在计算机的显示器上显示“欢迎”字样。

【例1.1】　在计算机的屏幕上显示“欢迎”字样的C程序。

```
//显示"欢迎"字样的 c 程序
#include "stdio.h"
int main()
{
    printf("欢迎");
    return 0;
}
```

人与人的交流需要语言，人与计算机的交流也同样需要语言，这就是程序设计语言。通过程序设计语言，人们可以教会计算机做一些事情，从而减轻自己的工作强度。

程序设计语言是一组用来定义计算机程序的语法规则。也就是说，程序设计语言必须是计算机能够理解的，如果我们把上面的C语言程序稍加修改，计算机就不能理解了。例如，将语句“printf("欢迎");”改为“printf("欢迎);”（注意，只差了一个双引号）。尽管

从学习程序设计的角度来说，程序设计的语法不是重点，但是，由于计算机不能理解语法错误的程序，因此学习语法也是必要的。

计算机程序是由计算机指令构成的序列。计算机按照程序中的指令逐条执行，就可以完成相应的操作。程序例 1.1 的指令是在显示器上显示“欢迎”，计算机执行该程序时，就会完成相应的工作，此时，我们可以在显示器上看到“欢迎”字样。

程序员使用程序设计语言定义计算机完成特定任务所需要使用的数据和计算机的动作。计算机程序的操作对象是“数据”。例 1.1 中使用的数据是"欢迎"（在 C 语言中称为字符串），计算机的动作是在显示器上显示数据"欢迎"（这里要注意的是，双引号并不在屏幕上显示出来，显示的只是双引号中的内容：欢迎）。

计算机自己不会做任何工作，它所做的工作都是由人们事先编好的程序来控制的。程序需要人来编写，使用的工具就是程序设计语言。目前，通用的计算机还没有识别自然语言的能力，而只能识别特定的计算机程序设计语言。一般情况下，计算机程序设计语言分为两类，一类是低级语言，另一类是高级语言。

低级语言的主要特点是直接依赖计算机的硬件系统，而高级语言不再依赖计算机的硬件系统，用高级语言编写的程序几乎可以不加修改地运行在不同机型的计算机上。

需要强调的是，无论采用何种语言来编写程序，程序在计算机上的执行都是由 CPU 所提供的机器指令来完成的。机器指令是用二进制表示的指令集。每种类型的 CPU 都有与之对应的指令集。

1.1.1 低级语言

低级语言包括机器语言和汇编语言。

计算机内部只能识别二进制，直接使用二进制 0 和 1 表示的指令序列来编程的语言就是机器语言。使用机器语言编写的程序，都是用 0 和 1 表示的，例如，“10111000111010000000000011”的功能是将 1000 送入寄存器 AX 中。使用机器语言编写程序，对程序员来说，不仅需要高超的技巧，还要特别心细，哪一位数字错了都不行，同时，程序员必须准确无误地牢记每一条指令的二进制编码。

机器语言的优点是执行速度快，并且可以直接对硬件进行操作，例如主板上的 BIOS，可以编写设备的驱动程序等。

机器语言的缺点也是显而易见的。首先是可读性差，就是编写程序语句“10111000111010000000000011”的人也未必马上就能看懂该句表示的是什么命令；其次，是可维护性差，用机器语言编写的程序很难看懂，又如何谈维护呢？最后就是可移植性差，因为不同的机型有自己的一套机器指令，与其他机型的机器指令不兼容。另外，用机器语言编写程序的生产效率低下，并且不易保证程序有好的质量。

为了方便记忆和编程，人们用一些符号和简单的语法来表示机器指令，这就是汇编语言。汇编语言是特定机器的机器指令的助记符，它依赖于特定的 CPU 体系结构。例如，“10111000111010000000000011”用汇编语言表示就是“mov ax,1000”。用“mov ax,1000”表示“将 1000 送入寄存器 AX 中”，可以看出可读性要好于机器指令。但是 CPU 并不能

识别汇编语言，因此，需要一个“翻译”程序将汇编语言翻译成机器语言，我们把这种将汇编语言翻译成机器语言的程序称为“汇编器”。汇编语言与机器语言的指令是一一对应的，所以，除了提高了一些可读性，汇编语言从根本上并没有改变机器语言的特点。可以说，汇编语言是面向机器语言的。当然，汇编语言也仍然具备机器语言的优点。许多大型系统（例如操作系统）的核心部分都是用汇编语言写的，因为这部分工作需要很高的效率，直接和硬件打交道。

由于用低级语言编写的程序不具备良好的可读性、可维护性和可移植性，因此人们发明了高级语言。

1.1.2 高级语言

高级语言是一种比较接近自然语言和数学语言的程序设计语言。高级语言非常符合人类的逻辑思维，抽象程度大大提高，高级语言的出现大大提高了程序员的工作效率，降低了程序设计的难度，并改善了程序的质量。用高级语言编写的程序看起来更像是英语，很容易读懂，不但使程序具备良好的可读性和可维护性，而且使更多的人掌握了程序设计方法，从而使计算机技术得到迅速的应用和普及。

例如，在程序例 1.1 中，main 的意思是主要的，说明函数是主函数，printf 的意思是输出，输出信息到显示器上，也就是在显示器上显示信息。

而下面的语句段

```
if  (u>=v)
    max=u;
else
    max=v;
```

表示的是“如果 u 大于或等于 v，则 max＝u，否则 max＝v”。对于稍稍有点英语基础的人就很容易理解语句的含义，也便于记忆。需要注意的是，这里的“＝”与数学语言等号有着本质的区别，不能等同，我们将在介绍 C 语言的运算符时，详细地加以讨论。

当然，由于计算机内部不能直接识别高级语言，从高级语言到机器语言要经过编译程序进行“翻译”，就像不懂英语的人要与讲英语的人交流需要翻译一样。一条高级语言的语句对应若干条机器指令，高级程序设计语言在不同的平台上会被编译成不同的机器语言。也可以说，高级语言独立于机器的特性是靠编译程序为不同机器产生不同的机器指令来实现的。因此，用高级语言编写的程序具有很好的可移植性。

编译程序分为两种，一种是解释系统，另一种是编译系统。解释系统是对高级语言编写的程序翻译一句执行一句，解释系统的工作与我们生活中的同声翻译的工作很相似；而编译系统是将高级语言编写的程序文件全部翻译成机器语言，生成可执行文件以后再执行，就像开会现场没有同声翻译，每个人拿着事先翻译好的演讲者的文稿看，也同样能明白演讲者在说什么。高级语言几乎在每一种机器上都有自己的编译程序，编译程序又称编译器。C 语言的编译程序属于编译系统。

C 语言具有高级语言的基本特征，也可以像低级语言一样对位、字节和地址进行操

作，因此很多学者认为 C 语言是中间语言。但是，由于本书是针对初学者的，不会对 C 语言的低级语言特征进行过多的讨论，本书的大部分内容涉及的将是 C 语言的高级语言特征。

1.2 通过实例认识 C 程序的结构

用 C 语言编写的程序就是 C 程序，也可以称为 C 语言源程序。下面我们将通过几个简单的程序例子，使读者对 C 程序的构成有个感性的认识。

1.2.1 问候界面

用户界面是人机交互的窗口，它直接影响程序的使用价值。作为一个好的程序员，应该建立这样的理念，一个内部设计良好但用户界面不好的应用程序就不是一个好的程序。尽管在本书中我们将不会讨论如何将界面设计得像 Windows 一样漂亮，本书的程序中使用的也都是一些非常简单的界面，但是，还是希望初学者能明白好的用户界面是为了给使用者（也包括自己）带来方便。

例如，当程序在读写磁盘时，应该在屏幕上显示消息："正在读写磁盘，请稍候..."；而程序需要用户从键盘输入用户名时，应该提示用户"请输入用户名："。

例 1.1 就是一个简单的问候界面，同时可以作为测试调试环境的最简单的程序。

尽管这是一个非常小的程序，功能也十分简单，但通过该程序我们还是能够清楚地了解 C 程序构成的要点。同时，这个程序也很实用，一般可以用它测试编译环境是否能够正常使用。

下面从几个方面分析程序例 1.1。

（1）在 C 语言中书写注释的方法。

书写注释是为了说明程序的功能和目的以及一些编程细节，如果想做一名优秀的程序员，必须养成为程序写注释的好习惯。

在 C 程序中，"//"后面表示是注释，编译系统会跳过注释行，不会对注释行的内容进行翻译。按照惯例，一般要在程序的开始说明整个程序的目的和功能，并在必要时，为每一组（甚至是每一行代码）写出注释，以增加可读性。

（2）C 源程序的主函数结构。

每个 C 程序都必须包含一个主函数 main()，也只能包含一个主函数。例 1.1 中只有一个主函数。从"int main()"到符号"}"结束，是对主函数的定义。C 程序的执行是从主函数内的第一句执行语句开始，到主函数中的最后一句结束。主函数以及 C 程序中的每个函数都必须用一对大括号将一段程序括起来，用一对大括号括起来的部分是一个程序模块，在 C 语言中也称为分程序，每个函数中都至少有一个分程序。

（3）C 语句的语句结束符。

分号";"是 C 语句的结束符，C 语句一般包括执行语句和声明语句。

"printf("欢迎");"是执行语句，用分号结束。初学者经常忘记在语句后面书写分号，

编译系统会对缺少分号的程序给出错误提示。

(4) C语句的书写格式。

C语句的书写格式是比较自由的。C语言的语法不硬性规定语句从某列开始书写，但是好的程序员应该学会使用缩进格式，例如，“printf("欢迎");”语句在main函数内部，书写时不能与int main()对齐，而是向右移动了几个格。

下面的程序也能通过编译并正确执行，但是书写格式很别扭，不容易看懂。

```
#include "stdio.h"
int main(){printf
("欢迎"); return 0;}
```

(5) C语言中关键字和特定字的特征。

C语言的关键字和特定字使用小写字母。int和main是关键字，include是特定字，都必须用小写。初学者往往忽略这一点，以为main与MAIN是相同的。如果将主函数写为MAIN()，编译程序将提示出错信息。

(6) printf是C语言提供的标准输入输出库函数，它的功能是将用两个双引号括起来的内容“欢迎”输出到标准输入输出设备显示器上。

(7) 预处理命令的使用。

以“#”开始的语句是预处理命令。本例中的#include "stdio.h" 语句就是一条预处理命令。注意预处理命令的后面是不带分号的！

预处理命令是在编译系统对C语言的命令进行翻译之前需要由预处理程序处理的语句。“#include "stdio.h""语句的功能是要求预处理程序将stdio.h文件的全部内容放置到程序中来，作为程序的一部分。文件stdio.h中包括一些重要的定义，在本程序中的“printf("欢迎");”语句的执行需要stdio.h文件的帮忙，没有这条预处理命令，该程序将不能通过编译系统的“翻译”。系统提示的编译错误是“Function 'printf' should have a prototype”，意思是说“函数'printf'没有原型”，而printf函数的原型就包含在文件stdio.h中。

1.2.2 计算里约奥运会中国军团新人的人数

【例1.2】 根据报道，里约奥运会中国军团有711人，新人占总人数6成以上，计算新参加奥运会的人数至少是多少。

```
//-------------计算新参加奥运会的人数-----------------------
#include "stdio.h"
int main()
{  int  total,new_total;          //变量定义
   total=711;                     //参加里约奥运会的中国总人数
   new_total=total * 0.60;        //计算
   printf("中国新参加奥运会的人数至少是 %d.\n",new_total);     //输出
   return 0;
}
```

运行结果：

```
中国新参加奥运会的人数至少是 426.
```

分析与说明：

(1) 变量与变量的定义。

例 1.2 中的“int total, new_total;”定义了两个变量，这两个变量实际代表了两块存储空间，分别命名为 total 和 new_total，而 int 说明这两个存储空间的数据类型为整型，int 是类型声明符，C 语言中还有 float，char 和 double 等类型声明符。

变量是由程序命名的一块计算机内存区域，用来存储一个可以变化的数值。每个变量保存的必须是一个特定的数据类型的数值。

C 语言规定，任何变量都要先经过定义，才能在程序中使用。变量定义实际上意味着存储空间的分配。如果没有进行存储空间的分配，数据就无法存储，这样的 C 程序不能通过编译。

例如：将例 1.2 中的定义语句“int total, new_total;”去掉，程序将不能通过编译系统的编译。系统会提示两个编译错误：“undefined symbol 'total'”和“undefined symbol 'new_total'”，意思分别是“'total'没有定义”和“'new_total'没有定义”。

(2) 使用直接常量(又称无名常量或文字常量)。

常量是在程序执行过程中不会变化的数值，直接常量就是在代码中直接书写的数值，没有名字。例如，“total =711;”一句中的 711 和“new_total= total * 0.60;”中的 0.60 都是常量，711 是整型常量，0.60 是浮点常量。

(3) 赋值运算符“=”。

赋值运算符是高级程序设计语言中一个非常特殊的运算符，任何一种高级程序设计语言中都必须提供该运算符。赋值运算符的功能是用赋值号右边的值覆盖赋值号左边的变量单元，也可以说，是使赋值号左边的变量的内容与赋值号右边的值相等。

赋值运算符最简单的用法是：赋值运算符的左边是一个变量，右边是一个常量。其功能是将右边常量的值送到左边的变量中，使赋值号左边的变量内容与常量相等。例如，“total =711;”就表示使 total 中的内容变为 711。

注意：尽管 C 语言的赋值号恰巧使用的是数学上的等号，但是在概念上是截然不同的，不要将其与数学上的等号混淆。例如，针对整数，“x=x+1”在数学上是不可能成立的，但是，在 C 语言中这是一个被程序员经常使用的语句，表示将变量 x 的内容增加 1，假如原来 x 的内容是 1，执行语句“x=x+1”后，x 的内容变为 2。

(4) 运算符“*”。

C 语言的算术运算符与数学符号是类似的，“new_total= total * 0.60;”表示将 total 的值乘以 0.60，结果送入到 total 变量中。注意：由于 C 语言不能识别数学上的乘号，因此用星号代替。

(5) printf 函数的调用格式。

在例 1.1 和例 1.2 中都调用了系统提供的库函数 printf 函数，该函数负责显示内容

到标准输出设备——显示器上。如果不将运算结果显示到屏幕上，是无法调试程序的。在此，先简单地介绍一下调用 printf 函数的格式：

```
printf(" 格式信息", 数据参数 1, 数据参数 2,...);
```

其中，数据参数可有可无。如果有，它们的输出格式控制信息应该包括在“格式信息”中。

① 格式信息中字符除了冠以斜杠“\”和“%”的字符，其他字符原封不动按照原样输出到屏幕上。

例如，“printf("中国新参加奥运会的人数至少是 %d.\n",new_total); ”一句中的“中国新参加奥运会的人数至少是”是原样输出的。

② 格式信息中的%和其后面的字符 d 分别是转换声明符和转换字符(合起来称为转换声明)，它指定了显示参数时的格式。在%和转换字符之间还可以加一些特殊字符，用来控制输出的域宽等。

例如，“printf("中国新参加奥运会的人数至少是%d.\n",new_total);”一句中的“%d”负责控制变量 new_total 的输出格式。

C 语言规定，转换声明符的个数应与数据参数的个数相等。

“printf("%d %d %d\n",x,y,z);”是正确的，但是，“printf("%d %d \n",x,y,z);”和“printf("%d %d %d\n",x,y);”都不正确。最值得引起注意的是，后面这两条语句能够通过编译，而执行结果可能不对！

③ 格式信息中的“\n”是字符转义序列。“\n”表示换行。

1.2.3 计算有线电视 n 年的费用

【例 1.3】 计算有线电视 3 年的费用，并在屏幕上显示结果。

```
//-------------求有线电视 n 年的费用----------------------
#include "stdio.h"
void show(int year);                    //函数声明
int main()
{  show(3);                             //计算有线电视 3 年的费用
   printf("再见!\n");
   return 0;
}
void show(int year)                     //(自定义)函数定义
{  int a,b,total;                       //变量定义
   a=18;                                //每个月的费用
   b=12;                                //一年 12 个月
   total=a*b*year;                      //计算
   printf("%d 年的费用是 %d 元.\n",year,total);        //输出
}
```