面向21世纪高等院校计算机规划教材

C语言及程序设计

杜忠友　刘　浩　叶曙光　姜庆娜　编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

面向21世纪高等院校计算机规划教材

C语言及程序设计

杜忠友　刘　浩　叶曙光　姜庆娜　编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书浓缩了作者多年教学改革的实践经验，以白皮书和CFC2006为指导，按照认识规律，对章节顺序进行了合理安排，做到先易后难、循序渐进，语言叙述注重图文并茂，理论讲解注重结合实际应用、能力训练。本书主要内容包括：C语言概述，数据类型、运算符与表达式，顺序结构程序设计，选择结构程序设计，循环结构程序设计，数组与字符串，函数，对函数的进一步讨论，指针，结构体、共用体、枚举和用户定义的类型，位运算，文件，上机实验与指导。

本书适合作为高等院校C语言程序设计课程的教材，也可作为高职高专、成人高等教育、社会培训的教材，还可作为C语言自学者的教材或参考书。

图书在版编目（CIP）数据

C语言及程序设计／杜忠友等编著. —北京：中国铁道出版社，2008.11
面向21世纪高等院校计算机规划教材
ISBN 978-7-113-09296-2

Ⅰ.C… Ⅱ.杜… 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（2008）第175328号

书　　名：C语言及程序设计
作　　者：杜忠友　刘　浩　叶曙光　姜庆娜　编著

策划编辑：严晓舟　陈士剑
责任编辑：黄园园　　**编辑部电话**：（010）63583215
封面设计：付　巍　　**封面制作**：白　雪
责任印制：李　佳

出版发行：中国铁道出版社（北京市宣武区右安门西街8号　邮政编码：100054）
印　　刷：三河市兴达印务有限公司
版　　次：2008年12月第1版　　2008年12月第1次印刷
开　　本：787mm×1092mm　1/16　**印张**：18.75　**字数**：441千
书　　号：ISBN 978-7-113-09296-2/TP・2999
定　　价：32.00元

前　言

“C 语言程序设计”在许多高校都作为程序设计语言的第一门课程进行教学。笔者从事 C 语言程序设计课程的教学已有 20 余年，在不断融合课程发展趋势和从事教学改革实践的过程中，积累了许多教学经验，有不少收获和感悟。笔者感到有必要将其整理出来，与大家进行资源共享。本书在编写过程中，以教育部高等学校非计算机专业计算机基础课程教学指导分委员会《关于进一步加强高等学校计算机基础教学的意见暨计算机基础课程教学基本要求》（简称“白皮书”）和中国高等院校计算机基础教育改革课题教研组发布的《中国高等院校计算机基础教育课程体系 2006》（简称 CFC2006）为指导。这有利于体现新的教学思维和时代感。

本课程向学习者介绍结构化（模块化）程序设计的基本思想和方法，通过学习结构化程序设计语言，了解用计算机解决问题的一般方法，掌握程序设计的思路和基本方法，掌握编写和调试简单应用程序的方法，使学习者养成利用计算机解决工作、生活中的实际问题的习惯，提高应用计算机方面的能力和素质。

本书的主要特色如下：

一、按照认识规律，对章节顺序做了合理安排，做到先易后难、循序渐进，叙述表达注重图文并茂、通俗易懂，理论讲解注重结合实际应用、能力训练（特别是编程能力和调试能力），并适时融入了分析、启发、引导性的内容和思考性的问题，有助于学习者很快进入角色，进而对本书、本课程产生兴趣，易于教学，易于自学。

二、内容的详与略、宏观总揽与细节说明的关系控制得当，提纲挈领、纲举目张，使得学习起来思路清晰、概念清楚，容易把握程序设计的基本思想。

三、例题、习题视野广阔、生动典型而有吸引力，能够使学习者在学习探求过程中“阅尽人间春色”，领略程序设计领域的经典题目，也会使学习者在吸引力、求知欲的作用下，在兴趣盎然的氛围中更好地领会、理解和掌握所学的语法规则、算法及程序设计的思想、方法、技巧，从而取得理想的学习效果。

四、最后一章是上机实验与指导，这有助于结合实际，强化操作，加强实践环节，激励创新意识，使学习者有针对性地进行上机实验，提高编程能力和调试能力。课本和实验融为一体，使得一书在手，课本实验全有，体现了全面性特色。

五、本书配有教学课件，供教、学双方参考使用。课件可至 http://edu.tqbooks.net 网站的“下载专区”中下载。

六、做到提升学生的知识—能力—素质，把握教学的难度—深度—强度，体现基础—技术—应用，提供教材—实验—课件支持。

本书共分13章，内容包括：C语言概述，数据类型、运算符与表达式，顺序结构程序设计，选择结构程序设计，循环结构程序设计，数组与字符串，函数，对函数的进一步讨论，指针，结构体、共用体、枚举和用户定义的类型，位运算，文件，上机实验与指导。另外包括4个附录：ASCII字符编码一览表，关键字及其用途，运算符的优先级别和结合方向，C库函数。

本书适合作为高等院校C语言程序设计课程的教材，也可作为高职高专、成人高等教育、社会培训的教材，还可作为C语言自学者的教材或参考书。

本书由山东建筑大学的杜忠友教授、刘浩教授，西北大学的叶曙光研究生，山东建筑大学的姜庆娜副教授编著。为了集思广义，吸收了山东建筑大学的多位老师参与编写：王晓闽老师参与了第1章的编写，孙晓燕老师参与了第2章的编写，张海林老师参与了第3章的编写，靳天飞老师和何淑娟老师参与了第6章的编写，姜玉波老师参与了第7章的编写，解艳艳老师和杨磊老师参与了第8章的编写，李锋老师参与了第11章的编写，赵欣老师参与了第12章的编写。

本书参考了大量的国内外文献，在此向这些文献的作者表示深深的敬意和衷心的感谢！

奉献给读者的这本书虽经反复修改，力求精益求精，但由于篇幅较大、问题复杂等原因，难免会有疏漏、不妥甚至错讹之处，恳请各位专家和读者提出宝贵意见（E-mail:du-zy@163.com），以便再版时将您的意见纳入书中，使本书愈来愈精品化，更好地为读者服务。

编　者

2008年10月

目　录

第 1 章

C 语言概述

学习“C 语言程序设计”这门课程的目的和意义是什么呢？本课程是向学习者介绍结构化（模块化）程序设计的基本思想和方法，通过学习结构化程序设计语言，了解用计算机解决问题的一般方法，掌握程序设计的思路和基本方法，掌握编写和调试简单应用程序的方法，使学习者养成利用计算机解决工作、生活中的实际问题的习惯，提高计算机方面的能力和素质。

为使学习者对 C 语言有一个概括的了解，本章将简单地介绍 C 语言的产生过程及特点、C 语言程序的结构与书写格式和 C 语言程序的开发过程等。

1.1 C 语言的产生过程及特点

1.1.1 C 语言的产生过程

20 世纪 60 年代，随着计算机科学的迅速发展，高级程序设计语言得到了广泛的应用。然而，还没有一种可以用于书写操作系统和编译程序等系统程序的高级语言，人们不得不用汇编语言（或机器语言）来书写，但汇编语言存在着不可移植、可读性差、研制软件效率不如高级语言等缺点，给编程带来很多不便。为此，人们对能用于系统程序设计的高级语言的开发就变得势在必行了。

1967 年，首先由 Martin Richards 开发出 BCPL（Basic Combined Programming Language，基本组合编程语言）。作为软件人员开发系统软件的描述语言，BCPL 语言的突出特点是：

① 结构化的程序设计。

② 直接处理与机器本身数据类型相近的数据。

③ 具有与内存地址对应的指针处理方式。

1970 年，Ken Thompson 继承并发展了 BCPL 的上述特点，设计了 B 语言。当时，美国 DEC 公司的 PDP-7 小型机 UNIX 操作系统，就是使用 B 语言开发的。

由于 B 语言过于简单，缺乏数据类型，功能有限，因此 1972 年至 1973 年间，美国 Bell 实验室的 Dennis M. Ritchie 对 B 语言做了进一步的充实和完善，正式推出了 C 语言。C 语言既保留了 B 语言精练、接近硬件的优点，又克服了 B 语言过于简单、数据无类型的缺点。随后 Ken Thompson

和 D. M. Ritchie 用 C 语言重写了 UNIX 操作系统 90%以上的代码，即 UNIX 第 5 版，这一版本奠定了 UNIX 的基础，使其逐渐成为最重要的操作系统之一。

随着 UNIX 的成功和广泛流行，C 语言也引起了人们的注意，并成为软件开发中深受人们欢迎的一种程序设计语言。随着 C 语言的广泛应用，出现了多种不同版本的 C 语言编译系统，1983 年，美国国家标准化协会（ANSI）制定了标准 ANSI C。1987 年，ANSI 又公布了 87 ANSI C。1990 年，国际标准化组织（ISO）接受 87 ANSI C 为 ISO C 的标准（ISO 9899：1990）。

1983 年，贝尔实验室在 C 语言的基础上又推出了 C++。C++扩充和完善了 C 语言，成为一种面向对象的程序设计语言。C++提出了一些更为深入的概念，它所支持的面向对象的概念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与传统的结构化程序设计不同的思维方式和编程方法。因此，C 语言是 C++的基础，学完 C 语言再学习 C++，会收到事半功倍的效果。

1.1.2 C 语言的特点

C 语言把高级语言的特征同汇编语言的能力结合了起来，因此，C 语言具有许多独到的特点，以下仅从使用者的角度加以讨论。

① C 语言短小精悍，基本组成部分紧凑、简洁。C 语言一共有 32 个标准的关键字（参见附录 B）、34 种运算符（参见附录 C）以及 9 种控制语句（见第 3 章），语句的组成精练、简洁，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的双重特点，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、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，这些丰富的数据结构和控制结构以及以函数调用为主的程序设计风格，保证了利用 C 语言所编写的程序结构化好。

④ C 语言提供了某些接近汇编语言的功能。这为编写系统软件提供了方便条件。如可以直接访问物理地址，并能进行二进制位运算等。

⑤ C 语言程序生成的目标代码质量高，程序执行效率高。C 语言程序所生成的目标代码的效率仅比用汇编语言描述同一个问题低 10%~20%，因此，用 C 语言编写的程序执行效率高。

⑥ C 语言程序可移植性好。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，如数据的输入、输出等都是通过调用系统提供的库函数来实现的，而这些库函数本身并不是 C 语言的组成部分。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 语言本身也有与优势相对立的弱点。

① 运算符多，不易记忆。C 语言运算符多，运算符有 15 种优先级，而且有些运算符的含义和优先级与人们的日常习惯不一致，这使得 C 语言在具有运算和处理方便、灵活优势的同时，也具有不容易记忆的弱点。

② C 语言的语法限制不太严格。例如，对数组下标越界不作检查，由编程者自己保证程序正确；一些变量的数据类型可以通用，如整型数据与字符型数据及逻辑型数据；放宽了语法检查，这在增强了程序设计的灵活性的同时，相应地检查错误的任务也部分地转到了编程者身上，这在一定程度上也降低了 C 语言的某些安全性，对编程人员提出了更高的要求。

C 语言的这些特点，大家现在或许还不能很好地理解，等学习完这门课程再想一想，就会有所感触。

1.2　C 语言程序的结构、书写格式和简单的 C 语言程序

1.2.1　C 语言程序的结构

C 语言程序由一个或多个函数组成，但任何一个完整的 C 语言程序，都必须包含一个且只有一个名为 main 的函数，它是程序运行时调用的第一个函数。一个完整的 C 语言程序除了 main 函数之外，还可以包括用户自己编写的函数。C 语言程序是函数式结构，函数是 C 语言程序的基本单位。其结构如下所示，其中 f1 到 fn 代表用户自定义的函数：

```
文件包含
宏定义
函数声明
条件编译
外部变量说明(即声明或定义)
结构体、共用体等的定义
main( )
{
   声明部分
   语句部分
}

类型 f1(参数)
{
   声明部分
   语句部分
}

类型 f2(参数)
{
   声明部分
   语句部分
}
…
类型 fn(参数)
{
声明部分
语句部分
}
```

其中，main 函数之前的 6 个部分不一定每个程序都有。

1.2.2 C 语言程序的书写格式

C 语言程序的书写格式如下：

① 用 C 语言书写程序时较为自由，既可以一行写一个语句，也可以一行写多条语句，一条语句也可以分几行来写。每条语句用英文“;”结尾。为了增加可读性，一般一条语句占一行，并可适当加一些注释或空行。注释以“/*”开头，以“*/”结束，注释的内容写在“/*”和“*/”之间。注释并不是程序中必须出现的内容，也就是说，编译系统并不理睬注释信息。

② C 语言程序要求关键字都使用小写字母。在 C 语言中，小写字母和大写字母是不同的。例如，小写的 int 是关键字，大写的 INT 则不是。关键字在 C 语言中不能用做其他目的，不能用做变量或函数的名字。

③ C 语言程序是结构化程序设计语言，为了层次结构分明，不同结构层次的语句从不同的起始位置开始书写，同一层次中的语句缩进左对齐。

书写格式可以从下面的 C 语言程序中看到。

1.2.3 简单的 C 语言程序

下面看一个简单的 C 语言程序。

【例 1.1】从键盘输入两个整数，求这两个整数之和并显示出来。

```
#include <stdio.h>
main( )
{
    int a,b,sum;
    scanf("%d%d",&a,&b);                          /*读入两个整数，存入变量 a 和 b 中*/
    sum=a+b;
    printf("The sum of %d and %d is %d",a,b,sum);  /*输出两数之和 sum*/
}
```

此程序共有 8 行。下面逐行进行解释。

① 第 1 行以“#”开头的是编译预处理命令，是指在编译代码之前由预处理程序处理的命令。本例中的#include <stdio.h>命令是将文件 stdio.h 找到并包含到程序中来，作为程序的一部分。

② 第 2 行的 main 是主函数名，C 语言规定必须用 main 作为主函数名。其后的圆括号中间可以是空的，但这一对圆括号不能省略。main 是主函数的起始行。每个 C 语言程序都必须有一个并且只能有一个主函数。一个 C 语言程序总是从主函数开始执行。

③ 第 2 行之后的 6 行是主函数体。主函数体自“{”开始到“}”结束。其间可以有说明部分（即声明部分或定义部分）和执行语句。一个 C 语言程序总是执行到主函数体的最后一个语句结束。

④ 第 4 行的 a、b、sum 都是变量。这一行是定义部分，定义它们都是整数型的变量。int 是关键字，必须小写。

⑤ 第 5～7 行的 3 条语句是执行语句。执行语句必须放到变量定义之后。

⑥ 第 5 行的 scanf 是输入库函数，它要求从键盘上输入两个整数（以空格等为分隔符），并将它们送到 a、b 变量中。它要求将 a、b 用其地址&a 和&b 表示，双引号中的两个%d 是十进制整数格式符，指明对变量 a、b 的格式要求。

⑦ 第 6 行是将 a 和 b 的值相加，相加的结果赋给变量 sum。

⑧ 第 7 行的 printf 是输出库函数，它的作用是进行格式化输出，其参数包括两部分：一是双引号内的“格式控制”部分，用于对输出的数据进行格式说明；二是“输出表列”部分，表示要输出的变量。所有这些语句都放在大括号{}之内，各语句之间用英文半角分号“;”隔开。

⑨ 第 5 行和第 7 行后面的部分是注释。注释的内容位于“/*”和“*/”之间，可用英文或中文进行书写。

⑩ 第 4 ~ 7 行属同一层次，缩进左对齐。其他 4 行属同一层次，左对齐。

该程序的一次运行结果如下：

输入为：

```
3  5↙        （↙表示按【Enter】键）
```

输出为：

```
The sum 3 and 5 is 8
```

下面再看一个 C 语言程序。

【例 1.2】 计算 s=1+2+3+…+50。

```
#include <stdio.h>
int sum(int x)                                  /*自定义 sum 函数，求 1+2+3+…+x*/
{
    int i,y;
    y=0;
    for (i=1;i<=x;i++)
    y=y+i;
    return(y);
}
main( )
{
    int n=50, s;
    s=sum(n);                                   /*函数调用，用实参 n 代替形参 x*/
    printf("%s %d\n","sum of 1+2+...+50 is",s);  /*输出 s*/
}
```

在该程序中，自定义了 sum 函数，用于计算 1+2+3+…+x，x 为参数。程序运行时先调用 main 函数，数值 50 赋给 n，然后调用 sum 函数计算 1+2+3+…+n 并将所求之值 y 返回调用函数 main，赋予变量 s，最后输出 s 的值。

该程序的运行结果如下：

```
sum of 1+2+...+50 is 1275
```

通过上面两个程序，大家对 C 语言程序已经有了一个初步的认识，可以看到：

① C 语言程序完全是由函数构成的，而且每个程序可由一个或多个函数组成。C 语言程序的函数式结构使得 C 语言程序非常容易实现模块化，便于阅读和维护。

② 一个 C 语言源程序不论由多少个函数组成，都有并且只能有一个 main 函数，即主函数。

③ C 语言程序总是从 main 函数开始执行，而不论 main 函数在程序的什么位置，也就是说，可以将 main 函数放在程序的任何位置。

④ 每一条语句都必须以分号结尾。但#include <stdio.h>是命令，不是语句，因此之后不能加分号。

⑤ C 语言中没有专门的输入/输出语句，这里的输入/输出是通过 scanf 和 printf 两个库函数实现的。

1.3 C 语言程序的开发过程及开发环境

1.3.1 C 语言程序的开发过程

在计算机上进行 C 语言程序开发通常包括 4 个步骤，即编辑、编译、连接和运行。

1. 编辑源程序（Edit）

用 C 语言编写的程序称为源程序。编辑源程序是指使用某种编辑软件，如记事本、Word、编译系统自带的编辑功能等对源程序进行编辑（包括输入、修改、保存等）的过程。

源程序经编辑由键盘输入后，形成源程序文件以文本文件的形式存储在外存储器（如 U 盘或硬盘）中。源程序文件的名字由用户选定，但扩展名均为.c（即源程序文件均带有扩展名.c）。

2. 编译源程序（Compile）

在程序运行之前，必须用系统提供的编译程序对源程序文件进行编译。编译程序要进行语法检查，若没有发现错误，则编译后产生目标文件。目标文件是由“目标代码”组成的，目标代码是二进制指令代码。目标文件的主文件名和源程序的主文件名一致，但扩展名为.obj。若编译程序发现有错误，则输出错误信息，此时程序员应对程序进行再编辑，改正程序错误后，再进行编译，直到编译正确为止。

3. 连接目标文件及库文件（Link）

源程序经编译后产生的目标文件尽管由二进制指令代码组成，但它还不能直接在计算机上运行，因为编译所生成的目标文件（*.obj）是相对独立的模块，需要通过连接程序把它和其他目标文件以及系统所提供的库函数进行连接装配。连接装配生成可执行文件后就可运行了。可执行文件的主文件名和源程序、目标程序的主文件名一致，但扩展名为.exe，并自动将它保存到磁盘上作为可执行文件。连接时如果出现错误，则应查看所需的库函数的目标文件是否存在。

4. 运行程序（Run）

运行经过编译连接后产生的可执行文件（.exe 文件），便可得到程序的运行结果。如果运行结果错误，则需要对算法进行检查，重新编辑源程序，直至得到正确的运行结果。

综上所述，开发一个 C 语言程序的过程如图 1-1 所示。

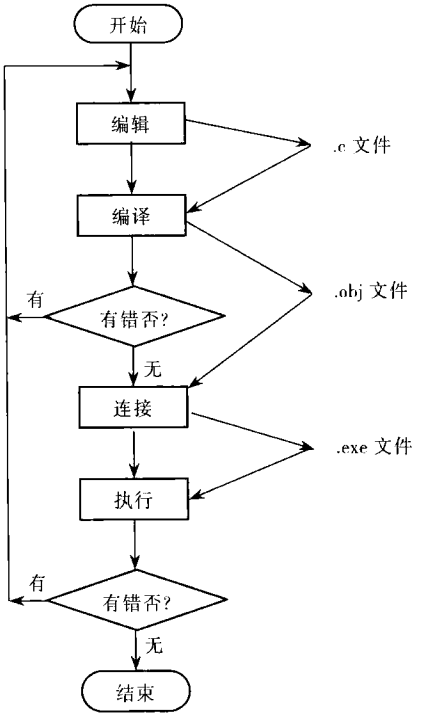

图 1-1 C 语言程序的开发过程