高等学校计算机专业规划教材

C语言与程序设计

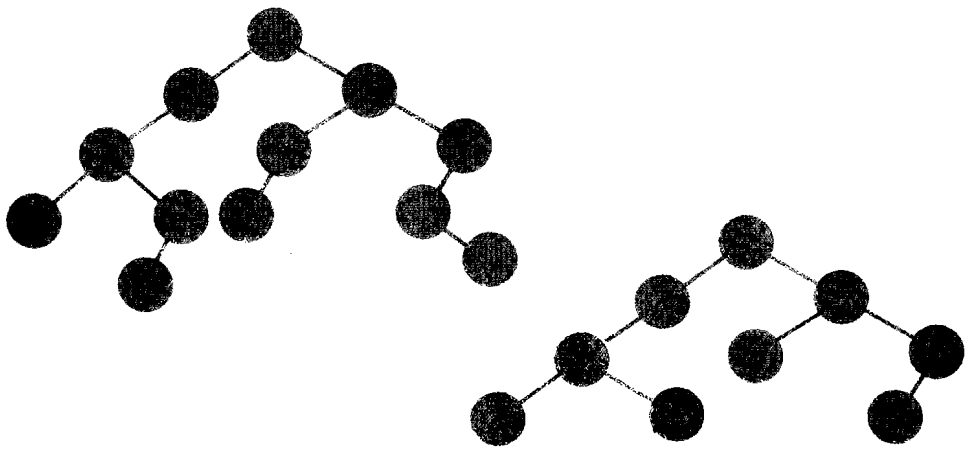
赵学军 钱旭 主编
李彬 杨峰 任培花 编著

清华大学出版社

高等学校计算机专业规划教材

C语言与程序设计

赵学军 钱旭 主编
李郴 杨峰 任培花 编著



清华大学出版社
北京

内 容 简 介

本书是根据教育部《关于进一步加强高校计算机基础教学的意见》而编写的。全书共分 9 章，内容包括概述、基础知识、结构化程序设计、数组、函数、指针、结构体和共用体、文件及 C++ 初步，前面部分是传统的 C 语言内容，是程序设计基础，后面部分介绍了面向对象程序设计初步知识。本书把传统的面向过程的内容与现代面向对象的内容有机结合与过渡学习，使读者能够尽快掌握程序设计基础知识。

本书可作为高等学校各专业的程序设计教材，也可作为培训和自学教材及辅导教材。

图书在版编目 (CIP) 数据

C 语言与程序设计 / 赵学军等主编. --北京：清华大学出版社，2013.2
高等学校计算机专业规划教材
ISBN 978-7-302-30481-4

Ⅰ. ①C… Ⅱ. ①赵… Ⅲ. ①C 语言－程序设计－高等学校－教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 250854 号

责任编辑：龙启铭
封面设计：常雪影
责任校对：时翠兰
责任印制：王静怡

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 刷 者：北京市人民文学印刷厂
装 订 者：三河市新茂装订有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：16.5　　**字　　数**：382 千字
版　　次：2013 年 2 月第 1 版　　**印　　次**：2013 年 2 月第 1 次印刷
印　　数：1～3000
定　　价：28.00 元

产品编号：049794-01

前言

本书主要根据教育部《关于进一步加强高校计算机基础教学的意见》对高等学校大学生针对计算机基础学习的基本要求而编写的教材。

多年来，C语言的特点决定了它，一直是我国高等学校学生对计算机技术学习的基础及主干课程，是高校的公共基础课，从而受到学校广大师生的重视。

随着计算机技术的发展以及教学实践的需要，对大学生的程序设计能力也有了更高的要求。不仅要求大学生掌握程序设计语言的基本概念和基础知识，还要提高其程序设计能力，不仅要求其能够编写面向过程的结构化程序，而且要初步掌握面向对象的程序设计技术。因此，本书不但将C语言基础与程序设计知识有机融合，而且将传统的面向过程程序设计与现代的面向对象设计内容有机结合。就是希望学生通过该课程的学习，打好计算机语言学习的基础，学会程序设计思路，在今后的学习和工作中，通过补充适当知识就能够编程解决各自专业领域的计算机应用问题。

目前，市面上的关于C语言教材及辅导很多，各有千秋。现就本教材的特点说明如下：

(1) 统一标准

此版教材注意吸取其他教材的优点，避其缺点，参考了最新的ASCII/ISO制定的C语言新标准C99，所有语法和关键字均遵守这一标准。书中所有程序代码从性能和风格上力求精简，程序完整，风格一致。

(2) 错误陷阱/重点提示

注重课程实践性特点，总结归纳了学生上机实践时常犯的错误，在每章后特别安排“错误陷阱/重点提示”，加以点拨。使学生掌握正确的、规范的编程方法，提高上机实践能力，发挥其学习主动性。

(3) 结构合理

根据教师多年的教学实践经验和学生上机指导经验，精心安排本书各个章节。编排顺序合理，重点突出，有利于学生循序渐进，逐渐深入，从而加深对计算机程序设计基本概念的理解，训练逻辑思维能力，培养严谨的科学作风。

(4) 程序设计方法的多样性和实际的编程环境

本书对程序设计方法的多样性进行探讨，对语言特征上的优缺点进行描述，引导学生融会贯通，举一反三，灵活应用，逐步提高；以往大部分C语

言教材都是以 Turbo C 为编程环境的，考虑到目前 C++ 的普及和广泛应用，本书使用 VC++ 编程环境，贴近实际应用。

(5) 内容充实

本书对难以理解的知识均由相关的图示来直观讲解。同时力求兼顾知识面的广度和深度，不仅涵盖了 C 语言的大部分知识点，而且对相关的数据结构、软件工程项目开发的知识及 C++ 初步知识做了讲解，以便向目前应用广泛的面向对象编程工具过渡。

根据作者从事计算机教育工作二十多年的经验，就如何更好地学习 C 语言与程序设计，谈几点建议：

(1) 近些年，由于计算机技术高速发展，面向对象的程序设计语言逐渐登上程序设计舞台，但这并不意味着面向过程的 C 语言过时了。在面向对象程序设计中仍要用到面向过程的思想，面向过程的程序设计还是计算机程序设计人员的基本功。在学习了 C 语言的基础上再学一些面向对象的初步知识，为今后进一步学习和使用计算机技术打下坚实基础。

(2) 学习 C 语言与程序设计课程过程中，既要精通语法知识又要构造设计好程序算法，即将语言基础与程序设计思想二者统一结合。

(3) 无论用哪种计算机语言进行程序设计，其基本规律都是一样的。因此，扎实学好 C 语言，在以后需要时可以很快掌握其他语言，达到举一反三的目的。

(4) 学生学习程序设计不是一朝一夕就能成为编程高手的，要把主要精力放在最基本的内容上，打好基本功最重要。对不同设计能力的学生可分层布置程序设计题目。

(5) C 语言与程序设计是强实践性课程，在掌握基本概念的同时要多动手编程并上机实践。所以，在设计考试题时，不仅强调概念题还要兼顾程序设计题及上机实践题。

(6) 能够运行 C 语言的编译器很多，但都有一定差异，应该尝试使用不同的编译环境。

本书共 9 章，分为三个部分：第一部分为前三章，分别是概述、基础知识及结构化程序设计；第二部分为中间三章，分别是数组、函数和指针；第三部分为后三章，分别是结构体和共用体、文件及 C++ 初步；前八章是传统的 C 语言内容，是程序设计基础。最后一章介绍面向对象程序设计基础知识。

作者建议本书的教学时数如下：课堂教学 36～54 学时，上机实践 40～60 学时。

本书由赵学军主编、审稿并统稿，钱旭主编及指导；第 1 章、第 2 章、第 3 章由李郴审稿及编写，王玉芹、李海芳参加编写及实验；第 4 章、第 5 章、第 6 章由赵学军审稿及编写，郭明华、彭翔、任培花参加编写及实验；第 7 章、第 8 章、第 9 章由杨峰审稿及编写，乔晓冬、尹权参加编写及实验；苏红旗、徐慧指导及审稿；任培花、李刚、李忠参加编写、实验及审稿统稿等工作。

最后，对多年来关心支持本书和本书作者的领导、同事和朋友们表示由衷的感谢。尤其对中国矿业大学(北京)的学校领导、教务处领导、机电学院领导和计算机系全体师生表示感谢！对清华大学出版社龙启铭编辑的密切合作与支持表示感谢！

由于水平有限，加之时间较紧，因此错误和问题难免，恳请读者批评指正。

作　者

2013 年 1 月于中国矿业大学(北京)

目录

第3章 C程序设计 /35

第4章 数组与字符串 /84

第 8 章 文件操作 /199

第 9 章 面向对象和 C++基础 /214

第 1 章

C 语言概述

1.1 C 语言的发展简史

C 语言于 20 世纪 70 年代初问世，是一种被广泛应用的程序设计语言。最初的 C 语言描述于 1978 年出现在 B. W. Kernighan 和 D. M. Ritchie 合著的“THE C PROGRAMMING LANGUAGE”著作中，通常简称为“K&R”版本，这是一部经典著作，它给出了 C 的标准。随着 C 语言的不断扩充、发展和普及，1983 年，美国国家标准化协会(ANSI)着手为 C 语言制定新的标准，该标准 1989 年被正式采用，称为 ANSI C。作为正常维护，1995 年、1999 年又对 ANSI C 进行了修订和补充，但仍保持原来的基本性质。目前流行的 C 编译系统都是以 ANSI C 为基础的。

早期的 C 语言主要用于 UNIX 操作系统。由于 C 语言的强大功能及优点逐渐为人们认识，到了 20 世纪 80 年代，C 开始进入其他操作系统，并很快在大、中、小及微型等各类计算机上得到了广泛的应用。目前微型计算机上使用的 C 编译系统主要有：Microsoft C、Borland C、Turbo C、Quick C 等，这些 C 编程环境都遵循着 ANSI C 标准，并且在此基础上各自作了一些扩充，使用起来更加便捷。

在 C 的基础上，1983 年贝尔实验室的 Bjarne Strou-strup 又推出了功能更加强大的 C++，C++ 进一步扩充和完善了 C 语言。作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，为程序员提供了一种与传统结构程序设计不同的思维方式和编程方法，更适合大型应用程序的开发，但是，C++ 也是最复杂的编程语言之一，C 是 C++ 的基础，C++ 和 C 在很多方面是兼容的。因此，掌握了 C 语言，再进一步学习 C++ 就能以一种熟悉的语法来学习面向对象的语言，从而达到事半功倍的目的。实际上，目前网络上广泛使用的 Java、JavaScript、C# 等语言也是在 C 语言的基础上发展起来的。

作为卓越的计算机语言之一，C 语言具有如下特点：

(1) C 语言是一种结构化程序设计语言，它层次清晰，便于按模块化方式组织程序，易于调试和维护。

(2) C 语言语法简洁、方便并灵活，可自由书写程序。

(3) C 语言的表现能力和处理能力极强，它不仅具有丰富的运算符和数据类型，便于实现各类复杂的数据结构，还可以直接访问内存的物理地址，进行位一级的运算(使之能够胜任开发操作系统)。

(4) 由于 C 语言实现了对硬件的编程操作，因此 C 语言集高级语言和低级语言的功能于一体，既可用于系统软件的开发，也适合应用软件的开发。

(5) 移植性强，可广泛移植到各种型号的计算机和不同的操作系统上，从而形成多种版本的C语言。

(6) 能够生成较高质量的目标代码，所以，程序执行时效率也高。

1.2 初识C程序

学习程序设计语言最有效的途径就是用它编写程序，下面我们通过几个简单的例子来了解C语言程序。

例1.1 输出一串字符。

```
/* ch1-1.cpp */
#include <stdio.h>                          /* 编译预处理 */
void main()                                 /* 主函数 */
{
    /* 调用 printf 函数输出结果 */
    printf("Hello,world!\n");               /* 语句 */
}
```

该程序的运行结果是输出一串字符：

```
Hello,world!
```

程序结构分析：

(1) 注释。

注释是对程序的注解，为了正确且清晰地反映程序的逻辑，可以在程序的任意位置加入注释，以提高程序的可读性，注释内容不参加程序的编译和执行。

注释部分书写形式如下：

```
/* 注释内容 */    或    //注释内容
```

"/*　　*/"为块注释，//为单行注释。有两点需要注意：首先，编译时，每个注释实际上都被视为一个空格；其次，注释不能出现互相嵌套的情况。

(2) 编译预处理命令。

程序第一行#include <stdio.h>是编译预处理命令，作用是提供C标准函数库中输入输出函数的有关信息，在编译前将文件"stdio.h"的内容加入到文件"ch1-1.cpp"中，作为其中的一部分。目前读者只需了解程序中只要用到C标准函数库中的输入输出函数，就应该在程序的开头写入这行代码，至于原因，后面的章节有详细讲解。

(3) 函数与主函数。

① 程序由一个或多个函数组成。

一个C程序，无论简单还是复杂，都是由一个或多个函数组成的，每个函数实现一个特定的功能，函数名可以按照编程者的喜好或根据函数的功能取名。

② 必须有且只能有一个主函数main()，()不能省略。

C程序总是从main函数开始执行，最后在main中结束，其他函数通过嵌套调用得以

执行。这意味着，一个C程序必须有，并且只能有一个main函数。另外，main函数前面的“void”表示该函数为空类型，即执行该函数后不产生函数值。

(4) 程序语句

① C程序由语句组成。

花括号{}用来框住函数中的所有语句，称函数体。该例中，main函数体只有一条语句：

```
printf("Hello,world!\n");
```

printf()是C标准函数库提供的输出函数，其作用是将双引号中的字符串原样输出，即显示双引号中的内容。“\n”是换行符，作用是将光标移至下一行行首。该例在输出字符串“Hello,world!”后将光标移至下一行行首，即回车换行。

② 用“;”作为语句终止符。

C程序中允许一行内写多条语句，或将一条语句分写在多行上，但每条语句结尾处必须有一个分号，分号是C语句的必要组成部分，是语句结束的标志。

例 1.2 求两个整数的和。

```
#include <stdio.h>
void main()
{
    /*定义三个整型变量,num_1、num_2表示数变量,sum表示和变量*/
    int num_1,num_2,sum;
    num_1=23;                                    /*对变量num_1赋值*/
    num_2=14;                                    /*对变量num_2赋值*/
    sum=num_1+num_2;                             /*求两个整数的和*/
    printf("num_1=%d , num_2=%d , sum is %d\n",num_1,num_2,sum);
}
```

程序分析：

(1) int num_1,num_2,sum; 是变量定义部分，这里定义num_1、num_2及sum是整型变量，其中可以存放整型数据。C语言规定任何变量都要先定义，再使用。

(2) 程序中

```
num_1=23;
num_2=14;
sum=num_1+num_2;
```

是三条赋值语句。=是赋值运算符，作用是把其右边表达式的值赋给左边的变量，这里是把值23赋给变量num_1，把值14赋给变量num_2，把num_1+num_2的运算结果赋给变量sum。

(3) 程序最后一条语句中的“%d”是输入输出格式说明符，输出时该符号并不显示，而是在此位置上以“十进制整数”的形式显示后面对应的变量的值(每一个%d顺次对应一个变量的值)，于是程序的输出结果为：

```
num_1=23 , num_2=14 , sum is 37
```

例 1.3 求圆面积。

```
#include <stdio.h>
#define PI 3.1415926
void main()
{
    float r,s;
    float area(float r);                    /* 函数原型声明 */
    scanf("%f ",&r);
    s=area(r);
    printf("半径为%f 的圆面积为%f",r,s);
}
float area(float r1)
{
    return(PI * r1 * r1);
}
```

程序分析：

(1) 该程序包含两个函数：主函数 main 和用户自定义函数 area，程序的功能由两个函数共同完成。

(2) 主函数 main 中的语句 float r,s; 是变量定义部分，这里定义 r,s 为浮点型变量，其中可以存放浮点型数据。

(3) main 中的 float area(float r1);是函数声明语句。该程序在主函数 main 中调用了 area 函数，函数声明语句的作用是使 C 的编译系统能够正确识别和调用 area 函数。

(4) main 中的语句 scanf("%f",&r); 的作用是将用户从键盘上输入的 1 个浮点型数据送入变量 r 中。scanf()是 C 标准函数库提供的输入函数，其中的"%f" 是输入输出格式说明符，它要求按照"十进制小数"的形式将数据输入到指定的变量中去。

(5) main 中的语句 s=area(r); 是函数调用语句，作用是调用 area 函数。调用过程中，主调函数将变量 r 的值传送给被调函数中的变量 r1，并执行被调函数 area 中的语句。

(6) 程序中 area 是用户自定义的函数，area 函数的 float area(float r1)是函数首部，表示 area 带有一个参数 r1，在 area 被调用时，这一个参数的值由主调函数中的对应参数(本程序中是 r 的值)传递过来；函数执行后将运算结果作为返回值带回，该返回值为 float 型。

(7) area 函数的功能是求半径为 r 的圆面积，并通过 return 语句将结果返回给主函数 main，返回值通过函数名 area 被带回到 main 函数中调用 area 函数的位置。

程序的运行情况如下：

用户从键盘输入：10.0 <回车>

输出结果为：半径为 10.000000 的圆面积为 314.159260

通过上面的例子，我们可以了解到 C 程序的结构特点如下：

(1) 函数是构成 C 程序的基本单位。一个 C 程序可包含多个函数，但必须有并且只

能有一个 main 函数，C 程序总是从 main 函数开始执行，最终还是由 main 函数中结束。

(2) 函数首部和函数体是组成函数的两部分。函数首部包括函数名、函数类型、函数参数(形参)名、参数类型。函数体是由函数首部下面的花括弧{}括起来的部分，如果一个函数内有多个花括弧，则最外层的一对为函数体的范围；函数体一般包括声明部分和执行部分，在声明部分定义变量或对所调用的函数进行声明，执行部分则由若干条语句组成。

(3) C 程序书写格式特点。

- 尽量选择见名知义的标识符，适当地加入注释；
- 不使用行号，无程序行概念；
- 可使用空行和空格；
- 常用锯齿形格式书写(可以通过 TAB 键、缩进、空行和空格配合使用形成锯齿形代码)。

1.3 C 语言实验环境

程序是一组计算机能识别和执行的指令。用高级语言编写的程序称为源程序(source program)，1.2 节中的例子就是 C 的源程序。从上述例子我们可以看到，用高级语言编写的源程序与人们熟悉的自然语言和数学语言非常接近，但计算机只能识别和执行由 0、1 组成的二进制的指令，不能识别和执行用高级语言编写的指令，因此，必须先用一套预先编好的起翻译作用的“编译程序”程序，把 C 源程序翻译成二进制形式的“目标程序(object program)”，然后将该目标程序与系统的函数库和其他目标程序连接起来，最终形成可执行程序。因此，在计算机上编写并执行一个 C 语言程序通常包括 4 个阶段：编辑、编译、连接和运行。

(1) 编辑(Edit)：在 C 语言编程环境中录入和编辑 C 源程序，并以磁盘文件的形式保存，一般用“.c”作为文件扩展名，如“ok.c”。扩展名为“.c”的文件称为 C 的源程序文件。

(2) 编译(Compile)：利用“编译程序”将编辑好的源程序转换成二进制代码，形成目标程序，系统自动将目标程序以磁盘文件的形式保存，文件名与其源程序文件名相同，扩展名为“.obj”。例如对“ok.c”编译后系统将自动产生目标文件“ok.obj”。编译过程中，编译程序首先对源程序进行分析，检查程序中存在的语法错误，并给出出错报告，编程人员根据报告对程序进行编辑修改，然后重新编译。这个过程往往会反复多次，直到编译通过为止。

(3) 连接(Link)：编译所生成的目标文件“*.obj”还不能直接执行，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，生成可执行文件才能够执行。连接过程中如果出现错误(如缺少需要的库函数等)，同样会给出错误信息报告。编程人员根据错误报告排除错误，重新连接，直到程序正确无误后自动形成扩展名为 exe 的可执行程序。系统自动将生成的可执行程序以磁盘文件的形式保存，文件名仍然保持原来的名字，扩展名为“.exe”。例如对“ok.obj”实现连接后系统将自动产生可执行文件

“ok.exe”。

(4) 运行(Run)：执行经连接生成的可执行文件，得到运行结果。这里需要注意的是，可能由于解决问题的算法存在问题而使程序编写具有逻辑错误，导致错误的运行结果；也可能由于语义上的错误，如用0做除数等，使得运行过程出现错误……此时，应该对算法进行检查，重新编写源程序，直至得到正确的运行结果。

综上所述，要执行一个C程序，应经过如图1-1所示的步骤。

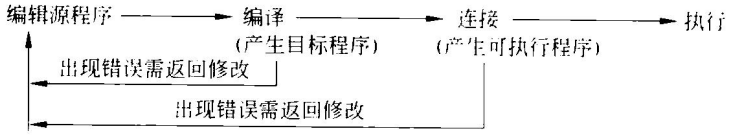


图1-1 执行C程序的步骤

为了编译、连接和运行C程序，必须有相应的C编译系统。使用较多的C编译系统有Turbo C、Turbo C++、Visual C++等，它们都是集成的开发环境，即C源程序的编辑、编译、连接及运行等都可在这个环境下进行。本书介绍用Visual C++ 6.0来编译C程序。

(1) 启动Visual C++ 6.0，进入如图1-2所示的Visual C++ 6.0集成环境。

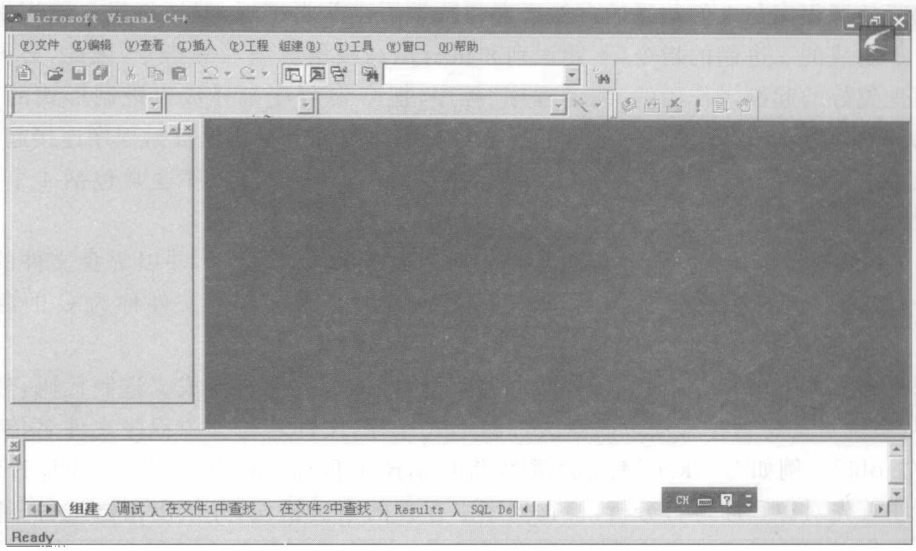


图1-2 Visual C++ 6.0集成环境

(2) 选择“文件|新建”命令，打开如图1-3所示的“新建”对话框。

(3) 单击“新建”对话框中的“文件”按钮，出现如图1-4所示的对话框。

(4) 在图1-4中，选择“C++ Source File”，同时在“文件名”中输入文件名，单击“确定”按钮，出现如图1-5所示的C源程序编辑界面。

用户可在源程序编辑界面输入C源程序，并用“文件”菜单中的“保存”命令保存。