高等职业教育计算机类专业“十二五”规划教材

C语言程序设计项目教程

主编 陈翠红 黄玲

CDIO教育理念

面向就业
能力培养
项目导向
任务驱动

国防工业出版社
National Defense Industry Press

高等职业教育计算机类专业“十二五”规划教材

C语言程序设计项目教程

主　编　陈翠红　黄　玲
副主编　陆金江　蔡传军　庄　彦
参　编　芮素文　胡　煜　吴　俊

国防工业出版社
·北京·

内容简介

本书详细介绍了C语言的基础知识。全书共分为12章，主要内容包括C语言概述，数据类型、运算符和表达式，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，函数，指针，结构体与共用体，位运算，文件等相关知识。全书以项目+案例的模式编写。以“学生成绩管理系统”项目贯穿始终，将该项目的子目标分解到各个章节，并在最后一章进行整合。本着理论以“够用为主”的原则，在每章中，将关联案例提炼到章节开始处，然后再给出相关知识链接。

本书内容丰富，重点突出，实用性较强，适合作为高职高专院校计算机相关专业教材，也可作为其他非专业本科生自学C语言教材。

图书在版编目(CIP)数据

C语言程序设计项目教程/陈翠红，黄玲主编. —北京：国防工业出版社，2012.7

高等职业教育计算机类专业“十二五”规划教材

ISBN 978-7-118-08112-1

Ⅰ.①C... Ⅱ.①陈...②黄... Ⅲ.①C语言－程序设计－高等职业教育－教材 Ⅳ.①TP312

中国版本图书馆CIP数据核字(2012)第108642号

※

国防工业出版社出版发行

（北京市海淀区紫竹院南路23号 邮政编码100048）

北京奥鑫印刷厂印刷

新华书店经售

*

开本787×1092 1/16 印张17¾ 字数435千字

2012年7月第1版第1次印刷 印数1—4000册 定价35.00元

（本书如有印装错误，我社负责调换）

国防书店：(010)88540777 发行邮购：(010)88540776

发行传真：(010)88540755 发行业务：(010)88540717

前　言

在众多的程序设计语言中，C 语言以其灵活性和实用性受到了广大计算机应用人员的喜爱。C 语言是既得到美国国家标准化协会（ANSI ）标准化，又得到工业界广泛支持的计算机语言之一，几乎任何一种机型、任何一种操作系统都支持 C 语言开发；C 语言在巩固其原有应用领域的同时，又在拓展新的应用领域，支持大型数据库开发和 Internet 应用。一旦掌握了 C 语言，就可以较为轻松地学习其他任何一种程序设计语言，为后续的面向对象程序设计 Java 程序设计、C#程序设计等程序设计语言，以及数据结构、操作系统、数据库编程、网站开发、单片机编程等课程的学习打下基础。

目前关于 C 语言方面的书籍和教程很多，但是适合高职教育的教材大多采用传统的编写方式，教师教学及学生自学普遍认为枯燥乏味。本书总结众多从事 C 语言教学的一线教师的教学经验，推出项目 + 案例教学的模式编写。

全书共分为 12 章，主要内容包括 C 语言概述，数据类型、运算符和表达式，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，函数，指针，结构体与共用体，位运算，文件等相关知识。全书以“学生成绩管理系统”项目贯穿始终，将该项目的子目标分解到各个章节，并在最后一章进行整合。本着理论以“够用为主”的原则，在每章中，将关联案例提炼到章节开始处，而后再给出相关知识链接。内容丰富，重点突出，实用性较强。

本书由陈翠红、黄玲任主编，负责全书的统稿、修改和定稿工作；由陆金江、蔡传军、庄彦任副主编；由芮素文参编。其中第 1、4 章由芮素文（安徽国防科技职业学院）编写，第 2、6 章由陆金江（安徽财贸职业学院）编写，第 3 章由庄彦（安徽工商职业学院）编写，第 5、7 章由陈翠红（安徽工商职业学院）编写，第 8、10 由黄玲（安徽工商职业学院）编写，第 9、11 章由蔡传军（安徽医学高等专科学校）编写，第 12 章由陈翠红、黄玲共同编写；另外，参加本书编写及检验等工作的还有胡煜（安徽国际商务职业学院），吴俊（安徽医学高等专科学校）等老师。

本书所有案例都已在真实环境中调试通过，读者可索取课件，电子邮箱：44189762@ qq. com 或 cch_ahbvc0801@ sina. com。也恳请广大读者对书中的错误及不足之处给予指正。

编者

目　录

[第一部分　项目提出]

[第二部分　项目分解]

[第三部分 项目整合]

第一部分　项目提出

第1章　C语言概述

【学习目标】

1. 了解C语言的形成和发展的过程。
2. 了解程序设计语言的概念。
3. 理解C语言的特点。
4. 熟悉C语言程序的运行步骤。
5. 熟练Turbo C和VC++6.0开发环境。

【项目介绍】

学生成绩管理系统利用计算机对学生成绩进行管理，可以进一步提高学校办学效益和现代化水平，提高广大教师的工作效率，实现学生成绩管理工作流程的系统化、规范化和自动化。本系统涉及程序设计各个方面的知识，通过该项目的实现使读者能对C语言有一个全面的认识，能够综合利用所学知识解决实际问题。

学生成绩管理系统包括信息的录入、显示、查找、插入、删除、修改、排序、统计、存盘等常规功能，如图1-1所示。

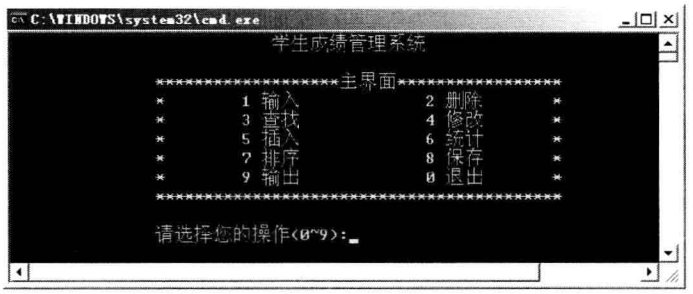

图1-1　学生成绩管理系统功能菜单

本书将本项目的各个功能目标分解到第二部分的每个章节，最后在第三部分提供详细的设计思路和解决方案，读者可以通过对C语言知识的系统学习，最终能独立完成本项目的所有功能。

1.1　C语言的发展历史及特点

计算机语言是进行程序设计的工具，故而也称为程序设计语言，可分为机器语言、汇编语言和高级语言。C语言是目前国际上广泛使用的一种计算机高级语言，适合编写系统软件，

也可以编写应用软件。

1.1.1 C 语言的发展历史

C 语言的发展过程与 UNIX 相辅相成，紧密地联系在一起。

早期的操作系统等系统软件主要是用汇编语言编写的，如 UNIX 操作系统。由于汇编语言依赖于计算机硬件，程序的可读性和可移植性都比较差。为了提高可读性和可移植性，最好改用高级语言，但一般高级语言难以实现汇编语言可以直接对硬件进行操作的功能，如对内存地址的操作、位(bit)操作等。人们设想能否找到一种既具有一般高级语言特性，又具有低级语言特性的语言，集它们的优点于一身。于是，C 语言就在这种情况下应运而生了。

C 语言是在 B 语言的基础上发展起来的，它的根源可以追溯到 ALGOL 60。1960 年出现的 ALGOL 60 是一种面向问题的高级语言，它离硬件比较远，不宜用来编写系统程序。1963 年英国的剑桥大学推出了 CPL(Combined Programming Language)。CPL 在 ALGOL 60 的基础上接近硬件一些，但规模比较大，难以实现。1967 年英国剑桥大学的 Matin Richards 对 CPL 作了简化，推出了 BCPL(Basic Combined Programming Language)。1970 年美国贝尔实验室的 Ken Thompson 以 BCPL 为基础，又作了进一步简化，它使得 BCPL 能挤压在 8KB 内存中运行，这个很简单的而且很接近硬件的语言就是 B 语言(取 BCPL 的第一个字母)，并用它写了第一个 UNIX 操作系统，在 DEC PDP-7 上实现。1971 年在 PDP-11/20 上实现了 B 语言，并写了 UNIX 操作系统。但 B 语言过于简单，功能有限，并且和 BCPL 都是“无类型”的语言。

1972 年至 1973 年间，贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计出了 C 语言(取 BCPL 的第二个字母)。C 语言既保持了 BCPL 和 B 语言的优点(精练、接近硬件)，又克服了它们的缺点 (过于简单、数据无类型等)。最初的 C 语言只是为描述和实现 UNIX 操作系统提供一种工具语言而设计的。1973 年，K. Thompson 和 D. M. Ritchie 两人合作把 UNIX 的 90% 以上用 C 语言改写，即 UNIX 第 5 版。原来的 UNIX 操作系统是 1969 年由美国的贝尔实验室的 K. Thompson 和 D. M. Ritchie 开发成功的，是用汇编语言写的，这样，UNIX 使分散的计算系统之间的大规模联网以及互联网成为可能。

C 语言发展过程如图 1-2 所示。

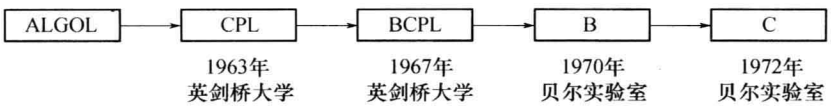

图 1-2　C 语言发展过程

后来，C 语言作了多次改进，但主要还是在贝尔实验室内部使用。直到 1975 年 UNIX 第 6 版公布后，C 语言的突出优点才引起人们普遍注意。1977 年出现了不依赖于具体机器的 C 语言编译文本《可移植 C 语言编译程序》，使 C 语言移植到其他机器时所需做的工作大大简化了，这也推动了 UNIX 操作系统迅速地在各种机器上实现。随着 UNIX 的日益广泛使用，C 语言也迅速得到推广。1978 年以后，C 语言已先后移植到大、中、小、微型机上，如 IBM System/370、Honeywell 6000 和 Interdata 8/32，已独立于 UNIX 和 PDP 了。以 1978 年由美国电话电报公司(AT&T)贝尔实验室正式发表的 UNIX 第 7 版中的 C 编译程序为基础，Brian W. Kernighan 和 Dennis M. Ritchie(里奇)合著了影响深远的名著《The C Programming Language》，常常称它为“K&R”，也有人称为“K&R 标准”或“白皮书”(white book)，它成

为后来广泛使用的C语言版本的基础。但在“K&R”中并没有定义一个完整的标准C语言。为此，1983年，美国国家标准化协会(ANSI)X3J11委员会根据C语言问世以来各种版本对C语言的发展和扩充，制定了新的标准，称为ANSI C，ANSI C比原来的标准C语言有了很大的发展，1987年，ANSI又公布了新标准——87 ANSI C。1988年K&R修改了他们的经典著作《The C Programming Language》，按照ANSI C标准重新写了该书。现在C语言已风靡全世界，成为世界上应用最广泛的几种计算机语言之一，后来的Java、C++、C#都是以C语言为基础发展起来的。

1.1.2 C语言的特点

一种语言之所以能存在和发展，并具有生命力，总是有其不同于(或优于)其他语言的特点。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，9种控制语句，程序书写形式自由，主要用小写字母表示，压缩了一切不必要的成分。

(2) 运算符丰富。C语言的运算符包含的范围很广泛，共有34种运算符。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。

(3) 数据结构丰富，具有现代化语言的各种数据结构。C语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。能用来实现各种复杂的数据结构(如链表、树、栈等)的运算。尤其是指针类型数据，使用起来比PASCAL更为灵活、多样。

(4) 具有结构化的控制语句(如if...else语句、while语句、do...while语句、switch语句、for语句)。用函数作为程序的模块单位，便于实现程序的模块化。C语言是良好的结构化语言，符合现代编程风格的要求。

(5) 语法限制不太严格，程序设计自由度大。例如对数组下标越界不做检查，由程序编写者自己保证程序的正确。对变量的类型使用比较灵活，例如整型数据与字符型数据可以通用。一般的高级语言语法检查比较严，能检查出几乎所有的语法错误。而C语言允许程序编写者有较大的自由度，因此，放宽了语法检查。程序员应当仔细检查程序，保证其正确，而不要过分依赖C语言编译程序去查错。“限制”与“灵活”是一对矛盾。限制严格，就失去灵活性;而强调灵活，就必然放松限制。一个不熟练的编程人员，编一个正确的C语言程序可能会比编一个其他高级语言程序难一些。也就是说，对用C语言的人，要求对程序设计更熟练一些。

(6) C语言能进行位(bit)操作，能实现汇编语言的大部分功能，可以直接对硬件进行操作。因此C语言既具有高级语言的功能，又具有低级语言的许多功能，可用来写系统软件。C语言的这种双重性，使它既是成功的系统描述语言，又是通用的程序设计语言。

有人把C语言称为“高级语言中的低级语言” 或“中级语言”，意为兼有高级和低级语言的特点。一般仍习惯将C语言称为高级语言，因为C程序也要通过编译、连接才能得到可执行的目标程序，这是和其他高级语言相同的。

C语言的以上特点，读者现在也许还不能深刻理解，待学完C语言以后再回顾一下，就会有比较深的体会。总之，C语言对程序员要求较高。程序员使用C语言编写程序会感

到限制少，灵活性大，功能强，可以编写出任何类型的程序。学习和使用 C 语言的人已越来越多。

1.2 C 程序的运行

计算机只能执行机器语言，C 语言是一种高级程序设计语言，而任何高级语言源程序都要“翻译”成机器语言，才能在机器上运行。“翻译”的方式有两种：一种是解释方式，即对源程序解释一句执行一句;另一种是编译方式，即先把源程序“翻译”成目标程序(用机器代码组成的程序)，再经过连接装配后生成可执行文件，最后执行可执行文件而得到结果。

C 语言是一种编译型的程序设计语言，它采用编译的方式将源程序翻译成目标程序(机器代码)。运行一个 C 程序，从输入源程序开始，要经过编辑源程序文件(.c)、编译生成目标文件(.obj)、连接生成可执行文件(.exe)和执行四个步骤。

1.2.1 C 程序的结构

示例：

```
/* 第一个 C 程序*/                    /*注释*/
#include “stdio.h”                    /*编译预处理命令*/
void main(void)                       /*函数*/
{
    printf( “Hello. \n” );            /*语句*/
}
```

运行结果如图 1-3 所示。

图 1-3 示例运行结果

从此示例可以看出，C 语言程序的结构大体可分为以下几个组成部分。

1. 函数

函数是程序的基本单位，是程序的主体，程序由一个或多个函数组成，但是每个 C 程序有且仅有一个主函数 main()。主函数的位置可以在程序的任意位置，它是程序的入口，程序执行总是从 main 开始，在 main 中结束，其它函数通过调用得以执行。

C 函数由函数名、形式参数和函数体三部分组成，其格式如下：

返回值类型 函数名（[形式参数]）

```
{
    函数体                            /*变量定义语句、实现函数功能的语句等*/
}
```

例如：

```
int Add(int x,int y)                  /*函数的首部*/
```

```
{                                          /* “{” 代表函数体开始*/
   int sum;                                /*变量定义语句*/
   sum=x+y;                                /*具体实现函数功能的语句*/
   return(sum);
}.                                         /* “}” 代表函数体结束*/
```

2. 语句

语句是组成程序的基本单元，以分号（;）作为结束标志。C 程序的书写格式较为自由，一条语句可以写在多行上，一行上也可以写若干条语句。C 程序区分大小写，C 语言的基本语句都是用小写字母表示的。

3. 编译预处理命令

每个以符号“#”开头的行，称为编译预处理行，是 C 语言提供的一种模块工具。

4. 注释

用/*……*/括起的内容，可对程序进行注释，其作用是给程序设计者一种提示或记号。注释使用时不能嵌套，且注释内容不参加程序的执行，主要作用是提高程序的可读性，故而不产生编译代码。

1.2.2 C 程序的运行步骤

C 语言程序的运行步骤包括以下过程。

(1) 上机输入或修改已有的源程序（.c 文件），称为编辑。

(2) 通过“编译程序”对源程序进行编译（包括预处理），将其翻译成计算机能识别和执行的二进制指令形式的“目标程序”(.obj 文件)，称为编译。

(3) 将目标程序与系统函数库以及其他目标程序连接起来，形成可执行的目标程序(.exe 文件)称为连接。

(4) 运行可执行的目标程序得到结果，称为运行。

对得到的运行结果需要进行验证，如果结果不正确，则需重新调试程序。C 程序的运行步骤可用图 1-4 表示。

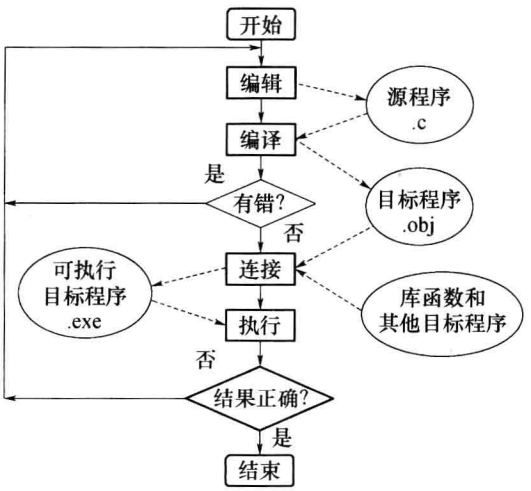

图 1-4 C 语言程序运行步骤