



规划教材

普通高等学校应用型人才培养系列规划教材
丛书主编 陈明

案例式C语言教程

ANLISHI C YUYAN JIAOCHENG

陈 慧 马杰良 主 编
何 军 王春红 副主编

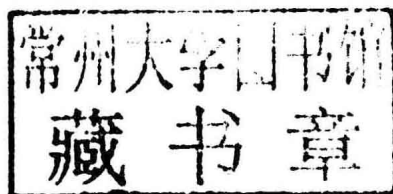
中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



普通高等学校应用型人才培养系列规划教材
丛书主编 陈明

案例式 C 语言教程

主 编 陈 慧 马杰良
副主编 何 军 王春红



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书主要是配合高等院校信息类相关专业“C 语言程序设计”课程理论教学而编写的。

本书内容共分 12 章, 采用从案例出发逐步阐述的形式, 较为系统和全面地讲解了 C 语言程序设计中主要的基础知识、常用语法规则和程序设计技巧, 并设计了大量实例。本书按读者认知规律进行编写, 内容丰富、形式新颖、结构合理、图文并茂, 编程与理论结合紧密。

本书适合作为高等院校相关专业学生的 C 语言程序设计教材, 也可供其他相关技术人员参考。

图书在版编目 (CIP) 数据

案例式 C 语言教程 / 陈慧, 马杰良主编. — 北京 :
中国铁道出版社, 2011. 2
普通高等学校应用型人才培养系列规划教材
ISBN 978-7-113-12175-4

I. ①案… II. ①陈… ②马… III. ①
C 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 (2011) 第 008002 号

书 名: 案例式 C 语言教程
作 者: 陈 慧 马杰良 主编

策划编辑: 杨 勇
责任编辑: 秦绪好
编辑助理: 陈 文
封面设计: 付 巍
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: 北京市兴顺印刷厂
版 次: 2011 年 2 月第 1 版 2011 年 2 月第 1 次印刷
开 本: 787mm×960mm 1/16 印张: 19.5 字数: 401 千
印 数: 3 000 册
书 号: ISBN 978-7-113-12175-4
定 价: 29.50 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社计算机图书批销部联系调换。

经过几十年的研究与探索，现代计算机系统功能越来越强大、应用越来越广泛。计算机的广泛应用一方面对人类社会的发展做出了卓越的贡献，另一方面也在推动计算机学科的高速发展，因而一直受到社会的高度关注。

由于计算机学科呈现出的学科内涵宽泛化、分支相对独立化、社会需求多样化、专业规模巨大化和计算教育大众化等特点，使得计算机企业成为了朝阳企业，这些IT企业需要大量的具有专门计算机技能的人才，但传统的研究型计算机教育是以学术教育为基础，以培养计算机精英为目的的计算机教育，与社会和行业对计算机高等教育人才的需求产生了矛盾——大学中单一的计算机精英型教育培养的人才已不能满足实际需要，凸显职业特征的计算机应用型教育异军突起，迅速发展，备受关注。这种矛盾促使教学模式呈现了职业性，并在培养面向知识应用和全面能力方面，提出了多种职业性教学模式，如网络工程师、软件工程师、动画设计师、硬件工程师等培养模式。因此，研究和实施计算机应用型人才培养模式势在必行。

什么是计算机应用型教育？我们通过长时间的教育研究和教学经验的总结，认为计算机应用型教育的培养目标可以利用知识、能力和素质三个基本要素来描述。

知识是基础、载体和表现形式，从根本上影响着能力和素质。学习知识的目的是为了获得能力和不断地提升能力。知识可以转化为能力和素质，能力对知识具有反作用，促进知识的不断发展。

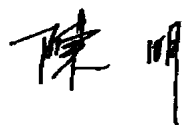
能力是核心，是应用型人才特征的突出表现。从计算机学科而言，培养的人才应具备计算思维能力、算法设计与分析能力、程序设计与实现能力、系统能力（系统的认知、设计、开发、应用能力）。而计算机应用型人才的能力有着其独特的属性，主要包括应用能力（专业能力）和通用能力。应用能力主要是指用所学知识解决专业实际问题的能力。通用能力是指跨职业能力，并不是具体的专业能力和职业技能，而是对不同职业的适应能力，也就是当职业发生变更时，这些能力依然在从业者身上起作用。计算机应用型本科人才所应具备的三种通用能力是：学习能力、工作能力和创新能力。

素质要素主要是指基本素质，即具有良好的公民道德和职业道德，具有合格的政治思想素养，遵守计算机法规和法律，具有人文、科学素养和良好的职业素质等。计算机应用型人才素质主要是指工作的基本素质，且要求在从业中必须具备责任意识，能够对自己职责范围内的工作认真负责地完成。

与此同时，我们认为计算机应用型教育培养目标的实现关键在于课程体系的构建，即课程内容和课程性质的确定。因此，我们将计算机应用型教育课程分为通用课程、基础课程、专业核心课程、专业选修课程、应用课程、实验课程、实践课程等，并建立了相应的教育课程体系，如公共基础课程平台、专业基础课程平台、专业选修课程平台、基本素质平台等，力图通过相应的课程开展来实现培养目标。

目前，应用型人才培养的研究方兴未艾，我们也将会在较长一段实践教学工作中继续探讨和总结经验。此次组织的这套“普通高等学校应用型人才系列规划教材”系列丛书是应社会需求而建设，经过系统规划与设计，定位于高等院校计算机应用型人才的培养，整套丛书从架构和具体教材的编写上都力主突出能力培养的理念。本系列教材现正在陆续出版中，希望各位老师和读者指正。

《普通高等学校应用型人才系列规划教材》丛书主编

A handwritten signature in black ink, consisting of the characters '陈明' (Chen Ming) in a stylized, cursive script.

“C 语言程序设计”是电气信息类相关专业一门重要的专业基础课程。C 语言是一种理想的结构化程序设计语言，它具有功能丰富、使用灵活方便、应用面广等特点，也是学习 C++ 语言和其他程序设计语言的基础。因此，C 语言已成为目前的主流程序设计语言之一，并逐渐成为其他相关专业的热门选修课程甚至专业基础课程。

“C 语言程序设计”课程具有理论与实践并重的特征。因此，学好这门课程不仅要掌握理论知识，而且还得具备运用理论知识处理实际问题的技能。C 语言程序设计学习过程中涉及的知识点和语法规则比较多，给大多数普通高校的学生学习这门课程带来困难。基于以上认识，我们编写了本书，旨在帮助学生更好地、系统性地学习 C 语言。目前，大部分现有的 C 语言学习教程主要是由介绍理论知识开始，然后给出例题。本书按读者认知规律进行编写，对每章知识点的讲解是从案例出发的。通过给出案例，让读者对问题应用场景有初步认识，然后逐步引出相关知识点。

本书的编写主要面向电气信息类相关专业的理工科学生，同时也可以作为计算机专业学生学习 C 语言程序设计的教材。为了满足大多数学生的学习需要，本书采用从案例出发逐步阐述的形式，较为系统和全面地讲解了 C 语言程序设计中主要的基础知识、常用语法规则和程序设计技巧，并设计了大量实例。

全书内容共分为 12 章。第 1 章是概述，介绍 C 语言的产生背景、特点、运行环境以及算法的相关知识。第 2 章介绍 C 语言的基本数据类型。第 3 章至第 5 章分别介绍结构化程序设计的三种基本结构：顺序结构、选择结构和循环结构程序设计。第 6 章介绍数组。第 7 章介绍函数。第 8 章介绍编译预处理。第 9 章介绍指针。第 10 章介绍两种构造类型：结构与联合。第 11 章介绍文件操作。第 12 章介绍程序调试方法。对每章中的例题，我们都给出了参考程序，在 VC++6.0 环境下运行实现并给出了运行结果，读者可以通过阅读参考程序和对应的运行结果检验对相关知识点的掌握程度。

本书由南京信息工程大学电子与信息工程学院及兄弟院校的教师结合多年的教学和科研实践经验编写而成。其中第 1、7 章由马杰良老师编写，第 4、5、6、8 章由陈慧老师编写，第 9、10、11、12 章由何军老师编写，第 2、3 章由王春红老师编写。全书由陈慧、马杰良两位老师进行统稿。在本书的编写过程中，参考了国内许多正式或非正式出版的相关著作，在此向这些著作的作者们致以衷心的感谢！

希望本书能够给学生进一步学好“C 语言程序设计”课程带来启迪和帮助。由于作者水平有限，加之编写时间仓促，书中难免存在不足和错误之处，敬请广大读者批评指教！

编 者

2010 年 12 月 于南京

第 1 章 概述.....	1
1.1 计算机程序设计语言的发展	1
1.1.1 机器语言和汇编语言	1
1.1.2 高级语言	2
1.2 C 语言简介	3
1.2.1 C 语言的产生与发展	3
1.2.2 C 语言的作用	4
1.2.3 C 语言的特点	4
1.3 如何学习 C 语言程序设计	5
1.3.1 认识程序设计	5
1.3.2 学习 C 语言程序设计的有效方法	6
1.3.3 创建 C 语言程序的一般步骤	6
1.4 C 语言程序的基本结构及书写规范	8
1.4.1 C 语言程序的基本结构	8
1.4.2 一个简单的 C 程序	9
1.4.3 C 语言的程序书写规范	12
1.5 Visual C++集成编译环境介绍	13
1.5.1 Visual C++集成编译环境的安装	13
1.5.2 Visual C++集成编译环境的简单使用	15
1.6 算法及算法表示	19
1.6.1 算法与程序设计	19
1.6.2 算法的表示	20
习题	24
第 2 章 基本数据类型	25
2.1 C 语言的字符集与词汇	25
2.1.1 C 语言的字符集	25
2.1.2 C 语言的词汇	26

2.1.3	信息的计算机存储	29
2.1.4	什么是常量和变量	30
2.1.5	赋值操作	31
2.2	C 语言中的基本数据类型	31
2.2.1	理解数据类型	31
2.2.2	整数类型	32
2.2.3	浮点类型	38
2.2.4	字符类型	40
2.2.5	字符串类型	41
2.2.6	枚举类型	42
2.3	混合运算及强制类型转换	43
2.3.1	基本运算符	43
2.3.2	优先级和结合性	44
2.3.3	类型转换	53
2.3.4	极限值	55
	习题	56
第 3 章	顺序结构程序设计	60
3.1	C 语句简介	60
3.1.1	表达式语句	61
3.1.2	赋值语句	61
3.1.3	函数调用语句	62
3.1.4	空语句	62
3.1.5	控制语句	62
3.1.6	复合语句	63
3.2	基本的信息输出函数	63
3.2.1	printf() 函数	63
3.2.2	字符输出函数	69
3.3	数据输入语句	70
3.3.1	scanf() 函数	70
3.3.2	键盘输入函数	75
3.4	关系运算和逻辑运算	76
3.4.1	C 语言中的关系运算符	77
3.4.2	关系表达式	77

3.4.3 逻辑运算符和表达式	78
3.5 顺序结构程序设计实例	80
习题	83
第 4 章 选择结构程序设计	87
4.1 用 if 语句实现选择结构	87
4.1.1 条件语句	88
4.1.2 条件运算符	91
4.2 用 switch 语句实现多分支选择结构	93
4.3 选择结构的嵌套	96
4.3.1 if 语句的嵌套	96
4.3.2 switch 语句的嵌套	97
习题	99
第 5 章 循环结构程序设计	100
5.1 for 循环结构	100
5.2 while 和 do...while 循环结构	103
5.2.1 while 语句实现循环	103
5.2.2 do...while 语句实现循环	104
5.3 continue 语句和 break 语句	105
5.3.1 continue 语句	105
5.3.2 break 语句	107
5.4 循环的嵌套	109
习题	112
第 6 章 数组	113
6.1 一维数组和多维数组的定义、引用和初始化	113
6.1.1 一维数组	116
6.1.2 多维数组	125
6.2 字符串与字符数组	129
6.2.1 字符数组的定义、引用和初始化	129
6.2.2 字符串和字符串的结束标志	132
6.2.3 字符串的输入和输出	133
6.2.4 字符串处理函数	138
习题	141

第 7 章 C 语言中的函数	142
7.1 为什么要使用函数	142
7.1.1 规划程序	143
7.1.2 自顶向下的程序设计方法	144
7.1.3 C 语言中的函数	145
7.2 定义函数	147
7.2.1 无参函数的定义	148
7.2.2 有参函数的定义	149
7.3 函数的调用	150
7.3.1 函数的调用	150
7.3.2 函数的递归调用	154
7.4 函数的参数	156
7.4.1 函数的参数	156
7.4.2 函数调用中的参数传递形式	158
7.4.3 数组在函数间的传递	165
7.5 函数的值	166
7.6 C 语言变量的作用域和存储类型	169
7.6.1 变量的作用域和生存期	169
7.6.2 变量存储类型	177
习题	185
第 8 章 编译预处理	191
8.1 宏定义	191
8.1.1 不带参数的宏定义	192
8.1.2 带参数的宏定义	197
8.2 文件包含	198
8.3 条件编译	200
习题	204
第 9 章 指针	205
9.1 指针与地址	206
9.1.1 指针变量	207
9.1.2 取地址运算符与间接寻址运算符	209
9.1.3 指针与函数	210
9.2 指针与数组	214

9.2.1	指针的算术运算.....	216
9.2.2	指针与一维数组.....	219
9.2.3	指针与多维数组.....	221
9.3	特殊的指针	223
9.3.1	指向字符串的指针.....	225
9.3.2	指向函数的指针.....	227
9.3.3	指向指针的指针.....	229
9.4	动态内存分配	230
9.4.1	C 语言内存分配函数	232
9.4.2	动态分配数组和字符串	235
9.4.3	释放存储	238
习题		239
第 10 章	结构与联合	241
10.1	结构	242
10.1.1	结构的定义	244
10.1.2	结构数组	247
10.1.3	结构指针	248
10.1.4	结构类型作为函数参数	254
10.1.5	typedef 类型定义符	256
10.2	联合	257
10.2.1	联合的定义	257
10.2.2	结构与联合的区别.....	259
习题		260
第 11 章	文件.....	261
11.1	文件概述	261
11.2	文件的分类	264
11.3	C 语言文件操作相关函数	265
11.3.1	文件指针	270
11.3.2	文件的打开与关闭.....	270
11.3.3	文件的读/写	272
11.3.4	文件检测函数	278
习题		279

第 12 章 程序调试方法	280
12.1 程序调试基本步骤	280
12.2 程序调试方法	286
12.2.1 开始调试与停止调试	286
12.2.2 单步执行	287
12.2.3 观察变量	290
12.2.4 查看内存与寄存器等	290
12.2.5 运行栈切换	291
附录 A 常用 ASCII 码对照表	293
参考文献	298

第

1

章

概 述

学习目标

通过本章的学习,使读者能够了解 C 语言的产生和发展,并对 C 语言的特点有初步的认识。通过引入实例初步了解 C 语言的运行环境以及运行 C 程序的具体过程;了解与 C 语言程序设计密切相关的算法的概念及其表示方法。

学习要求

- 了解 C 语言的特点和作用。
- 了解 C、Visual C++ 编译器的安装和基本操作。
- 了解 C 语言算法的概念及表示方法。

本章主要介绍 C 语言的特点和作用、学习 C 语言程序设计的有效途径以及 Visual C++ 编译器的安装和基本操作。

1.1 计算机程序设计语言的发展

语言是人类交流思想和情感的工具,当今世界很多民族都有自己的语言。每种语言都有自己的语法规则和词法规则并形成一个系统。

计算机程序设计语言是人类根据计算机硬件的特征所设计的,计算机可以识别的语言,用于描述人们希望解决问题的方法和步骤,供计算机阅读和执行。

1.1.1 机器语言和汇编语言

利用人造的工具代替人们完成计算工作,是人们一直的梦想。20 世纪,在强大的需求背景下,

由于技术和理论的突破，人们在 1946 年 2 月终于制作出人类历史上第一台数字电子计算机——ENIAC。这台由电子管组成的庞然大物，每秒可进行 5 000 次加法运算，标志着现代计算机的诞生。从计算机产生到目前短短的 60 多年时间中，计算机科学与技术得到了迅猛的发展，计算机已经成为信息化社会必不可少的工具。

计算机是由电子器件组成的，计算机的工作是通过电子器件之间的协调和信号交换来实现的。一个完整的计算系统机应该由硬件和软件组成，软件包含了计算机在解决问题的过程中所需要的各种程序和文档。计算机的所有工作都是在其存储器内部的程序控制下完成的，这些存储器内的程序是用户根据需求事先存入的，离开程序，计算机将一事无成。程序是指令的集合，计算机硬件所能识别的指令形式，只能是简单的“0”（低电平）和“1”（高电平）的组合。一台计算机的硬件系统所能识别的所有指令的集合，称为它的指令系统。这个指令系统是由计算机硬件系统的结构和设计者的思想所确定的。

由计算机硬件系统可识别的二进制指令所组成的语言称为机器语言。显然，机器语言便于计算机识别，但对于人类来说却是晦涩难懂，更难于记忆。在计算机发展的初期，软件工程师只能用机器语言来编写程序。这一阶段，在人类的自然语言和计算机语言之间存在着巨大的差距，软件开发的难度大、周期长、软件能实现的功能简单，界面也不友好。

汇编语言是将不同机器的指令映射为自然语言中不同的助记符，例如 ADD、SUB、MOV 等。此时，汇编语言与人类自然语言之间的差距略有缩小，但仍与人类的思维相差很远，以为用汇编语言写程序仍然要考虑很多的机器细节，因此，这时的程序设计仍然局限在专业的程序员范围内。

从机器语言到汇编语言是一大进步，这表明人与计算机的硬件系统不必使用同一种语言。程序员可以使用较适合人类思维习惯的语言，而计算机的硬件系统仍然识别机器指令。那么，两种语言之间的沟通如何实现呢？这时人们就开发出了一种翻译软件。从汇编语言到计算机指令之间的翻译软件称为汇编程序，它可以将程序员写的汇编助记符直接转化为机器指令，然后再由计算机去识别和执行。

1.1.2 高级语言

高级语言是相对机器语言和汇编语言而言的一类大众化或专业化计算机程序设计语言。高级语言的出现是计算机程序设计语言的一大进步。它屏蔽了计算机系统中硬件和操作系统的细节，提高了语言的抽象层次，程序中可以采用不同的数据类型来表示程序处理的对象，用语句来实现程序控制和数据操作。20 世纪 60 年代末出现的结构化程序设计语言进一步提高了语言的抽象层次。结构化的数据、结构化的语句、数据抽象、过程抽象使程序更便于体现客观事物的结构和逻辑含义，这使得编程语言与人类的自然语言更接近。C 语言就是这样的一种结构化程序设计语言。

1.2 C 语言简介

1.2.1 C 语言的产生与发展

早期的操作系统软件主要是用汇编语言编写的。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。改用高级语言编写系统软件，可以提高其可读性和可移植性，但一般的高级语言难以实现汇编语言的某些功能（汇编语言可以直接对硬件进行操作，例如，对内存地址的操作、位操作等）。因此，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。C 语言就在这种情况下应运而生的。

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（也就是说不能编写程序直接对计算机的硬件进行控制），因而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）语言。CPL 语言在 ALGOL 60 的基础上更接近了硬件一些，但规模比较大，难以实现。

1967 年英国剑桥大学的 Martin Richards 对 CPL 语言做了简化，推出了 BCPL（Basic Combined Programming Language）语言。1970 年，美国贝尔实验室的 Ken Thompson 以 BCPL 语言为基础，又做了进一步简化，设计出很简单且很接近硬件的 B 语言（取 BCPL 的第一个字母），并用 B 语言写出 UNIX 操作系统，在 PDP-7 计算机上实现。1971 年，在 PDP-11/20 计算机上实现了 B 语言，并写出了 UNIX 操作系统。

1972 年—1973 年，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、无数据类型等）。最初的 C 语言只是为描述和实现 UNIX 操作系统提供的一种工作语言。1973 年，K.Thompson 和 D.M.Ritchie 两人合作把 UNIX 的 90% 以上代码用 C 改写（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），实现了 C 语言的现实应用。

后来，C 语言多次进行改进，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的普遍注意。1977 年，出现了不依赖于具体机器的 C 语言编译版本，使 C 程序移植到其他机器时所做的工作大大得到简化，这也推动了 UNIX 操作系统迅速地在各种机器上实现。例如，VAX、AT&T 等计算机系统都相继开发了 UNIX。随着 UNIX 的日益广泛使用，C 语言也迅速得到推广。C 语言和 UNIX 可以说是一对孪生兄弟，在发展过程中相辅相成。1978 年以后，C 语言已先后移植到大、中、小、微型机上，已独立于 UNIX 和 PDP 计算机。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。

Brian W.Kernighan 和 Dennis M.Ritchie（合称 K&R）以 1978 年发表的 UNIX 第 7 版中的 C 编译

程序为基础，合著了影响深远的名著 *The C Programming Language*，本书介绍的 C 语言成为后来被广泛使用的 C 语言的基础，也被称为标准 C。1983 年，美国国家标准化协会（ANSI）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 有了很大的发展。K&R 在 1988 年修改了他们的经典著作 *The C Programming Language*，按照 ANSI C 的标准重新写了该书。1987 年，ANSI C 又公布了新标准——87 ANSI C。目前流行的 C 编译系统都是以它为基础的。

1.2.2 C 语言的作用

C 语言既有高级语言的特点，又具有汇编语言的特点；既是一种成功的系统设计语言，又是一种实用的程序设计语言；既能用来编写不依赖计算机硬件的应用程序，又能用来编写各种系统程序；是一种受欢迎、应用广泛的程序设计语言。“C 语言程序设计”是现今几乎所有高校理工类专业学生都需要学习的一门基础课程，这说明大家有一种共识，通过 C 语言的学习，可以锻炼学生的程序设计能力，培养学生程序设计的技能。在 IT 行业几乎所有的专业程序员都应该是 C 语言程序设计好手，能用 C 语言开发系统程序在 IT 行业具有很强的竞争力。

1.2.3 C 语言的特点

1. 简洁紧凑、灵活方便

C 语言一共只有 32 个关键字，9 种控制语句，程序书写自由，主要用小写字母表示。它把高级语言的基本结构和语句与低级语言的实用性相结合，使得 C 语言可以像汇编语言一样对位、字节和地址进行操作，而这三者是计算机最基本的工作单元。

2. 运算符丰富

C 语言的运算符包含的范围很广泛，共有种 34 个运算符。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 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其他高级语言中难以实现的运算。

3. 数据结构丰富

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类型；引入了指针概念，使程序效率更高。另外，C 语言具有强大的图形功能，支持多种显示器和驱动器，且计算功能、逻辑判断功能强大。

4. C 语言是结构式语言

结构式语言的显著特点是代码及数据的分隔化，即程序的各个部分除了必要的信息交流外彼此