



普通高等教育“十一五”国家级规划教材

教育部“高等学校教学质量与教学改革工程”立项项目

甘 勇 主 编  
金保华 闫红岩 副主编

# C语言程序设计基础 及进阶教程

计算机科学与技术专业实践系列教材

清华大学出版社



普通高等教育“十一五”国家级规划教材

计算机科学与技术专业实践系列教材

# C语言程序设计基础 及进阶教程

清华大学出版社  
北京

## 内 容 简 介

本书分上、中、下三篇共 12 章,循序渐进地讲述了 C 语言程序设计的编程技术,从基本概念的理解到常用算法的介绍,再到具体案例的应用,其中还穿插讲述了一些程序设计测试和调试方法。各部分内容具体分布如下。

第 1 章~第 9 章是基础篇,依次讲解了 C 语言基础知识、三大基本控制结构、数组、函数、指针、结构体、共用体、文件与低层接口等主要内容。

第 10 章为常用算法篇。介绍了计算机程序设计的基本方法(包括筛选法、迭代法、枚举、模拟、递推、分治与递归、贪心算法)和优化设计思想,通过实例说明了常用程序设计方法的实际应用和编程技巧。

第 11 章~第 12 章为综合案例篇,从系统需求、总体设计、函数设计、编码和测试进行了讲解。图书管理系统应用了基础阶段知识,工资管理系统应用了高级编程技术(汉字、图像、图形、动画、密码验证、鼠标和键盘操作、系统中断调用等)。案例之后有可增加功能、关键技术剖析和导师点评。

本书通俗易懂,便于读者自学;范例选取精心,代码规范,具有典型的代表性,可移植性强,便于读者创新思维。

本书主要适用于高等学校计算机及相关专业的教材,也可供自学者及软件开发人员参考。

为了方便读者的学习和教师教学,本书配有电子教案,并提供书中的实例和案例源代码。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

C 语言程序设计基础及进阶教程/甘勇主编. —北京:清华大学出版社,2010.9

(计算机科学与技术专业实践系列教材)

ISBN 978-7-302-23788-4

I. ①C… II. ①甘… 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(2010)第 171830 号

责任编辑:汪汉友

责任校对:时翠兰

责任印制:孟凡玉

出版发行:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185×260

印 张:21.5

字 数:519 千字

版 次:2010 年 9 月第 1 版

印 次:2010 年 9 月第 1 次印刷

印 数:1~4000

定 价:33.00 元

产品编号:039738-01

普通高等教育“十一五”国家级规划教材  
计算机科学与技术专业实践系列教材

## 编 委 会

主 任：王志英

副 主 任：汤志忠

编 委 委 员：陈向群 樊晓桢 邝 坚

孙吉贵 吴 跃 张 莉

# 前 言

C 系列语言是当今最流行的语言,C 语言是 C 系列语言的核心,是学习 C 系列语言的起点,同时也是了解和掌握计算机程序最基本的构成和运行特点最简洁的语言。C 语言具有表达能力强、功能丰富、可移植性好、使用灵活等优点,它既具备高级语言的特性,又具有直接操纵计算机硬件的能力,可用于系统软件和应用软件的开发,被国内外程序员广泛使用,已经成为高等院校计算机专业及相关专业的重要基础课,是学习程序设计的首选语言。

本书注重基础,突出应用,以实例贯穿全文,以编程应用为驱动,以通俗的语言和简要的内容阐述了 C 语言的程序设计思想和方法。全书分上、中、下三篇共 12 章,循序渐进地从基本概念到具体实践讲述 C 语言,并对具体知识点通过实例进行详细的讲解。其中第 1 章~第 9 章是基础篇:第 1 章介绍 C 语言的发展与特点、C 程序的组成及 C 程序的开发步骤;第 2 章讲解了 C 语言的数据类型、运算符和表达式;第 3 章通过简单程序设计,介绍了 C 语言的标准输入输出函数;第 4 章介绍了 C 语言的三种流程控制:顺序结构、选择结构和循环结构,并通过实例介绍了在循环结构中常用的穷举和迭代算法;第 5 章介绍了一维数组、二维数组、字符数组和字符串数组;第 6 章介绍了函数、变量与存储类型、编译预处理;第 7 章介绍了指针的概念,并举例说明指针在编程中的应用;第 8 章介绍了结构体、共用体、枚举类型的概念,重点讲解了链表的操作;第 9 章介绍了对文件操作。第 10 章为常用算法篇:介绍了计算机程序设计的基本方法:筛选法、迭代法、枚举、模拟、递推、分治与递归、贪心算法,通过实例说明了常用程序设计方法的实际应用和编程技巧。第 11 章~第 12 章是综合案例篇:第 11 章使用链表和文件等技术开发了图书管理系统;第 12 章使用了一些高级编程阶段的知识与技术开发了工资管理系统。

本书具有如下特点。

## 1. 注重基础内容讲解,突出实用性

本书不仅通俗地介绍了 C 语言中各个概念,而且在每个知识点后都配套有实例讲解;范例选取精心、代码规范,具有典型的代表性,可移植性强。

## 2. 强化编程思想和方法,突出应用性

书中所有范例都强调编程思想,在常用算法篇介绍了常见的程序设计方法,有意识引导学生提高编程能力和方法。并给出 C 程序的单文件和多文件编程的结构模式,培养学生在实际项目中的模块化程序设计意识。

## 3. 灌输软件工程思想,认知项目开发过程

下篇提供了两个综合案例——图书管理系统和工资管理系统。从系统需求分析、总体设计、函数设计、编码和运行测试进行了详细的讲解。其中,第一个系统应用了基础阶段的知识;第二个系统应用了高级编程技术,如汉字显示、图像显示、图形绘制、动画显示、密码验证、鼠标和键盘操作、系统中断调用等。前者适合基础阶段的实训或课程设计,后者适合 C 语言进阶(高级)阶段的实训或课程设计的范例。

通过案例引导学生巩固所学过的知识和技术,领略软件开发的步骤和方法,享受程序员

在开发过程的智慧与成就,体会学精一门语言的重要和不易,体验 C 语言的强大、高级编程技术的魅力和应用,从而达到提高学生的动手编程、刻苦钻研、创新思维的能力和热爱科学、团结协作的精神。

#### 4. 学练结合,巩固提高

在每章的后面都配备了一定量的习题和思考题,并有配套的实践教程,让学生进一步得到锻炼和提高。

另外,附录 A 给出了常用字符与 ASCII 码对照表,附录 B 提供了 ANSI C 关键字,附录 C 提供了 ANSI C 运算符的优先级与结合性,附录 D 提供了 ANSI C 中常用的标准库函数,附录 E 是常用的键盘按键的扫描码,附录 F 是常用的 DOS 和 BIOS 系统中断调用。

考虑到目前 Turbo C++ 3.0 仍然是 C 语言主流的编译器,非常适合于教学和学习,各类计算机考试也都以 Turbo C++ 3.0 为平台,故本书以 Turbo C++ 3.0 为基本平台。

本书由郑州轻工业学院甘勇教授主编,金保华和闫红岩为副主编并统稿主审,参加本书编写的人员还有韩丽、王捷、卢冰、李晔、钱慎一等。另外,张旭、何樱等同志参加了本书习题编写、程序调试、附录整理和题库研发等工作。

在此感谢几位作者鼎力相助,分享自己多年的教学经验,使更多有志于学好 C 语言的同学可以尽快地入门,并掌握好 C 语言这门计算机语言。

同时,感谢本书创作团队所在单位——郑州轻工业学院和计算机与通信工程学院的鼎力支持!

特别感谢清华大学出版社为本书所做的大量的策划和编辑工作!

由于编者水平有限,书中难免存在错误和不足之处,敬请读者和同行批评指正。作者 E-mail 地址: smxyhy@zzuli.edu.cn。

编者

于郑州轻工业学院计算机与通信工程学院

2010年9月

# 目 录

## 上篇 基础篇

第 1 章 C 语言概述 .....	3
1.1 C 语言的发展及特点 .....	3
1.1.1 程序设计语言 .....	3
1.1.2 C 语言的发展 .....	4
1.1.3 C 语言的特点 .....	5
1.2 C 程序的组成 .....	5
1.3 C 语言的基本语法单位 .....	8
1.4 C 程序的开发步骤 .....	9
习题 .....	10
第 2 章 数据类型、运算符和表达式 .....	11
2.1 数据类型 .....	11
2.1.1 C 的数据类型 .....	11
2.1.2 数值型数据的表示与存储形式 .....	12
2.1.3 C 字符型数据的表示和存储形式 .....	12
2.1.4 基本类型的名字和长度 .....	12
2.2 常量和变量 .....	14
2.2.1 常量的表示 .....	14
2.2.2 符号常量 .....	19
2.2.3 变量说明 .....	20
2.3 运算符和表达式 .....	21
2.3.1 算术运算 .....	22
2.3.2 关系运算 .....	23
2.3.3 逻辑运算 .....	24
2.3.4 自增自减 .....	25
2.3.5 位运算 .....	26
2.3.6 赋值运算 .....	28
2.3.7 条件运算 .....	30
2.3.8 顺序求值运算 .....	31
2.4 数据类型转换 .....	32
2.4.1 类型转换的规则 .....	32
2.4.2 类型转换的方法 .....	33
习题 .....	35

<b>第 3 章 简单的 C 程序设计</b> .....	38
3.1 C 语句 .....	38
3.1.1 C 语句分类 .....	38
3.1.2 表达式语句 .....	38
3.1.3 C 的语句概述 .....	39
3.2 标准的文件输入输出函数 .....	39
3.2.1 字符输入输出函数 .....	39
3.2.2 格式化输出函数 .....	40
3.2.3 格式化输入函数 .....	43
3.3 简单程序设计举例 .....	49
习题 .....	50
<b>第 4 章 C 语言流程控制</b> .....	52
4.1 复合语句 .....	52
4.2 选择结构 .....	53
4.2.1 if 语句的嵌套 .....	56
4.2.2 switch 语句 .....	57
4.3 循环结构 .....	59
4.3.1 while 语句 .....	59
4.3.2 do...while 语句 .....	62
4.3.3 for 语句 .....	63
4.3.4 多重循环 .....	65
4.3.5 流程转移语句 .....	67
4.3.6 循环结构程序设计应用 .....	71
习题 .....	75
<b>第 5 章 数组</b> .....	77
5.1 一维数组 .....	77
5.1.1 一维数组的定义 .....	77
5.1.2 一维数组元素的引用 .....	78
5.1.3 一维数组的初始化 .....	79
5.1.4 一维数组的运算 .....	79
5.2 二维数组 .....	83
5.2.1 二维数组的说明、引用和存储结构 .....	83
5.2.2 二维数组的初始化 .....	84
5.2.3 二维数组的运算 .....	85
5.3 字符数组与字符串数组 .....	87
5.3.1 字符数组的定义和引用 .....	87
5.3.2 字符数组的初始化 .....	88
5.3.3 常用字符串处理函数 .....	88

5.3.4	字符串数组 .....	91
	习题 .....	92
<b>第 6 章</b>	<b>函数与程序结构 .....</b>	<b>93</b>
6.1	C 程序的一般结构 .....	93
6.2	函数定义和函数声明 .....	94
6.2.1	函数定义 .....	94
6.2.2	函数原型声明 .....	96
6.2.3	函数调用 .....	97
6.2.4	参数传递 .....	98
6.2.5	参数数目可变的函数 .....	99
6.3	变量与存储类型 .....	99
6.3.1	局部变量和全局变量 .....	99
6.3.2	变量生命周期 .....	101
6.3.3	存储类型区分符 .....	101
6.4	递归函数 .....	103
6.4.1	递归函数概念 .....	103
6.4.2	递归程序设计 .....	104
6.5	编译预处理 .....	105
6.5.1	宏替换 .....	105
6.5.2	文件包含 .....	107
6.5.3	条件编译 .....	108
	习题 .....	110
<b>第 7 章</b>	<b>指针 .....</b>	<b>112</b>
7.1	指针的概念 .....	112
7.1.1	变量的地址和指针变量 .....	112
7.1.2	指针说明和指针对象的引用 .....	113
7.2	指针参数 .....	117
7.3	数组的指针表示 .....	120
7.3.1	一维数组的指针表示 .....	120
7.3.2	数组作函数参数时的指针表示 .....	123
7.3.3	字符数组的指针表示 .....	126
7.3.4	多维数组的指针表示与指向数组的指针 .....	129
7.4	指针数组 .....	133
7.4.1	指针数组的概念 .....	133
7.4.2	指针变量的指针 .....	135
7.4.3	main 函数的参数 .....	136
7.5	函数的指针 .....	138
7.6	指针函数 .....	139

7.7	指针相关运算	141
7.8	程序举例	143
	习题	146
<b>第8章</b>	<b>结构体、共用体和枚举类型</b>	<b>148</b>
8.1	结构体	148
8.1.1	结构体的概念	148
8.1.2	结构体变量的定义	149
8.1.3	结构体变量的初始化和引用	152
8.1.4	结构体数组	154
8.1.5	指向结构体数据的指针	158
8.2	链表	161
8.2.1	链表的概念	161
8.2.2	链表与动态存储	162
8.2.3	链表的建立	163
8.2.4	对链表的操作	164
8.3	共用体	167
8.3.1	共用体变量的定义	167
8.3.2	共用体变量的引用	169
8.4	枚举类型和自定义类型	170
8.4.1	枚举类型	170
8.4.2	自定义类型	171
	习题	173
<b>第9章</b>	<b>输入输出与低层接口</b>	<b>176</b>
9.1	文件概述	176
9.1.1	文件的概念	176
9.1.2	C文件的分类	177
9.1.3	缓冲文件系统	178
9.1.4	文件类型指针	178
9.1.5	文件的处理过程	179
9.2	文件的打开与关闭	179
9.2.1	文件的打开(fopen 函数)	180
9.2.2	以重定向方式打开文件	181
9.2.3	文件的关闭(fclose 函数)	181
9.3	文件的读写	182
9.3.1	字符读写函数(fgetc 和 fputc)	182
9.3.2	字符串读写函数(fgets 和 fputs)	185
9.3.3	读写数据块函数(fread 和 fwrite)	188

9.3.4	格式化读写函数(fscanf 和 fprintf)	191
9.4	文件的定位及随机读写	193
9.4.1	文件的定位	194
9.4.2	文件的随机读写	194
9.5	文件的出错检测	196
	习题	198

## 中篇 常用算法篇

<b>第 10 章</b>	<b>常用程序设计方法</b>	205
10.1	筛选法求素数及素数应用	205
10.1.1	筛选法求素数	205
10.1.2	素数表的应用	207
10.2	模拟	208
10.3	递推	212
10.4	分治与递归	215
10.5	贪心算法	219
	习题	223

## 下篇 综合案例篇

<b>第 11 章</b>	<b>图书管理系统</b>	227
11.1	系统需求分析	227
11.2	系统总体设计	230
11.3	函数设计	231
11.4	参考代码	236
11.5	运行测试	254
11.6	增加功能	257
11.7	关键技术剖析	258
11.8	导师点评	258
11.9	思考题	259
<b>第 12 章</b>	<b>工资管理系统</b>	260
12.1	系统需求分析	260
12.2	系统总体设计	262
12.3	函数设计	264
12.4	参考代码	271
12.5	运行测试	303
12.6	增加功能	310
12.7	关键技术剖析	311
12.8	导师点评	312



行及階級等長，其成敗是難斷的言辭。了聯合與否情狀，即屬上本本。其  
其真意，其意其味，其意其味。其意其味，其意其味。其意其味，其意其味。  
其意其味，其意其味。其意其味，其意其味。其意其味，其意其味。



上 篇  
基 础 篇

在本书上篇循序渐进地介绍了 C 语言的编程基础知识,并对具体知识进行了详细的实例讲解。包括第 1~9 章,主要讲述 C 语言概述,数据类型、运算符和表达式,简单的 C 程序设计,C 语言流程控制,数组,函数与程序结构,指针,结构体、共用体和枚举类型,输入输出与低层接口等。

# 第 1 章 C 语言概述

## 本章内容

- C 语言的发展与特点。
- C 程序的组成。
- C 语言的基本语法单位。
- C 程序的开发步骤。

## 本章目标

- 了解 C 语言的发展与特点,理解 C 语言的组成。
- 掌握 C 程序的开发步骤。
- 熟悉 C 语言的开发环境。

## 1.1 C 语言的发展及特点

### 1.1.1 程序设计语言

#### 1. 程序与指令

程序(program)是为实现特定目标或解决特定问题而用计算机语言编写的命令序列的集合。程序是由序列组成的,告诉计算机如何完成一个具体的任务。由于现在的计算机还不能理解人类的自然语言,所以还不能用自然语言编写计算机程序。

指令是计算机完成制定操作的命令。在计算机内部指令是由二进制代码(机器码)组成。

#### 2. 程序设计语言的发展

计算机的数学理论基础是图灵于 1937 年提出的图灵机模型,而现代电子计算机的体系结构及实际计算模型则是来自冯·诺依曼 1946 年提出的“程序放入内存,顺序执行”的思想。因此,现在的计算机通常被称为冯·诺依曼计算机。计算机语言的发展历程也从此正式开始,计算机语言的使用人员也称为程序员。

计算机语言的发展从机器语言到汇编语言,再到高级语言。

**机器语言:** 程序是机器指令的序列,使用机器指令编写程序,是人们最初的选择。机器指令的集合就是机器语言。机器语言是二进制的,不易被人理解和掌握;而且程序不易移植。

**汇编语言:** 将每条机器指令配上一个助记符,如 ADD、JUMP 等就形成简单汇编语言。简单汇编语言中的语句与机器指令一一对应。将简单汇编中的与机器相关部分分离出去,由系统完成,就形成宏汇编。现在所说的汇编语言,一般都指宏汇编。汇编语言比机器语言

容易一些,但还是较难掌握;移植性仍然不强。

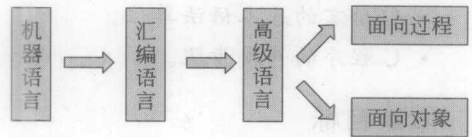
**FORTRAN:** 第一个高级程序设计语言,20世纪50年代由IBM开发,主要用于科学计算,现在仍有人使用。

**BASIC:** 主要用于初级计算机教育,在微型计算机发明后,得到大发展。微软公司发展起步的第一门语言。

**Pascal:** 专为计算机教育而发明的程序设计语言,对于促进结构化程序设计方法的普及有很大作用,现在仍有许多人在学习和使用。

**C/C++:** C与UNIX操作系统结伴而生,由美国的贝尔实验室开发,目标代码效率高,可以用来编系统软件。C++也是贝尔实验室发明,是在C上增加了面向对象特性,是现在使用最广泛的程序设计语言。

**Java:** 最新的面向对象程序设计语言,面向Internet,由Sun公司开发,可以一次编程,到处运行。



程序设计语言的发展如图 1.1 所示。

图 1.1 程序设计语言的发展过程

### 1.1.2 C 语言的发展

C语言是一种面向过程的通用语言,是一种功能强大的计算机高级语言,它既适合于作为系统描述语言,又适合于作为通用的程序设计语言。一个C源程序需要经过编辑、编译和连接后才可运行,对C源程序编译后生成目标文件(.obj),对目标文件和库文件连接后生成可执行文件(.exe)。程序的运行是对可执行文件而言的,所以程序的开发需要语言处理系统的支持,选择一个功能强的语言处理系统可以使程序的开发工作事半功倍。

C语言是国际上广泛流行的计算机高级程序设计语言,它于1973年由美国的贝尔实验室设计发布的。目前,在微型计算机上广泛使用的C语言编译系统有Microsoft C、Turbo C、Borland C等。虽然它们的基本部分都是相同的,但还是有一些差异,所以要了解自己所使用的C编译系统的特点和规定(参阅相应的手册)。

C语言的前身ALGOL 60语言(也称为A语言),1963年,剑桥大学将ALGOL 60语言发展成为CPL(Combined Programming Language)语言。该语言比较接近于硬件,但规模较大,实用性不强。

1967年,剑桥大学的Martin Richards对CPL语言进行了简化,于是产生了BCPL(Basic Combined Programming Language)语言。该语言与CPL相比大为简化,既有结构化程序设计语言的特点,也能直接处理与硬件相关的数据。

1970年,美国贝尔实验室的Ken Thompson将BCPL进行了修改,并为它起了一个有趣的名字“B语言”,并且用B语言写了第一个UNIX操作系统。

1973年,美国贝尔实验室的Dennis M. Ritchie在B语言的基础上最终设计出了一种新的语言,他取了BCPL的第二个字母作为这种语言的名字,这就是C语言。为了使UNIX操作系统推广,1977年Dennis M. Ritchie发表了不依赖于具体机器系统的C语言编译文本《可移植的C语言编译程序》。

1978年Brian W. Kernighian和Dennis M. Ritchie出版了名著《The C Programming Language》,从而使C语言成为目前世界上流行最广泛的高级程序设计语言。

1988年,随着微型计算机的日益普及,出现了许多C语言版本。由于没有统一的标准,使得这些C语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准研究所(American National Standards Institute, ANSI)为C语言制定了一套ANSI标准,俗称C89标准,成为现行的C语言标准。C语言发展迅速,而且成为最受欢迎的语言之一,主要因为它具有强大的功能。许多著名的系统软件,如Windows操作系统、Oracle数据库等都曾经有过用C语言编写的版本。用C语言加上一些汇编语言子程序,就更能显示C语言的优势了,像PC-DOS、WORDSTAR等就是用这种方法编写的。另外,C语言更适合于单片机、嵌入式开发等低层的应用中。

在ANSI标准化后,C语言的标准在一段相当的时间内都保持不变,尽管C++继续在改进。在20世纪90年代才经历了改进,这就是ISO 9899:1999(1999年出版)。这个版本就是通常提及的C99。它被ANSI于2000年3月采用。

C99是在C89的基础上发展起来的,增加了基本数据类型、关键字和一些系统函数等。在初学阶段C89(ANSI C)和C99的区别是不易察觉的,所以不必太在意这个。

目前的C99标准其实在许多编译器中已被或多或少的支持了,目前完全支持的有MinGW、Borland C++、dev-C++等。如果想了解更多C99标准的新特性,可以参考相关技术书籍。

本书以C89标准为蓝本来介绍C语言的知识。

### 1.1.3 C语言的特点

C语言是结构化程序设计语言。C语言同时具有汇编语言和高级语言的双重特性。从用户应用、实现难易程度、程序设计风格来看,C语言的特点如下。

(1) 表达能力强。丰富的数据类型和运算符,可直接访问内存物理地址和硬件寄存器,能实现二进制位运算。可直接对硬件进行操作所以又被称为中间语言。

(2) 流程控制结构化、程序结构模块化。有顺序、分支、循环三种控制结构。

(3) 语言简练。如 $i=i+1$ ;在C中可写为 $i++$ ;

(4) 使用灵活。数据类型无严格的对应关系。

(5) 生成的目标代码质量好,程序执行效率高,速度快。

(6) 程序通用性、可移植性好。C语言没有依赖于硬件的输入输出语句,而采用系统的库函数进行输入输出操作所以C语言不依赖于任何硬件系统。

## 1.2 C程序的组成

学习程序设计语言和学习人类自然语言是一样的,学习外语时,首先要学习字母表、单词、语法、语义语用。更重要的是要通过不断地编程实验,逐步去领会和掌握程序设计的思想和方法。熟练的编程技能是在知识和经验不断积累的基础上发展起来的。对于初学者就要试着编写程序,先模仿例子,试着改写并逐步体会。就像学习外语时要背很多文章后,很自然地可以用外语写文章。

首先通过例子,先认识一下简单的C语言程序。