

中国高等学校计算机科学与技术专业（应用型）规划教材

丛书主编 陈明

C程序设计教程

——理论与实践

刘维富 陈建平 王春明 编著



清华大学出版社

中国高等学校计算机科学与技术专业（应用型）规划教材

丛书主编 陈明

C程序设计教程

——理论与实践

刘维富 陈建平 王春明 编著

清华大学出版社

北京

内 容 提 要

本书中的理论篇按 C 语言实际编程能力形成的两个关键期,将 C 语言课堂教学内容分为两大单元,即结构化程序设计(第 1~4 章)和模块化程序设计(第 5~10 章),知识体系按知识链优化。以典型范例程序为主体,适时阐述有关程序设计的思想、方法、C 语言语法、基本算法和编程技巧,理论联系实际,注重读者实际编程能力的培养。不回避教学 and 实际编程中的难点,想方设法地将内容讲清讲透,力求使读者突破难点,学以致用。

实践篇与理论篇配套,包括实验指导。前 12 个实验与课堂教学同步,每个实验包括调试题、编程题和选做题;第 13 个实验可用于课程设计。C 语言程序开发环境。介绍主流的 C 程序集成开发环境——Visual C++ 6.0 和等级考试的上机环境——Turbo C 2.0。

全书按 C 语言标准(C89)编写。精心设计的例题、调试题、编程题对读者深入理解、准确掌握和熟练运用 C 语言极具参考价值和挑战性。

本书不仅适合作为高校学生学习 C 语言的教材,而且适合程序设计的初学者或有一定基础、希望突破编程难点的读者参考。

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 程序设计教程:理论与实践/刘维富,陈建平,王春明编著. —北京:清华大学出版社, 2011.1

(中国高等学校计算机科学与技术专业(应用型)规划教材)

ISBN 978-7-302-22958-2

I. ①C… II. ①刘… ②陈… ③王… 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 105414 号

责任编辑:谢 琛 赵晓宁

责任校对:李建庄

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

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

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

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

印 刷 者:北京密云胶印厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260

印 张:25.25

字 数:627 千字

版 次:2011 年 1 月第 1 版

印 次:2011 年 1 月第 1 次印刷

印 数:1~3000

定 价:38.00 元

产品编号:037449-01

编 委 会

主 任：陈 明

副主任：蒋宗礼 卢先和

委 员：	常 虹	陈国君	陈 峻	陈晓云	陈笑蓉
	丛 琳	方路明	段友祥	高文胜	巩君华
	关 永	郭 禾	郝 莹	何胜利	何晓新
	贺安坤	胡巧多	李陶深	李仲麟	刘东升
	刘贵龙	刘晓强	刘振华	路 游	马杰良
	毛国君	苗凤君	宁 玲	施海虎	宋长龙
	宋立军	孙践知	孙中胜	汤 庸	田俊峰
	万本庭	王让定	王锁柱	王 新	王兆青
	王智广	王志强	谢 琛	谢书良	徐孝凯
	徐子珊	杨建刚	姚 琳	叶春蕾	叶俊民
	袁 薇	张建林	张 杰	张 武	张晓明
	张艳萍	周 苏	曾 一	訾秀玲	

序 言

应用是推动学科技术发展的原动力,计算机科学是实用科学,计算机科学技术广泛而深入地应用推动了计算机学科的飞速发展。应用型创新人才是科技人才的一种类型,应用型创新人才的重要特征是具有强大的系统开发能力和解决实际问题的能力。培养应用型人才的教学理念是教学过程中以培养学生的综合技术应用能力为主线,理论教学以够用为度,所选择的教学方法与手段要有利于培养学生的系统开发能力和解决实际问题的能力。

随着我国经济建设的发展,对计算机软件、计算机网络、信息系统、信息服务和计算机应用技术等专业技术方向的人才的需求日益增加,主要包括软件设计师、软件评测师、网络工程师、信息系统监理师、信息系统的管理工程师、数据库系统工程师、多媒体应用设计师、电子商务设计师、嵌入式系统设计师和计算机辅助设计师等。如何构建应用型人才培养的教学体系以及系统框架,是从事计算机教育工作者的责任。为此,中国计算机学会计算机教育专业委员会和清华大学出版社共同组织启动了《中国高等学校计算机科学与技术专业(应用型)学科教程》的项目研究。参加本项目的研究人员全部来自国内高校教学一线具有丰富实践经验的专家和骨干教师。项目组对计算机科学与技术专业应用型学科的培养目标、内容、方法和意义,以及教学大纲和课程体系等进行了较深入、系统的研究,并编写了《中国高等学校计算机科学与技术专业(应用型)学科教程》(简称《学科教程》)。《学科教程》在编写上注意区分应用型人才与其他人才在培养上的不同,注重体现应用型学科的特征。在课程设计中,《学科教程》在依托学科设计的同时,更注意面向行业产业的实际需求。为了更好地体现《学科教程》的思想与内容,我们组织编写了《中国高等学校计算机科学与技术专业(应用型)规划教材》,旨在能为计算机专业应用型教学的课程设置、课程内容以及教学实践起到一个示范作用。本系列教材的主要特点如下:

1. 完全按照《学科教程》的体系组织编写本系列教材,特别是注意在教材设置、教材定位和教材内容的衔接上与《学科教程》保持一致。
2. 每门课程的教材内容都按照《学科教程》中设置的大纲精心编写,尽量体现应用型教材的特点。
3. 由各学校精品课程建设的骨干教师组成作者队伍,以课程研究为基础,将教学的研究成果引入教材中。
4. 在教材建设上,重点突出对计算机应用能力和应用技术的培养,注重教材的实践性。
5. 注重系列教材的立体配套,包括教参、教辅以及配套的教学资源、电子课件等。

高等院校应培养能为社会服务的应用型人才,以满足社会发展的需要。在培养模式、教学大纲、课程体系结构和教材都应适应培养应用型人才的目标。教材体现了培养目标和育人模式,是学科建设的结晶,也是教师水平的标志。本系列教材的作者均是多年从事计算机科学与技术专业教学的教师,在本领域的科学研究与教学中积累了丰富的经验,他们将教学研究和科学研究的成果融入教材中,增强了教材的先进性、实用性和实践性。

目前,我们对于应用型人才培养的模式还处于探索阶段,在教材组织与编写上还会有这样或那样的缺陷,我们将不断完善。同时,我们也希望广大应用型院校的教师给我们提出更好的建议。

《中国高等学校计算机科学与技术专业(应用型)规划教材》主编

陈明

2008年7月

前言

C 语言是面向过程的结构化、模块化的主程序设计语言,广泛用于编写系统软件和应用软件,广泛用于程序设计、数据结构等课程的教学。C 程序设计是高校普遍开设的计算机核心基础课程,涉及编程思想、方法、语法、算法、编程技巧、调试技术和操作技能,理论性、综合性和实践性强,使不少人感到难学、难入门,甚至入门后半途而废。

作者一直从事 C/C++ 程序设计教学、研究、建设和软件开发,阅读了国内外大量 C/C++ 语言教材,了解初学者学习 C/C++ 语言的困难,积累和总结了 C/C++ 语言教学的成功经验,形成了颇具特色的负反馈教学法,力求使 C/C++ 语言不再难学。

本书经过作者长期构思,精心写作,具有如下特点。

(1) 结构合理,层次分明。将 C 语言教学内容分为两大单元,即结构化程序设计(第 1~4 章)和模块化程序设计(第 5~10 章)。这样由浅入深,循序渐进的组织符合读者的认识规律和编程能力的形成规律,便于教学的组织、实施和考核,有利于教学效果的巩固和教学质量的提高。

(2) 体系优化,内容充实。准确的系统化知识体系是缩减教材篇幅、提升教材质量的法宝,充实的知识链是知识向能力转化的捷径。

① 对于 C 语言的特色内容、重点和难点内容——指针,首先从概念上澄清和纠正了以往教材中普遍存在的问题——“指针就是地址,地址就是指针”;其次从有利于读者形成稳固的编程知识链方面考虑,将指针合理分散到全书相关章节(第 1 章的 1.4 节,第 3 章的 3.4 节,第 5 章的 5.4 节,第 7~10 章),从一开始就接触指针,使读者有足够的时间分别从指针的不同应用领域逐步理解、掌握和运用指针,使指针不再神秘和恐怖,充分释放指针的强大编程威力。

② 进一步强化模块化程序设计。首先将代码模块化的内容(函数)提前到第 5 章讲解,其后所有章节例题中的程序基本采用主函数加多个辅助函数的形式实现,使读者尽早使用模块化思维方式编程。其次在后续章节陆续介绍数据模块化的内容,为编写复杂、大型程序奠定基础。

③ 充实并增加了编程中非常实用的内容,如程序调试简介(第 4 章的 4.7 节),程序的多文件组织(第 6 章的 6.4 节),带参数的 main 函数(第 8 章的 8.4.2 节),指针参数传递与数据安全(第 8 章的 8.5 节),void 型指针及通用类型程序设计(第 8 章的 8.6 节),参数个数可变的函数(第 8 章的 8.8 节),为读者进一步提高奠定了坚实的基础。

(3) 范例程序,易学易用。通过以基本语法和基本算法为主线的典型、综合范例程序,适时阐述有关程序设计的方法和思想,将 C 语言语法、基本算法、程序设计方法和编程技巧有机结合起来,使相关知识形成知识链,理论联系实际,注重读者实际编程能力培养。书中大量的范例程序经过作者精心挑选和精心设计,表达准确、简练,书写规范,模块的独立性和示范性强,大部分代码可直接或稍作修改即可在实际编程中使用,是读者练习编程和实际编程的活代码。

(4) 不避难点,力求突破。许多有一定基础的读者学习程序设计半途而废的重要原因之一是未能突破实际编程中的难点问题,这与不少主流教材以各种理由极力回避难点有密切的关系。本书针对教学和实际编程中的难点(如递归、指针、堆内存的分配和使用、单向链表等),精挑多选范例程序,想方设法讲清讲透,力求使读者突破难点,学以致用。

(5) 慎选标准,优择环境。本书介绍的是标准 C 语言,符合 ANSI/ISO C 标准(C89),引导读者按流行 C 语言标准编写程序。C 标准之所以选用 C89 而未选用最新的 C99,主要原因是 C89 得到广泛支持,且 C89 是制定 C++ 标准的基础。而 C99 扩展了一些 C++ 不兼容的内容,到目前为止尚未得到广泛应用。书中所有例题中的程序均在 Visual C++ 6.0 和 Dev C++ 4.9.9.2 中运行通过。前者是主流的商业化的 C/C++ 集成编程环境,后者是非商业化的 C/C++ 集成编程环境。

(6) 重视编程,强化实践。勇于挑战是学好 C 语言程序设计的前提,独立编程是学好 C 语言程序设计的必由之路。学习编程离不开实验,实验离不开调试技术。

① 实验指导。第 11 章以调试技术、基本算法、综合编程为核心内容,共拟 13 个实验。前 12 个实验与课堂教学同步,每个实验包括调试题、编程题和选做题,供不同层次读者选用;第 13 个实验可用于课程设计。

② C 语言程序开发环境。第 12 章和第 13 章分别介绍主流的 C 语言程序集成开发环境——Visual C++ 6.0 和等级考试的上机环境——Turbo C 2.0,使读者具备自主上机调试程序的能力。

书中带“*”号的章节是实际编程中非常实用的内容,但超出了 C 语言二级考试大纲,仅供读者选用。本书的配套课件是作者在校内讲授 C 语言程序设计课程的实际课件,内容翔实,可供读者从清华大学出版社网站(www.tup.com)免费下载。

全书由陈建平教授、王波教授主审,全程得到清华大学出版社的支持和帮助,在此表示衷心感谢。由于作者水平有限,书中难免有错误与不妥之处,恳请同行和读者批评指正。

作者

2010 年 12 月

目 录

上篇 理 论 篇

第 1 章 C 语言概述	3
1.1 C 语言的起源和发展	3
1.2 C 语言的主要特点	4
1.3 C 编译器	4
1.4 C 语言程序设计	5
1.5 C 语言程序的开发步骤和上机调试流程	7
习题	8
第 2 章 数据类型、运算符和表达式	9
2.1 C 语言的字符集、关键字和标识符	9
2.1.1 C 语言的字符集	9
2.1.2 标识符和关键字	9
2.2 C 语言的基本数据类型	10
2.2.1 基本数据类型	11
2.2.2 常量	13
2.2.3 变量	16
2.3 运算符与表达式	18
2.3.1 算术运算符与算术表达式	19
2.3.2 ++ 和 -- 运算符	22
2.3.3 赋值运算符与赋值表达式	22
2.3.4 数据类型转换	23
2.3.5 关系运算符与关系表达式	25
2.3.6 逻辑运算符与逻辑表达式	25
2.3.7 逗号运算符	26
2.3.8 条件运算符	27

2.3.9	sizeof 运算符	27
2.3.10	位运算符	28
2.4	常用库函数	29
2.4.1	数学库函数	29
2.4.2	伪随机函数	31
习题		31
第 3 章	基本类型数据的输入和输出	33
3.1	C 语言的输入输出	33
3.2	字符输入输出函数	33
3.3	格式化输出函数 printf	34
3.3.1	printf 函数概述	34
3.3.2	printf 函数的格式说明	35
3.3.3	printf 函数的使用	37
3.4	格式化输入函数 scanf	40
3.4.1	scanf 函数概述	40
3.4.2	scanf 函数的格式说明	41
3.4.3	scanf 函数的使用	42
3.5	程序举例	44
习题		46
第 4 章	流程控制	49
4.1	算法	49
4.1.1	算法的概念	49
4.1.2	算法举例	50
4.1.3	算法的特性	50
4.1.4	算法的表达	50
4.1.5	三种基本流程控制结构	51
4.2	C 语言的语句	52
4.3	选择结构语句	52
4.3.1	条件语句	53
4.3.2	开关语句	57
4.4	循环结构语句	60
4.4.1	while 语句	61
4.4.2	do...while 语句	62
4.4.3	for 语句	63
4.4.4	循环语句小结	66

4.5	转向语句	68
4.5.1	break 语句	68
4.5.2	continue 语句	70
4.5.3	goto 语句简介	70
4.5.4	exit 函数	71
4.6	程序举例	71
4.7	程序调试简介	78
4.7.1	程序的错误类型	78
4.7.2	程序调试	79
	习题	79

第5章 函数 81

5.1	函数定义和调用	81
5.1.1	函数定义	81
5.1.2	函数调用	83
5.1.3	函数原型	85
5.2	标识符的作用域	88
5.2.1	块作用域(局部作用域)	88
5.2.2	文件作用域(全局作用域)	89
5.2.3	函数原型作用域	90
5.2.4	函数作用域	91
5.3	变量的存储种类	91
5.3.1	自动变量	92
5.3.2	寄存器变量	92
5.3.3	外部变量	93
5.3.4	静态变量	95
5.3.5	小结	97
5.4	指针基础	97
5.4.1	地址的概念	98
5.4.2	指针的概念	98
5.4.3	指针变量	98
5.4.4	指针做函数的参数	99
5.4.5	指针做函数的返回值	101
5.5	递归函数	102
5.6	模块化程序设计方法	106
	习题	111

第 6 章 编译预处理	115
6.1 文件包含	115
6.2 宏	117
6.2.1 不带参数的宏	117
6.2.2 带参数的宏	119
6.3 条件编译	121
6.4 程序的多文件组织	125
6.4.1 程序的多文件组织方法	125
6.4.2 多文件程序举例	125
6.4.3 多文件程序的编译和链接	126
习题	127
第 7 章 数组	129
7.1 一维数组	129
7.1.1 一维数组的定义、引用和初始化	129
7.1.2 一维数组的赋值和输入输出	131
7.1.3 一维数组做函数参数	133
7.1.4 排序	136
7.1.5 查找	140
7.1.6 大整数运算和高精度运算	144
7.2 二维数组	148
7.2.1 二维数组的定义	148
7.2.2 二维数组的引用	149
7.2.3 二维数组的初始化	149
7.2.4 二维数组的赋值和输入输出	150
7.2.5 应用举例	151
7.3 字符数组	153
7.3.1 字符数组的定义、初始化和使用	153
7.3.2 字符数组的输入和输出	155
7.3.3 字符串处理函数	158
7.3.4 应用举例	161
习题	165
第 8 章 指针	171
8.1 指针的运算	171
8.1.1 指针的赋值	171

8.1.2	指针的算术运算	173
8.1.3	指针的关系运算	174
8.1.4	指针值的输出	175
8.2	指针与数组	176
8.2.1	指针与一维数组	176
8.2.2	指针与多维数组	177
8.2.3	指针与字符串	180
8.3	指针数组和指向指针的指针变量	182
8.3.1	指针数组	182
8.3.2	指向一维数组的指针变量	185
8.3.3	指向指针的指针变量	186
8.4	指针与函数	187
8.4.1	数组做函数的参数	187
8.4.2	带参数的 main 函数	190
8.4.3	返回值为指针的函数	193
8.4.4	指向函数的指针及通用算法设计	194
8.5	指针参数传递与数据安全	197
8.5.1	const 变量	197
8.5.2	const 指针做函数参数	198
8.6	void 型指针及通用类型程序设计	200
8.7	堆内存的申请、使用和释放	203
8.7.1	malloc 函数和 free 函数	204
8.7.2	应用举例	205
8.8	参数个数可变的函数	208
	习题	212

第 9 章 自定义数据类型 215

9.1	结构体	215
9.1.1	结构体类型的定义	216
9.1.2	结构体变量的定义	216
9.1.3	结构体变量的使用	218
9.1.4	位域	223
9.2	单向链表	227
9.2.1	单向链表的概念	227
9.2.2	单向链表的建立和基本操作	228
9.3	共用体	233
9.3.1	共用体类型的定义	233
9.3.2	共用体类型变量的定义和使用	233

9.4	枚举	237
9.4.1	枚举类型的定义	238
9.4.2	枚举类型变量的定义	238
9.4.3	枚举类型变量的使用	239
9.5	类型别名	243
	习题	245
第 10 章	文件	247
10.1	基本概念	247
10.1.1	字节流	247
10.1.2	文件	247
10.1.3	缓冲	248
10.1.4	文件指针	248
10.2	文件的打开与关闭	249
10.2.1	文件的打开	249
10.2.2	文件的关闭	250
10.3	文本文件的读写	250
10.3.1	fputc 函数和 fgetc 函数	251
10.3.2	fgets 函数和 fputs 函数	254
10.3.3	fprintf 函数和 fscanf 函数	255
10.4	二进制文件的读写	257
10.4.1	fread 函数和 fwrite 函数	257
10.4.2	文件的位置指针与随机读写	258
10.5	出错状态的检测和清除	263
10.5.1	ferror 函数	263
10.5.2	clearerr 函数	263
10.6	综合应用举例	263
	习题	267

下篇 实 践 篇

第 11 章	实验指导	271
11.1	实验目的与要求	271
11.2	实验一 VC++ 6.0 IDE 与 C 语言程序开发	273
11.3	实验二 基本数据的输入输出	279
11.4	实验三 选择结构	280

11.5	实验四 循环结构	282
11.6	实验五 函数	284
11.7	实验六 递归	286
11.8	实验七 数组	288
11.9	实验八 字符串	291
11.10	实验九 指针(1)	293
11.11	实验十 指针(2)	295
11.12	实验十一 结构体与单向链表	298
11.13	实验十二 文件	302
11.14	实验十三 综合	304
第 12 章	Visual C++ 6.0 集成开发环境	307
12.1	Visual C++ 的集成开发环境	307
12.1.1	菜单栏	307
12.1.2	工具栏	312
12.1.3	工作区窗口	314
12.1.4	文本编辑器和编辑菜单	315
12.1.5	文件菜单	317
12.1.6	Build 菜单	318
12.2	Visual C++ 开发 C 语言程序的基本过程	319
12.2.1	创建 C 语言程序的方式	319
12.2.2	源程序的编辑与格式化	322
12.2.3	多文件程序的组织	323
12.2.4	程序的编译、链接和运行	323
12.3	程序调试	324
12.3.1	程序的错误类型	324
12.3.2	编译错误、链接错误的查看和修改	325
12.3.3	运行错误、逻辑错误的判断与调试	326
12.3.4	基本调试手段	327
12.3.5	集成开发环境的跟踪调试功能	329
12.4	Visual C++ 的帮助功能	333
第 13 章	Turbo C 2.0 集成开发环境	335
13.1	Turbo C 2.0 的安装	335
13.2	TC 环境的启动与退出	336
13.3	TC 环境的工作界面介绍	337
13.4	编辑源程序	339

13.5	编译、连接和运行程序	341
13.6	设置 TC 工作环境	347
13.7	程序调试方法	348
13.8	TC 菜单项的功能	352
13.8.1	File 菜单	352
13.8.2	Edit 命令	353
13.8.3	Run 菜单	355
13.8.4	Compile 菜单	355
13.8.5	Project 菜单	356
13.8.6	Debug 菜单	356
13.8.7	Break/watch 菜单	357
13.8.8	Options 菜单	357
13.8.9	常用热键	360
13.9	常见错误信息	362
13.9.1	严重错误	362
13.9.2	一般错误	362
13.9.3	警告	366
13.10	联机帮助	367
附录 A	ASCII 码表	370
附录 B	数制	371
B.1	数制的基本概念	371
B.2	不同数制之间数的相互转换	372
附录 C	实数的表示法	374
附录 D	C 语言常用标准库函数	376
D.1	数学库函数	376
D.2	字符处理函数	377
D.3	字符串处理函数	378
D.4	内存操作函数	379
D.5	标准输入输出库函数	379
D.6	通用函数	381
D.6.1	堆内存分配函数	381
D.6.2	伪随机函数	381
D.6.3	数据转换函数	382
D.6.4	其他函数	382
D.7	访问函数可变参数的宏	383

D.8 时间和日期函数	384
附录 E C99 简介	385
参考文献	386