

普通高校本科计算机专业 **特色** 教材精选

C/C++ 语言程序设计

孟军 赵晶 王凡 编著

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

清华大学出版社



普通高校本科计算机专业 **特色** 教材精选

C/C++ 语言程序设计

孟军 赵晶 王凡 编著

清华大学出版社
北京

内 容 简 介

本书是介绍 C 语言程序设计的实用教材。全书以 C 语言程序设计为主, C++ 语言程序设计为辅。本书全面系统地介绍了 C 语言基本数据类型、运算符和表达式、数据的输入输出、分支和循环控制结构、数组、函数、指针、结构体、共用体和枚举、编译预处理、文件操作方法, 以及 C++ 语言中类和对象的概念、继承性与派生类、多态性的用法、简单的面向对象的程序设计方法。读者在学会使用 C 语言后, 可以轻松过渡到 C++ 语言的学习。

本书以基本语法规则为主线, 通过典型应用向读者介绍程序设计技术。本书以面向应用、重视实践、便于学习为原则, 通俗易懂, 由浅入深, 便于初学者学习和掌握, 为读者能够尽快利用 C 语言和 C++ 语言编写大型实用程序打下良好的基础。

本书不仅可作为高等院校程序设计课程的入门教材, 也可作为广大 C 语言和 C++ 语言编程者自学程序设计方法的参考用书。

版权所有, 翻印必究。举报电话: 010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

C/C++ 语言程序设计/孟军, 赵晶, 王凡编著. —北京: 清华大学出版社, 2004. 9

(普通高校本科计算机专业特色教材精选)

ISBN 7-302-09062-9

I. C… II. ①孟… ②赵… ③王… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 071198 号

出 版 者: 清华大学出版社

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

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

组稿编辑: 张瑞庆

文稿编辑: 霍志国

印 刷 者: 北京嘉实印刷有限公司

装 订 者: 三河市新茂装订有限公司

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 19 字数: 421 千字

版 次: 2004 年 9 月第 1 版 2004 年 9 月第 1 次印刷

书 号: ISBN 7-302-09062-9/TP·6400

印 数: 1~5000

定 价: 25.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或 (010)62795704

编审委员会

主任：蒋宗礼

副主任：李仲麟 何炎祥

委员：（排名不分先后）

王向东 宁 洪 朱庆生 吴功宜 吴 跃

张 虹 张 钢 张为群 余雪丽 陈志国

武 波 孟祥旭 孟小峰 胡金初 姚放吾

原福永 黄刘生 廖明宏 薛永生

秘书长：王听讲

出版说明

INTRODUCTION

在我国高等教育逐步实现大众化后,越来越多的高等学校将会面向国民经济发展的第一线,为行业、企业培养各级各类高级应用型专门人才。为此,教育部已经启动了“高等学校教学质量和教学改革工程”,强调要以信息技术为手段,深化教学改革和人才培养模式改革。如何根据社会的实际需要,根据各行各业的具体人才需求,培养具有特色显著的人才,是我们共同面临的重大问题。具体地,培养具有一定专业特色的和特定能力强的计算机专业应用型人才则是计算机教育要解决的问题。

为了适应 21 世纪人才培养的需要,培养具有特色的计算机人才,急需一批适合各种人才培养特点的计算机专业教材。目前,一些高校在计算机专业教学和教材改革方面已经做了大量工作,许多教师在计算机专业教学和科研方面已经积累了许多宝贵经验。将他们的教研成果转化为教材的形式,向全国其他学校推广,对于深化我国高等学校的教学改革是一件十分有意义的事。

清华大学出版社在经过大量调查研究的基础上,决定编写出版一套“普通高校本科计算机专业特色教材精选”。本套教材是针对当前高等教育改革的新形势,以社会对人才的需求为导向,主要以培养应用型计算机人才为目标,立足课程改革和教材创新,广泛吸纳全国各地的高等院校计算机优秀教师参与编写,从中精选出版确实反映计算机专业教学方向的特色教材,供普通高等院校计算机专业学生使用。

本套教材具有以下特点:

1. 编写目的明确

本套教材是在深入研究各地各学校办学特色的基础上,面向普通高校的计算机专业学生编写的。学生通过本套教材,主要学习计算机科学与技术专业的基本理论和基本知识,接受利用计算机解决实际问题的基本训练,培养研究和开发计算机系统,特别是应用系统的基本能力。

2. 理论知识与实践训练相结合

根据计算学科的三个学科形态及其关系,本套教材力求突出学科的理论与实践紧密结合的特征,结合实例讲解理论,使理论来源于实践,又进一步指导实践,学生通过实践深化对理论的理解,更重要的是使学生学会理论方法的实际运用。在编写教材时突出实用性,并做到通俗易懂,易教易学,使学生不仅知其然,知其所以然,还要会其如何然。

3. 注意培养学生的动手能力

每种教材都增加了能力训练部分的内容,学生通过学习和练习,能比较熟练地应用计算机知识解决实际问题。既注重培养学生分析问题的能力,也注重培养学生解决问题的能力,以适应新经济时代对人才的需要,满足就业要求。

4. 注重教材的立体化配套

大多数教材都将陆续配套教师用课件、习题及其解答提示,学生上机实验指导等辅助教学资源,有些教材还提供能用于网上下载的文件,以方便教学。

由于各地区各学校的培养目标、教学要求和办学特色均有所不同,所以对特色教学的理解也不尽一致,我们恳切希望大家在使用教材的过程中,及时地给我们提出批评和改进意见,以便我们做好教材的修订改版工作,使其日趋完善。

我们相信经过大家的共同努力,这套教材一定能成为特色鲜明、质量上乘的优秀教材,同时,我们也希望通过本套教材的编写出版,为“高等学校教学质量和教学改革工程”作出贡献。

清华大学出版社

前言

PREFACE

C 语言是一种受到广泛重视和应用的通用程序设计语言。在 30 年的发展历史中, C 语言不断丰富和完善, 使其成为进行结构化程序设计的优选语言。C 语言不仅可以设计系统软件, 也可以设计应用软件。C 语言是一种高级语言, 同时还具有低级语言(汇编语言)的许多特点, 可以对系统硬件及其外设接口进行编程, 显示其较强的代码可移植性、较高的开发效率和维护性。

随着计算机应用的不断深入, C 程序设计语言已经无法胜任处理高度复杂的程序, 程序的调试、扩充和维护变得极其困难。为了弥补结构化程序设计的不足, 支持面向对象的程序设计的 C++ 语言应运而生。

C++ 语言是 20 世纪 90 年代以来最受欢迎的程序设计语言之一。它不仅保留了 C 语言高效简洁的特点, 更重要的是支持一种全新的面向对象的程序设计机制, 使得 C++ 语言成为既可以进行结构化程序设计, 又可以进行面向对象程序设计的开发高质量软件的最佳语言。

本书是根据作者多年的 C 语言教学经验和实际应用的体会而写成的, 主要以介绍 C 语言的基本概念和结构化程序设计思想为主, 同时介绍 C++ 语言的基本概念、最简单的 C++ 的程序结构、面向对象编程的 4 个要点(封装、数据隐藏、继承和多态性), 并演示在 C++ 的集成开发器 Visual C++ .NET 环境下, 用 C++ 语言设计面向对象的 DOS 应用程序的过程, 使读者加深理解面向对象程序设计的基本思想, 提高程序设计能力。

本书的主要特点:

(1) 强化基本概念。无论是 C 语言还是 C++ 语言, 重点讲解基本概念、基本程序设计思想、常用算法分析和训练。

在讲解基本概念和语法时, 提供了适当的例题, 为培养程序设计的能力, 对例题中出现的算法, 基本上都给出了分析和提示。每章都包含“应用举例”一节, 以便深入理解本章的主要内容。

(2) 简洁清晰, 通俗易懂。本书力争用简练通俗的语言, 描述程序设计的基本概念, 使读者较容易抓住结构化程序设计和面向对象程序设计

的设计思想,以便强化读者程序设计的能力。本书避免让读者陷入面面俱到、复杂而烦琐的语法细节中。

(3) 每章的最后都附有一定量的习题,这些习题对于读者进一步巩固已学习的内容,对于熟悉一些常用算法都大有好处。这些习题除基本概念和编程题之外,还安排了一些选择题、改错题和填空题等,这样可使读者在学习知识点的同时,也熟悉了考试题型。

(4) 学习 C 语言除掌握基本理论之外,还必须加强实践训练。本书中 C 语言部分的所有例题都在微机上使用 Turbo C 2.0 调试通过, C++ 部分的例题是采用 VC++ 6.0 调试通过的。读者可以边学习边上机,迅速提高编程水平。

作者建议:

(1) 对于初次学习高级语言进行程序设计的读者来说,重点应放在程序设计思想和方法的训练上,因此应该选择相对容易理解、功能强大、应用广泛的 C 语言作为首选程序设计语言。在学习的过程中,不要把精力放在细枝末节上,不要好高骛远急于设计复杂的应用程序,甚至 Windows 程序,否则会舍本逐末。只有打好程序设计的基础,强化基本算法的设计功底,才是明智之举。因为无论多么复杂的程序,都离不开一系列的基本算法。

另外,必须强化上机实践,才能更好地体会所学的内容,而一个编好的程序能够正确运行才是最终的目的。选择 C 语言的集成开发器 Turbo C 2.0,可以在一个简单易学的环境中调试 C 程序。

(2) 由于 C++ 语言在编写复杂的大型应用软件方面显示了强大的生命力,由 C 过渡到 C++ 是必然趋势。C++ 是从 C 进化而来,先学习 C 语言,有利于学好 C++ 语言。在学习 C++ 语言时,要把学习的重点放在面向对象程序设计的基本概念和面向对象程序设计思想方面,建议利用面向对象程序设计思想编写 DOS 应用程序,以加深体会面向对象程序设计思想的精髓。不要急于编写面向对象的 Windows 程序,因为它会过早地使读者把精力放在 Windows 程序设计的机制上面。

由于作者水平有限,经验不足,书中一定有不少缺点和错误,敬请同行和广大读者批评指正。

作 者

2004 年 3 月于大连理工大学

目 录

CONTENTS

第1章 C语言概述	1
1.1 C语言简介	1
1.1.1 C语言的发展历史	1
1.1.2 C语言的特点	2
1.2 简单的C语言程序	3
1.2.1 C语言程序的结构	3
1.2.2 C语言程序的书写规则	6
1.3 C语言的编译与实现	6
1.3.1 C语言程序的开发过程	6
1.3.2 如何运行C语言程序	6
习题	8
第2章 数据类型、运算符和表达式	9
2.1 C语言的数据类型	9
2.1.1 数据类型的概念	9
2.1.2 C语言的数据类型	9
2.2 常量和变量	10
2.3 基本数据类型	11
2.3.1 整型数据	11
2.3.2 实型数据	13
2.3.3 字符型数据	15
2.3.4 符号常量	16
2.4 运算符与表达式	17
2.4.1 算术运算符和算术表达式	17
2.4.2 赋值运算符和赋值表达式	20
2.4.3 关系运算符和关系表达式	21
2.4.4 逻辑运算符和逻辑表达式	22

2.4.5	条件运算符和条件表达式	24
2.4.6	移位运算符	24
2.4.7	逗号运算符和逗号表达式	26
2.5	数据类型之间的转换与运算	27
2.5.1	自动类型转换	27
2.5.2	强制类型转换	28
2.6	其他运算符	29
2.7	应用举例	29
习题		30
第3章	语句	33
3.1	C 语言语句概述	33
3.2	数据的输入和输出	34
3.2.1	字符输入函数	34
3.2.2	格式化输入函数	35
3.2.3	字符输出函数	38
3.2.4	格式化输出函数	39
3.3	if 条件选择语句	42
3.3.1	if 语句的一般形式	42
3.3.2	if 语句的嵌套形式	44
3.4	switch 开关分支语句	45
3.5	循环语句	47
3.5.1	while 循环语句	47
3.5.2	do...while 循环语句	48
3.5.3	for 循环语句	49
3.5.4	循环的嵌套	51
3.6	控制结构中的跳转语句	52
3.6.1	break 语句	52
3.6.2	continue 语句	53
3.6.3	goto 语句	54
3.6.4	return 语句	55
3.7	应用举例	56
习题		59
第4章	数组	61
4.1	一维数组	61
4.1.1	一维数组的定义	61
4.1.2	一维数组元素的引用	62

4.1.3 一维数组的初始化	63
4.2 二维数组	65
4.2.1 二维数组的定义	65
4.2.2 二维数组元素的引用	66
4.2.3 二维数组的初始化	67
4.3 字符数组和字符串	70
4.3.1 字符数组的定义	70
4.3.2 字符串	70
4.3.3 字符数组的初始化	71
4.3.4 字符数组的输入输出	72
4.3.5 常用的字符串处理函数	74
4.4 应用举例	76
习题	79
第5章 函数和存储类	81
5.1 函数的概念	81
5.2 函数的定义和调用	82
5.2.1 函数的定义	82
5.2.2 函数的调用	84
5.3 函数的返回值及其类型	85
5.4 函数的参数及其传递方式	87
5.4.1 非数组作为函数参数	87
5.4.2 数组名作为函数参数	88
5.5 函数的嵌套调用和递归调用	90
5.5.1 函数的嵌套调用	90
5.5.2 函数的递归调用	92
5.6 变量的作用域和存储类	94
5.6.1 变量的作用域	94
5.6.2 变量的存储类	96
5.7 内部函数和外部函数	101
5.8 应用举例	102
习题	107
第6章 指针	109
6.1 指针的基本概念	109
6.2 指针变量的定义和引用	110
6.2.1 指针变量的定义	110
6.2.2 指针变量的引用	112

6.3 指针与数组	113
6.3.1 用指针访问一维数组	114
6.3.2 用指针访问多维数组	115
6.3.3 用指针访问字符串	117
6.3.4 指针数组	118
6.3.5 多级指针	120
6.4 指针与函数	121
6.4.1 指针作为函数的参数	121
6.4.2 函数的返回值为指针	122
6.4.3 指向函数的指针	123
6.4.4 命令行参数	124
6.5 动态数组的实现	126
6.5.1 动态内存分配函数	126
6.5.2 一维动态数组的实现	128
6.5.3 二维动态数组的实现	129
6.6 应用举例	129
习题	132
第7章 结构体、共用体和枚举	137
7.1 结构体类型	137
7.1.1 结构体类型的定义	137
7.1.2 结构体变量和数组的定义	138
7.1.3 结构体变量和数组的初始化	140
7.1.4 结构体变量和数组的引用	142
7.1.5 结构体类型的嵌套	146
7.1.6 结构体与函数	147
7.2 共用体类型	148
7.2.1 共用体类型的定义	148
7.2.2 共用体变量和数组的定义	148
7.2.3 共用体变量和数组的引用	149
7.2.4 共用体类型的嵌套	150
7.3 动态数据结构	151
7.3.1 链表的定义	151
7.3.2 动态链表的建立	151
7.3.3 链表的插入操作	152
7.3.4 链表的删除操作	152
7.4 枚举类型	153
7.5 typedef 语句	154

7.6 位段	155
7.7 应用举例	156
习题	161
第8章 编译预处理	163
8.1 宏定义	163
8.1.1 不带参数的宏定义	163
8.1.2 带参数的宏定义	165
8.2 文件包含	166
8.3 条件编译	167
8.4 应用举例	170
习题	171
第9章 文件	173
9.1 文件概述	173
9.2 缓冲文件系统	174
9.2.1 文件类型指针	174
9.2.2 文件打开函数	174
9.2.3 文件关闭函数	175
9.2.4 文件写函数	176
9.2.5 文件读函数	177
9.2.6 文件定位函数	179
9.2.7 其他函数	180
9.3 非缓冲文件	181
9.3.1 文件打开函数、创建函数和关闭函数	181
9.3.2 文件读函数和写函数	182
9.3.3 文件定位函数	182
9.4 应用举例	183
习题	184
第10章 面向对象的程序设计方法及 C++ 简介	187
10.1 面向对象程序设计的基本概念	187
10.1.1 结构化程序设计与面向对象程序设计	187
10.1.2 面向对象程序设计的重要特征	190
10.1.3 从 C 语言到 C++ 语言	191
10.1.4 一个简单的 C++ 程序实例	191
10.2 C++ 对 C 的扩充	192
10.2.1 注释行	192

10.2.2	const 关键字	192
10.2.3	内存管理	193
10.2.4	传递引用	193
10.2.5	内联函数	197
10.2.6	无名共用体	197
10.2.7	带默认形参值的函数	198
10.2.8	枚举类型	198
10.3	C++ 面向对象程序设计的基础	200
10.3.1	类与对象	200
10.3.2	构造函数与析构函数	204
10.3.3	友元	211
10.3.4	const 成员	213
10.3.5	静态成员变量和静态成员函数	214
	习题	217
第 11 章	继承性与派生类	219
11.1	类的层次与继承	219
11.1.1	派生类的声明	220
11.1.2	派生类的 3 种继承方式	221
11.2	派生类的构造函数与析构函数	226
11.2.1	派生类的构造函数	226
11.2.2	派生类的析构函数	228
11.3	名字重载与名字隐藏	230
11.4	多继承与虚基类	233
11.4.1	多继承中的二义性问题	233
11.4.2	虚基类	235
11.4.3	继承与组合	237
	习题	238
第 12 章	多态性	241
12.1	函数重载	241
12.2	运算符重载	243
12.2.1	运算符重载的规则	243
12.2.2	运算符重载函数的 2 种形式	244
12.3	虚函数	246
12.3.1	一般虚函数成员	246
12.3.2	纯虚函数	247
12.3.3	虚析构函数	248

习题.....	250
第 13 章 C++ 的 I/O 流库	253
13.1 输入流格式和输出流格式.....	253
13.2 格式控制.....	254
13.3 重载提取符和插入符.....	255
13.4 文件的输入输出.....	257
13.4.1 文件的打开和关闭.....	257
13.4.2 I/O 成员函数.....	258
13.4.3 文本文件与二进制文件的读写.....	260
13.5 应用实例.....	261
习题.....	265
第 14 章 运行面向对象的程序	267
14.1 VC++ .NET 简介	267
14.2 用 C++ 语言设计一个面向对象的 Win32 控制台应用程序	268
14.3 创建并运行一个 Win32 控制台应用程序	269
附录.....	273
附录 A C 语言的关键字.....	273
附录 B C 语言运算符的优先级与结合性	273
附录 C 常用字符与 ASCII 码对照表	274
附录 D 常用的 ANSI C 库函数.....	276
参考文献.....	283

第 1 章

CHAPTER

C 语言概述

1.1 C 语言简介

C 语言是一种应用广泛的高级程序设计语言。在 30 多年的发展过程中,人们已经利用 C 语言开发了许多大型的系统软件与应用软件。如著名的 UNIX 操作系统 90% 的代码,以及近年来出现的 Visual C/C++ 的编译程序等都是用 C 语言编写的。

由于 C 语言具有“低级语言”的特点,可以实现汇编语言的某些功能,所以 C 语言在开发嵌入式系统中显示出具有速度快、可移植性好和重用性好等优越性能。

C 语言是一种优秀的结构化程序设计语言,它本身也在不断发展。面向对象的概念发展和深化了 C 语言,从而产生了面向对象的 C++ 语言,以及可视化的编程工具 Visual C/C++。

学习 C 语言可以培养用高级程序设计语言解决实际问题的思维方法,以增强程序设计的能力。

本节介绍 C 语言的发展历史和其特点,以使读者对 C 语言有初步的认识。

1.1.1 C 语言的发展历史

C 语言是目前世界上最为流行的计算机高级语言之一。最早是用来描述 UNIX 操作系统及其上层软件的。C 语言的发展与 UNIX 操作系统一样,经历了产生、完善、广泛应用的过程。

C 语言的发展历史可以追溯到 ALGOL 60。1960 年出现的 ALGOL 60 对程序设计语言的设计有较大的影响。它是一种面向问题的高级语言,它与计算机硬件离得太远、太抽象、太笼统。1963 年,英国剑桥大学和伦敦大学推出了 CPL(combined programming language)。CPL 在 ALGOL 60 的基础上比较接近硬件,但规模比较大,难以实现。1967 年,英国剑桥大学的 Martin Richards 对 CPL 进行了简化,推出了 BCPL(basic combined programming language)。它具有结构化好,具有指针处理方式和直接访问

内存地址等特点。1970 年,贝尔实验室的 Ken Thompson 在 BCPL 的基础上,又进一步进行简化,设计出很简单的而且接近硬件的 B 语言(取 BCPL 的第一个字母),并用 B 语言写了第一个 UNIX 操作系统,且在 PDP-11 上实现。但由于 B 语言的一些缺陷(过于简单、功能有限),特别是它的数据无类型,使得在实际应用中遇到很多困难。后来,美国贝尔实验室的 Dennis Ritchie 在参加 PDP-11 的 UNIX 操作系统的开发工作中,对“B 语言”做了进一步的扩充与完善,并吸收了 ALGOL 60 和 Pascal 语言的一些优点,从 1971 年开始,用了一年的时间写了第一个 C 语言编译程序,并于 1972 年投入了使用。

1973 年, Ken Thompson 和 Dennis Ritchie 把 UNIX 操作系统用 C 语言重写了一遍,使 UNIX 系统的代码量比以前的版本增加了 1/3,并增加了多道程序设计功能。由于 C 语言具有良好的可移植性,因而为 UNIX 系统后来的蓬勃发展起到了促进作用。

虽然最初 C 语言是附属于 UNIX 操作系统,并且是在 PDP-11 计算机上实现的,但是现在的 C 语言却独立于 UNIX 操作系统。它的适应机种从 8 位机到巨型机,并与 FORTRAN, Pascal 等语言一样,已经成了微型机到巨型机上所共同使用的语言。UNIX 操作系统的一些主要特点,如便于理解、易于修改及具有良好的可移植性等,在一定程度上得益于 C 语言,所以,UNIX 操作系统的成功与 C 语言是密不可分的。随着 UNIX 的广泛使用, C 语言迅速得到推广。C 语言和 UNIX 操作系统在发展过程中相辅相成,互相促进。

以 1978 年发表的 UNIX V 中的 C 编译程序为基础, Ken Thompson 和 Dennis Ritchie 合著了著名的《The C Programming Language》,标志着 C 语言的诞生,它被称为标准 C。1983 年,美国国家标准局(ANSI)专门成立了委员会,目的是制定“一个没有歧义、与机器无关的 C 语言”,但仍然保留原 C 语言的主体。此项工作于 1989 年完成,该标准 1990 年被接受并作为国际标准,这就是今天所说的 ANSI C。

随着 C 语言的不断发展和其应用的普及, C 语言已经能够在多种操作系统(如 UNIX, DOS, Windows 等环境)下运行。C 语言作为最初的有效的面向过程的编程工具,逐渐被面向对象的编程工具 C++ 所替代,而在嵌入式系统的环境下, C 语言依然发挥着其不可替代的作用。在计算机的主流 PC 机上,由于 Windows 的广泛使用,基于 Windows 编程的 Microsoft Visual Studio 已经成为主流, Visual C++ 是程序员选择的主要编程工具。为了适应微软的 .NET 架构, Visual C++ .NET 是集成在这个平台上的重要的语言编程开发工具。

1.1.2 C 语言的特点

由于 C 语言有很多优点,所以成为目前应用最广泛的高级程序设计语言之一。C 语言的特点大致可以归纳为如下几点:

(1) 语言简练、紧凑,使用方便、灵活

C 语言中一共有 32 个关键字、45 个标准运算符以及 9 种控制语句。程序的书写形式自由,标识符的定义灵活,区分字母的大小写。许多成分都是通过显式函数调用来完成的,因此写出的 C 语言程序非常简练而功能强大。