

C语言 程序设计

本书编委会

浙江科学技术出版社

(计算机类)

世纪高等教育精品大系

浙江省高等教育重点教材

C 语言 程序设计

本书编委会

浙江科学技术出版社

世纪高等教育精品大系

(计算机类)



图书在版编目(CIP)数据

C 语言程序设计/《C 语言程序设计》编委会编. —杭州: 浙江科学技术出版社, 2004.8

(世纪高等教育精品大系)

ISBN 7-5341-2436-0

I. C... II. C... 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 067669 号

丛 书 名	世纪高等教育精品大系
书 名	C 语言程序设计
编 著	本书编委会
出版发行	浙江科学技术出版社
联系电话	(0571) 85152486
印 刷	浙江大学印刷厂
开 本	787×1092 1/16
印 张	21.75
字 数	541 000
版 次	2000 年 2 月第 1 版 2004 年 8 月第 2 版
印 次	2006 年 1 月第 10 次印刷
书 号	ISBN 7-5341-2436-0
定 价	27.00 元
责任编辑	张祝娟
封面设计	孙 菁

本书著作权属于浙江省高校计算机等级考试办公室，我社独家享有本书的图书出版发行专有使用权。

任何单位、组织、个人未经我社和浙江省高校计算机等级考试办公室授权同意，不得擅自翻印、摘录、汇编本书中的有关内容或根据本书编制相关辅导读物。未经授权而有上述行为者，是对我社和浙江省高校计算机等级考试办公室合法权益的侵害，我社将通过法律手段追究其法律责任。

浙江科学技术出版社

《C 语言程序设计》

编 辑 委 员 会

主 任 方永平

副主任 胡维华

委 员 (以姓氏笔画为序)

陈庆章 何钦铭 项小仙 赵建民

俞瑞钊 蒋联海 雷 炜 楼程富

主 编 何钦铭 颜 晖

编著者 颜 晖 陆汉权 杨起帆 何钦铭

前言

程序设计是高校重要的计算机基础课程,它以编程语言为平台,介绍程序设计的思想和方法。通过该课程的学习,学生不仅要掌握高级程序设计语言的知识,更重要的是在实践中逐步掌握程序设计的思想和方法,培养问题求解和语言的应用能力。因此,这是一门以培养学生探索创新精神和实践能力为重点,特色鲜明的课程。

C 语言是得到广泛使用的程序设计语言之一,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,并以其丰富灵活的控制和数据结构、简洁而高效的语句表达、清晰的程序结构和良好的可移植性而拥有大量的使用者。目前,C 语言被许多高校列为程序设计课程的首选语言。

C 语言程序设计是一门实践性很强的课程,该课程的学习有其自身的特点,听不会,也看不会,只能练会。学习者必须通过大量的编程训练,在实践中掌握语言知识,培养程序设计的基本能力,并逐步理解和掌握程序设计的思想和方法。因此,C 语言程序设计课程的教学重点应该是培养学生的实践编程能力,教学模式要从知识传授转为能力培养,其教材也要以程序设计为中心来组织内容。

虽然目前介绍 C 语言的教材很多,也包括一些著名的教材,但在多年教学实践中,发现能较好适合大学教学要求的书并不多。现有的教材一般围绕语言本身的体系展开内容,以讲解语言知识为主,特别是语法知识,辅以一些编程技巧的介绍,不利于培养学生的程序设计能力和语言应用能力。

好的教材源于教学改革和教学实践,能体现教学改革的成果。浙江大学从 1997 年起,就开始实施全方位的程序设计课程的教学改革,目的就是为了培养学生的程序设计能力,以适应新世纪人才培养的需求。经过多年的建设,在教学内容、教学方法、教学手段和考核方式上,已经基本形成一套比较完整的体系,被评为“国家精品课程”,本教材就是在多年的教学实践基础上总结而成的。

全书以程序设计为主线,从应用出发,通过案例和问题引入内容,重点讲解程序设计的思想和方法,并结合相关的语言知识的介绍。全书共 13 章,分为基础篇和提高篇两部分,主要包括 3 方面的内容:基本内容(数据表达、数据处理和流程控制)、常用算法和程序设计风格,以及 C 语言应用中的一些处理机制(编译预处理和命令行参数)。其中基础篇侧重基本知识和基本编程能力,包括数据表达中的基本数据类型、简单构造类型和指针基础,数据处理中的表达式,以及流程控制中的顺序、分支、循环 3 种语句级控制方式和函数的使用这一单位级控制手段。提高篇包括指针和各种构造类型的混合运用,文件的使用,用结构化程序设计思想实现复杂问题的编程,以及基本算法等内容。

在教材的结构设计上,强调实践,使学生从第 1 周起,就开始练习编程,并贯穿始终。在前两章中,简单介绍一些背景知识和利用计算机求解问题的过程,然后从实例出发,介绍顺序、分支和循环 3 种控制结构以及函数的使用,使学生对 C 语言有一个总体的了解,并学习编写简

单的程序,培养学习兴趣。从第3章开始,逐步深入讲解程序设计的思想和方法,说明如何应用语言解决问题。为了提高读者的学习兴趣,对语言知识的介绍一般通过实例程序引入,还将程序设计的技巧、方法,以及编程中的常见错误分散在每节的内容中,以“编程风格”、“提示”和“思考”的形式给出。

基础篇包括第1~第8章,第1章介绍程序与程序设计语言的知识,以及利用计算机求解问题的过程;第2章从实例出发,简单介绍顺序、分支和循环3种控制结构以及函数的使用,以及在实例程序中用到的语言知识;第3章和第4章通过大量的例题,分别讲解分支、循环结构程序设计的思路和方法;第5章将散布在基础篇各个章节中的基本数据类型和常用表达式作了一个归纳汇总,使读者对C语言的数据表达和数据处理方式有一个清楚的认识,并要求读者在编程时能综合运用前5章介绍的编程思想和语言知识;第6章通过3个典型示例介绍了一维数组、二维数组和字符串的应用;第7章介绍指针的基本概念;第8章用案例说明结构类型在编程中的应用。提高篇包括第9~第13章,第9章讲解函数和程序结构方面的深入内容;第10章和第11章分别介绍指针和数组、指针与结构以及其他构造类型的概念和在编程中的应用;第12章介绍文件的使用;第13章介绍贪心法、分治法、回溯法以及动态规划问题等基本算法思想。

在课程教学中,建议将内容分为16个教学单元,基础篇11个单元,提高篇5个单元。第1章和第2章4~5个单元,第3~第13章一般每章一个单元,第13章也可作为阅读内容。

本书编写得到了浙江省高校计算机教学研究会的大力支持。第1章由浙江大学何钦铭教授编写,第2章由浙江大学城市学院颜晖副教授和杨起帆副教授编写,第3、第5和第6章由颜晖副教授编写,第4、第9、第12章以及附录由杨起帆副教授编写,第7、第8、第10、第11、第13章由浙江大学陆汉权副教授编写,全书由何钦铭教授和颜晖副教授主编并统稿。陆蓓、周子立、郭彩虹、唐云廷、胡同森、许巨定等老师参与了本书编写大纲的研讨及书稿的审定工作,提出了许多宝贵的意见,在此一并表示感谢。

本书可以作为各类大专院校、各类培训与等级考试的教学用书,也可作为对C程序设计感兴趣者的自学用书。相信通过本书的学习,能为你打下C程序设计的坚实基础。

另外,与本书教材配套使用的另一本教学或自学参考书《C语言学习及实践指导》将在2005年春季出书,书中将有本教材习题的参考答案和《C语言学习及实践指导》中附加习题的参考答案。此书是教师教学、学生或读者自学非常实用的辅助参考书。

计算机科学和技术在不断发展,计算机教学的研究和改革也从未停顿。希望在从事计算机基础教学的各位同仁的共同努力下,能不断提高C语言程序设计课程的教学质量和水平。

由于作者水平所限,对书中存在的谬误之处,敬请读者指正。

本书编委会

2004年7月

目 录

第一部分 基础篇

第1章 引 言	3
1.1 一个C语言程序	3
1.2 程序与程序设计语言	5
1.2.1 程序设计语言的功能	5
1.2.2 程序设计语言的语法	6
1.2.3 程序的编译与编程环境	9
1.3 C语言的发展历史与特点	9
1.4 实现问题求解的过程	10
习题1	14
第2章 用C语言编写程序	15
2.1 在屏幕上显示Hello World!	15
2.2 求华氏温度100°F对应的摄氏温度	17
2.2.1 程序解析	17
2.2.2 常量、变量和数据类型	17
2.2.3 算术运算和赋值运算	18
2.2.4 格式化输出函数printf()	19
2.3 计算分段函数	20
2.3.1 程序解析	20
2.3.2 关系运算	21
2.3.3 if-else语句	22
2.3.4 格式化输入函数scanf()	23
2.3.5 常用数学库函数	24
2.3.6 else-if语句	25
2.4 输出华氏—摄氏温度转换表	27
2.4.1 程序解析	27
2.4.2 for语句	28
2.4.3 指定次数的循环程序设计	32
2.5 计算12! / (5! + 7!)	35
2.5.1 程序解析	35
2.5.2 函数的定义	37
2.5.3 函数的调用	40

2.5.4 使用函数编写程序	43
习题 2	45
第 3 章 分支结构	47
3.1 输入一批字符, 统计各类字符的数量	47
3.1.1 程序解析	47
3.1.2 字符类型	48
3.1.3 逻辑运算	49
3.1.4 字符输入函数 <code>getchar()</code> 和字符输出函数 <code>putchar()</code>	50
3.2 使用 <code>switch</code> 语句统计字符	51
3.2.1 程序解析	51
3.2.2 <code>switch</code> 语句	53
3.3 分支结构程序设计	55
3.3.1 二分支结构和基本的 <code>if</code> 语句	56
3.3.2 多分支结构和嵌套的 <code>if</code> 语句、 <code>switch</code> 语句	58
习题 3	61
第 4 章 循环结构	62
4.1 用格里高利公式求 π 的近似值	62
4.1.1 程序解析	62
4.1.2 <code>while</code> 语句	64
4.1.3 输入一批学生的成绩, 统计平均分	65
4.2 统计一个整数的位数	66
4.2.1 程序解析	66
4.2.2 <code>do-while</code> 语句	67
4.2.3 循环语句的选择	68
4.3 判断素数	69
4.3.1 程序解析	69
4.3.2 <code>break</code> 语句和 <code>continue</code> 语句	70
4.4 求 $1! + 2! + \dots + 100!$	71
4.4.1 程序解析	71
4.4.2 嵌套循环	72
4.5 循环结构程序设计	75
习题 4	82
第 5 章 数据类型和表达式	85
5.1 数据的存储和基本数据类型	86
5.1.1 数据的存储	86
5.1.2 基本数据类型	87
5.2 常量和变量	89
5.2.1 常量	89



5.2.2 变 量	92
5.3 数据的输入和输出	94
5.3.1 整型数据的输入和输出	94
5.3.2 实型数据的输入和输出	95
5.3.3 字符型数据的输入和输出	97
5.4 类型转换	99
5.4.1 自动类型转换	99
5.4.2 强制类型转换	100
5.5 表达式	101
5.5.1 算术表达式	101
5.5.2 赋值表达式	104
5.5.3 关系表达式	105
5.5.4 逻辑表达式	107
5.5.5 条件表达式	109
5.5.6 逗号表达式	110
5.5.7 其他运算	110
5.5.8 程序解析	111
习题 5	112
第 6 章 数 组	118
6.1 排序问题	118
6.1.1 程序解析	118
6.1.2 一维数组的定义和引用	120
6.1.3 一维数组的初始化	121
6.1.4 使用一维数组编程	121
6.2 找出矩阵中最大值所在的位置	127
6.2.1 程序解析	127
6.2.2 二维数组的定义和引用	128
6.2.3 二维数组的初始化	129
6.2.4 使用二维数组编程	130
6.3 进制转换	133
6.3.1 程序解析	133
6.3.2 一维字符数组	134
6.3.3 字符串	135
习题 6	138
第 7 章 指针基础	142
7.1 指针变量的定义	144
7.2 指针的基本运算	146
7.3 指针变量的初始化	150
7.4 指针作为函数的参数	152



习题 7	156
第 8 章 结 构	161
8.1 使用结构表示学生信息	162
8.1.1 结构变量的定义	163
8.1.2 结构变量的引用	165
8.1.3 结构变量赋值	166
8.1.4 结构的嵌套定义	168
8.2 按成绩从高到低输出学生的信息	169
8.2.1 程序解析	169
8.2.2 结构数组	172
习题 8	173
第二部分 提高篇	
第 9 章 函数与程序结构	179
9.1 函数的组织	179
9.1.1 结构化程序设计方法	180
9.1.2 函数的嵌套调用	183
9.1.3 文件包含	183
9.1.4 函数与程序文件模块	185
9.2 递归函数	186
9.2.1 递归函数基本概念	186
9.2.2 递归程序设计	189
9.3 变量与函数	191
9.3.1 局部变量和全局变量	191
9.3.2 变量生命周期和静态局部变量	194
9.3.3 寄存器变量和外部变量	196
9.3.4 全局变量与程序文件模块	197
9.4 宏定义	197
9.4.1 宏基本定义	197
9.4.2 带参数的宏定义	199
9.5 编译预处理	201
习题 9	202
第 10 章 指针和数组	207
10.1 指针和数组	207
10.1.1 指针、数组和地址间的关系	208
10.1.2 数组名作为函数的参数	210
10.1.3 排序算法: 冒泡排序	213
10.2 指针和字符串	214
10.2.1 常用的字符串处理函数	214
10.2.2 字符串的指针表示	219

10.2.3 字符数组和字符指针	221
10.3 指针数组和指向指针的指针	227
10.3.1 指针数组	227
10.3.2 指向指针的指针	229
10.3.3 指针数组、二维字符数组和字符串	232
*10.3.4 命令行参数	239
10.4 指针和函数	242
10.4.1 指针作为函数的返回值	242
*10.4.2 指向函数的指针	243
习题 10	246
第 11 章 指针和结构及其他抽象数据类型	250
11.1 指针和结构	251
11.1.1 结构指针的概念和使用	251
11.1.2 结构指针作为函数的参数	253
11.2 单向链表	257
11.2.1 单向链表的定义	257
11.2.2 单向链表的常用操作	257
11.3 关于结构的进一步说明	267
*11.4 联合	268
11.4.1 联合的定义	268
11.4.2 联合变量的定义和引用	269
*11.5 枚举	274
11.5.1 枚举类型的定义	274
11.5.2 枚举变量的定义和引用	274
11.6 自定义类型 (typedef)	276
11.7 位运算与位段	278
11.7.1 位运算及位运算符	278
*11.7.2 位 段	280
习题 11	284
第 12 章 文 件	291
12.1 文件的基本概念	291
12.1.1 文本文件和二进制文件	292
12.1.2 缓冲文件系统	292
12.1.3 缓冲文件与文件类型指针	293
12.2 文件的打开与关闭	294
12.2.1 打开文件	294
12.2.2 关闭文件	296
12.3 文件的读写	296
12.3.1 字符文件读写	297



12.3.2 数值文件读写	301
*12.3.3 二进制文件读写	302
12.3.4 文件程序设计	302
习题 12	303
*第 13 章 基本算法	305
13.1 算 法	305
13.2 贪心法	309
13.3 分治算法	311
13.4 动态规划	313
13.5 回溯法	318
附录 1 C 语言上机操作指导	321
附录 1.1 C 语言程序的上机步骤	321
附录 1.2 Visual C 语言集成环境	322
附录 2 ASCII 码集	329
附录 3 C 语言中的关键字	330
附录 4 运算符优先级	331
附录 5 C 语言常用库函数	332

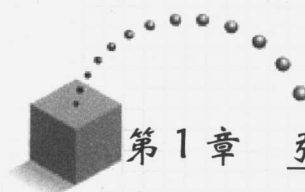
注：*为阅读内容



基

础

篇



第1章 引言

第2章 用C语言编写程序

第3章 分支结构

第4章 循环结构

第5章 数据类型和表达式

第6章 数组

第7章 指针基础

第8章 结构

引言



本章要点:

- ★ 什么是程序? 一般程序设计语言包含哪些功能?
- ★ 一般程序设计语言在语法上包含哪些内容?
- ★ 结构化程序设计有哪些基本的控制结构?
- ★ C语言有哪些特点?
- ★ C语言程序的基本框架如何?
- ★ 形成一个可运行的C语言程序需要经过哪些步骤?
- ★ 如何应用流程图描述简单的算法?

对于将C语言作为第一门编程语言(Programming Language)的读者来说,最关心的问题应该是如何尽快学会用C语言进行程序设计。要做到这一点,我们首先对程序设计语言,特别是C语言本身要有所了解,更重要的是通过不断的编程实践,逐步领会和掌握程序设计的基本思想和方法。

熟练的编程技能是在不断的知识与经验积累的基础上发展起来的。初学编程的人一开始由于缺乏足够的语言知识和编程经验,对于很简单的问题往往也会感到无所适从,不知如何下手编写程序。本书建议你从一开始学习C语言起就要试着编写程序,可以先模仿教科书中的程序,试着改写它并逐步体会、循序渐进,直到会独立地编写程序解决比较复杂的问题。

为了使读者能逐步从简单的模仿中体会程序设计的基本思想和方法,而不拘泥于具体的语法细节,本章作为这本教材的引言将简要介绍程序设计语言的功能、语法要素、C语言的特点以及程序设计求解问题的一般步骤等。

1.1 一个C语言程序

为了让读者对C语言有一个感性认识,首先来看一下用C语言编写的一个程序。

【例 1-1】求阶乘问题。输入一个正整数 n ,输出 $n!$ 。

源程序

```

#include <stdio.h>          /* 编译预处理命令 */
void main()                 /* 主函数 */
{
    int n;                  /* 变量定义 */
    int factorial(int n);    /* 函数声明 */

    scanf("%d", &n);        /* 输入一个整数 */
    printf("%d\n", factorial(n)); /* 调用函数计算阶乘 */
}

int factorial(int n)         /* 定义计算 n! 的函数 */
{
    int i, fact = 1;

    for(i = 1; i <= n; i++) /* 循环 */
        fact = fact * i;
    return fact;
}

```

运行程序时，输入 4，则输出 24，即 4 的阶乘。当然，你可以输入别的数，输出的将是另外一个值。

对于以上程序，不要求初学程序设计的读者一下子就能理解。上述程序中的许多内容将会在随后各章的内容中逐步介绍，但希望初学者能对 C 程序有个初步的印象。

C 程序是由函数（Function，一种子程序）所组成的。前面的程序涉及到 4 个函数：main()、factorial()、scanf()和 printf()。scanf()和 printf()是系统事先设计好的函数，分别用于数据的输入和输出；factorial()是程序中定义的函数，主要目的是求 n!，并将 n 作为函数的参数；main()函数是程序的主函数。

所有的 C 程序都有且只有一个 main()。C 程序都从 main()函数处开始运行，而且 main()函数结束时，程序也就结束了。对于上述例子，程序先执行 main()函数中的 scanf()函数调用，输入数据 n，然后调用 printf()函数，输出结果。当调用 printf()函数时，必须先知道所要输出的数据，即 factorial(n)。因此，此时发生了对 factorial()函数的调用（Call），调用该函数所获得的结果作为 printf()函数的参数（Argument），由 printf()函数负责将该值按十进制整数（%d）输出。

程序最根本的是对数据的处理，为此，需要处理的数据先要将其表示出来。在这个程序的 factorial()函数中，我们用整数型变量（Variable）n 表示要求阶乘的整数，同样在 main()函数中也用 n 来表示（用别的变量名字也可以，如 m）。同时，在 factorial()函数中，用 fact()存储计算的结果，用 i 表示 1 到 n 之间的某个整数。

程序需要对数据处理的过程进行控制。在这里，最主要的控制是将 1 到 n 的每个整数 i 乘到我们所要的结果变量 fact 中。这个控制通过 for 循环语句（Loop Statement）来实现。

