

高校非计算机专业计算机等级考试教材丛书

C语言

程序设计基础

徐君毅
于玉京
吴京

编



复旦大学出版社

高校非计算机专业计算机等级考试教材丛书

C 语言程序设计基础

徐君毅 于 玉 吴 京 编

复旦大学出版社

(沪)新登字 202 号

责任编辑：周仲良

C 语言程序设计基础

徐锡爵 于玉 吴京 编

复旦大学出版社出版

(上海国权路 579 号)

新华书店上海发行所发行 常熟文化印刷厂印刷

开本 850×1168 1/32 印张 9.75 字数 264,000

1992 年 8 月第 1 版 1993 年 3 月第 2 次印刷

印数 5,001—12,000

ISBN 7-309-00861-8/T·53

定价：6.10 元

内 容 提 要

本书是《高校非计算机专业计算机等级考试教材丛书》之一(共6种)。根据考试大纲要求,本书重点介绍C语言基础及其程序设计。主要内容有:基本运算符和表达式、语句、数组、函数、指针、结构、输入输出及文件使用等。

本书以微机上的C为背景,以ANSI C为标准介绍C程序设计,书中所有实例均已在TURBO C环境中运行过。作者考虑到初学者的特点,力求做到全书内容合理安排,阐述深入浅出,因此凡学过一门高级语言的读者都能通过学习掌握本书基本内容。

本书也可作为普通高校有关专业的教材和作为计算机应用工作者的参考书。

高校非计算机专业计算机等级考试 教材丛书编委会

主 任: 施伯乐

委 员 (按姓氏笔划为序)

吴立德 招兆铨 陆盛强

徐余麟 曹文君 钱乐秋

序 言

近年来,我国的计算机应用得到了迅速的发展。计算机已进入到各行各业,产生了巨大的社会和经济效益,已成为实现四个现代化不可缺少的工具。在如今,掌握一些计算机知识有助于事业的成功已成为人们的共识。大批科技人员和管理人员都把学习计算机技术作为知识更新的关键环节。继续教育是重要的,但对广大在校的非计算机专业的大学生开展计算机基础教育更是十分必要,把大学生培养成为既有自己的专业知识又有计算机应用知识的全面人才,对进一步提高整个社会的计算机应用水平有很大的现实意义。

为了进一步促进高校非计算机专业计算机的教学工作,普及计算机应用知识和提高计算机应用能力,上海市高等教育局建立了上海高校非计算机专业学生计算机应用知识和应用能力等级考试制度。该制度针对不同专业对计算机应用知识与应用能力的不同要求,把考试分为三级,即一级、二级、三级。

根据考试委员会颁布的有关第二等级考试的设置目标和考试范围,复旦大学组织力量编写了这套教材丛书。包括:《计算机应用初步》、《TRUE BASIC 语言程序设计基础》、《FORTRAN 语言程序设计基础》、《PASCAL 语言程序设计基础》、《C 语言程序设计基础》、《COBOL 语言程序设计基础》等六种。根据“要求具有使用计算机的基本知识和使用一门高级语言在计算机上编制程序和上机调试的能力”的考试目标,学生必须学完计算机应用初步和一门高级语言程序设计基础(BASIC、FORTRAN、PASCAL、C、COBOL 中任选一门)才能顺利通过第二等级的考试。使用这套教材的教学安排为:应用初步教学 20 学时(含上机),高级语言程序设计基础

教学 72 学时(含上机),共计为 92 学时。

非计算机专业的计算机教学特点是以应用知识和应用能力为主的。本套教材的作者都是从事这方面教学的教师,有较丰富的教学经验,教材中的许多内容直接取自于多年来积累的教学笔记。

本套教材适用于大学理工科非电类专业、计算机应用专业大专、计算机成人教育和继续教育的各类进修班、培训班,以及广大工程技术人员和管理人员。

欢迎广大师生在使用中对本套教材提出批评与改进意见。

施伯乐

1992.4.15

前 言

○是一种通用程序设计语言，早期的○语言被用来书写UNIX操作系统，它是和UNIX操作系统相辅相成地发展的。随着UNIX操作系统在国际上广泛流行，○已为人们普遍接受。目前○已脱离UNIX操作系统，几乎在各种大中小计算机及绝大部分微机上安装了○编译器。○语言已成为最受广大计算机专业工作者欢迎的高级语言之一。

○不仅是一种很好的系统语言，也是一种很好的应用语言。它不仅为计算机专业人员广泛使用，同时也为广大从事计算机应用的人员所使用。近年来，不少高等学校不仅在计算机专业，而且在一些非计算机专业，开设了○语言课程。本书就是为适应非计算机专业的教学要求而编写的一本○语言教材。

在本书的编写过程中，考虑到通用性，全书基本上是根据目前所发表的ANSI ○语言标准编写的，由于读者的对象基本上是非计算机专业的人员，所以对○语言中非常用的一些成分未作详细介绍。另外，由于读者主要使用微机上的C，因此本书的例题均在TURBO C(2.0版)上通过，由于○语言有较好的移植性，在其它○编译系统上，书中大部分例题不加改动也可运行。

本书共分八章。

第一章简单论述○的发展史以及它的特点，还列举了简单的○程序，使读者一开始对○程序有一个概括的了解。

第二章介绍○语言的基本数据类型、运算符和表达式。

第三章介绍○语言所使用的主要语句和控制结构。这两章所介绍的内容都是○语言的基本成分。

第四章介绍○语言中的数组。第五章介绍○中的主要成分

——函数,论述C中的函数结构、函数类型、参数、函数的递归以及C中很有特色的变量存储类,还介绍分程序结构及预处理。

第六章介绍C中最有特色的内容——指针,详细论述指针的基本概念、指针与数组的关系,函数指针以及命令行参数等内容,这一章的内容是学好C语言并付诸实践的关键部分。

第七章介绍结构和联合,包括结构的定义、结构成员的引用、结构数组、指向结构的指针和引用自身的结构等。

第八章介绍输入输出以及文件的使用,这些内容虽然不是C语言成分,但却是编写C程序所必需的知识,一个有用的C程序是少不了这些内容的。

在附录中列举了TURBO C上机实习使用命令及基本库函数。

本书由三人合作编写。第一、二、三、五章由吴京编写,第四、六章由徐君毅编写,第七、八章由于玉编写。全书由徐君毅、于玉定稿,并请蔡子经对全书进行了审阅。

本书是为高校非计算机专业计算机等级考试所编写的一本教材。在编写的过程中得到复旦大学计算机科学系有关专家的帮助,他们为本书的编写提出了不少宝贵意见。复旦大学出版社为本书出版做了大量的工作,使本书能尽早与读者见面,在此一并表示感谢。

由于时间紧迫,以及编者水平所限,疏漏之处难免,恳请读者及专家、同行提出宝贵意见。

编者

1992年3月

目 录

序 言	1
前 言	1
第一章 C 语言概述	1
§1.1 C 语言的历史和特点	1
§1.2 C 语言的基本程序结构	3
§1.3 C 语言程序的编译和运行	6
第二章 数据类型、运算符和表达式	8
§2.1 标识符	8
§2.2 变量说明	9
§2.3 基本数据类型和常量	11
§2.4 类型转换	18
§2.5 表达式与运算符	23
习题	43
第三章 控制语句	46
§3.1 循环语句之一——for 语句	46
§3.2 循环语句之二——while 语句	56
§3.3 循环语句之三——do 语句	59
§3.4 选择语句之一——if 语句	61
§3.5 选择语句之二——switch 语句	71
§3.6 标号语句和转向语句	76
习题	61
第四章 数组	83
§4.1 一维数组的定义和使用	83

	§4.2 二维数组的定义和使用	88
	习题	91
第五章	函数和程序结构	95
	§5.1 函数	96
	§5.2 变量的存储类及作用域规则	119
	§5.3 变量的初始化和分程序结构	126
	§5.4 C 预处理程序	133
	习题	138
第六章	指针	142
	§6.1 指针和地址	142
	§6.2 指针和数组	152
	§6.3 字符指针	160
	§6.4 返回指针的函数	164
	§6.5 指针数组和命令行参数	167
	§6.6 指向函数的指针	173
	习题	184
第七章	结构和联合	183
	§7.1 结构的说明和使用	186
	§7.2 结构数组及指向结构的指针	192
	§7.3 引用自身的结构	214
	§7.4 联合	232
	§7.5 位域和枚举类型	235
	习题	237
第八章	输入输出和文件处理	239
	§8.1 输入输出函数	239
	§8.2 文件的使用	243
	§8.3 输入输出重定向与错误处理	251
	§8.4 文件的随机访问	256
	§8.5 文件与数据流	257
	§8.6 程序实例	258

习题	272
附录一 TURBO C 常用基本命令	274
附录二 TURBO C 实用库函数	232
附录三 C 语言考试大纲	296

第一章 C 语言概述

§1.1 C 语言的历史和特点

一、C 语言的演变历史

C 语言是和 UNIX 操作系统相辅相成地发展起来的，特别在早期的历史中，它们是并肩发展的。C 语言作为一门程序设计语言，其起源可以追溯到 1963 年发表的 OPL 语言 (Combined Programming Language)，甚至更早可以追溯到有名的 ALGOL 60 语言。当时，ALGOL 语言由于过于抽象和一般而未能流行。有鉴于此，剑桥大学和伦敦大学仿照 ALGOL 语言设计了 OPL 语言，试图使之比 ALGOL 语言更实际一点。但是，OPL 语言仍然过于庞大和复杂，难以学习和编程。1967 年，剑桥的 Martin Richards 对 OPL 进行了一番删繁就简的工作，提出了 BCPL 语言 (Basic OPL)，它吸取了 OPL 的一些精华。后来贝尔实验室的 Ken Thompson 对 OPL 做了进一步的简化，于 1970 年设计了 B 语言。它们两人的工作看来是矫枉过正了，因为这两种语言实在过于简单，缺乏通用性。到了 1972 年，Dennis M. Ritchie 主要通过巧妙地使用数据类型的方法，使得他设计的 C 语言恢复了以上两种语言失去的一般性，而这种一般性的恢复并未牺牲以上两种语言所具有的优点，即与实际计算机硬件功能的联系。

1972 年，UNIX 操作系统用 C 语言重新编码，在 PDP-11 机上运行，用 C 语言重新书写的 UNIX 比原先的版本更易于理解、修改和扩充。这对于系统程序是特别有效的，而且用 C 语言书写的 UNIX 版本的系统具有良好的可移植性。

随着 UNIX 系统的广泛流行，C 语言开始为人们普遍接受。

现在，C语言已独立于UNIX系统而迅速发展，在各种类型的计算机系统上大多都配置了C语言。

二、C语言的特点

C语言具有许多优点，主要有以下几个方面。

1. 简洁

C语言很少提供直接处理诸如作为整体来看的字符串、集合、表或数组等复合对象的操作。C语言本身也没有提供输入、输出的手段。另外，C语言的关键字只有30来个，它只需要很少的字符就能描述一些所需的意义。

2. 高效

C语言是一种中级的语言，它的一些操作直接对应于实际计算机硬件所执行的工作。在许多方面它具有汇编语言的特性，因而能充分反映计算机的硬件功能，所以代码效率极高。统计表明，用C语言编写的程序经编译以后所得到的可执行代码仅比用汇编语言直接编写的代码效率低15~20%。所以C语言足以代替汇编语言来编写最讲究效率的那部分软件，比如操作系统、编译程序等。最明显的例证是UNIX操作系统，其早期版本的13000行代码中，只有800行是用汇编书写的，UNIX的上层软件，几乎都用C语言书写。

3. 可移植

虽然C语言本身与计算机的许多硬件能力相适应，但它却是独立于具体计算机体系结构的。因此，很容易书写出便于移植的程序。C语言及UNIX系统的流行与这一特性密切相关。

C语言还有其它优点，在此不尽概述。表1.1是用高级语言和低级语言的观点来衡量C的一些结果的。

表 1.1 用高级语言和低级语言的观点来衡量 C

高级语言	C 语言	低级语言
结构化	有	有
易于使用	有	有
安全性	有/无	有
复杂的数据结构	有	有
		快速性
		紧密性
		高效性
		地址和位运算

§1.2 C 语言的基本程序结构

在详细介绍 C 语言的各种成分以前，我们先介绍 C 语言的几个例子及相应的成分。一方面可以使读者对 C 语言程序有一个大概的了解，另一方面也可以使我们在逐步介绍一些语法规则的时候不至于分散精力再去介绍一些其它的语法成分。下面就来看几个具体的 C 语言程序。

例 1.1

```

/*First program*/
main ( )
{
    printf("\n\nI am now programming in C\n");
}
    
```

C 语言程序完全由一系列函数组成，这些函数可以放在一个文件里，也可以放在多个文件里。这一系列的函数必须有一个且只有一个以 main 为名的函数，这个函数称为主函数，整个程序从它开始执行，并且，一般来说，在它里面结束。

在这个 C 语言程序中，只有一个主函数，函数名 main 后面的一对括号是函数的标志，是不可以省略的。主函数中只有一条语句，printf 是一个格式输出函数，它在屏幕上显示括号内的字符串。在 C 语言中，字符串是用双引号括起来的。反斜线\可与后面跟随的一个字符合并在一起代表一个字符，如\n 表示换行、\t 表示水平制表符。

C语言允许将注解书写在程序的各个部分，注解由斜线后随星号开始，并以星号后随斜线结束。

```
/*注解的内容*/
```

程序中出现的一对花括号，称为程序体括号。用它们括起来的部分是函数体，程序的可执行部分都在函数体内。

C语言中，分号是语句的终结符，因而每一个可执行的C语句后面都必须带有分号。另外，C语言程序中不允许出现行号。

例 1.2

```
/*a plus b*/  
main ( )  
{  
    int a, b, sum; /*declare a, b, sum*/  
    a = 2;  
    b = 10;  
    sum = a + b;  
    printf("a PLUS b is equal to %d.\n", sum);  
}
```

主函数体内的第一条语句是说明语句，它说明 a、b、sum 是整型变量，变量的类型说明在以后几章将详细说明。

主函数体内的最后一条语句，是对 printf 函数的调用，它比例 1.1 中的情况稍微复杂一些，这里出现了两个参数，第一个参数是字符串，第二个是变量 sum。第一个字符串提供格式信息，在“%”后面出现的字符是格式转换，指出输出其它（第二个、第三个、……）参数的格式。而其它参数就是按指定格式输出的内容。字符串内其它字符，按原样打印。本例中的格式转换 %d 表示下一个参数（即第二个参数 sum）将作为一个十进制数打印出来。C语言是自由格式的语言，只要符合语句的形式，多条语句可以在一行中连续书写，一条语句也可以写在几行。例如，例 1.2 可以写成如下形式而功能不变。

```
/*a plus b*/
```



```
main() {int a, b, sum; /*declare a, b, sum*/ a=2; b=10;
sum=a+b; printf("a PLUS b is equal to%d. \n",
sum);
}
```

但是, 这样的写法不够清晰, 不便于阅读和修改。

例 1.3

```
/*sum of any two integers*/
main ( )
{
    int a, b, sum; /*declare a, b, sum*/
    printf("type a number please! a=?");
    scanf("%d", &a);
    printf("type another please! b=?");
    scanf("%d", &b);
    sum=a+b;
    printf("%d PLUS %d is equal to %d. \n",
a, b, sum);
}
```

函数 scanf 类似于 printf, 它按第一个参数指定的格式输入

表 1.2 常用格式转换

格式串	printf 中相应的参数输出形式	scanf 中相应的参数输入形式
%d(%1d)	十进制(长)整数	十进制(长)整数
%u(%1u)	十进制无符号(长)整数	十进制无符号(长)整数
%o(%1o)	八进制(长)整数	八进制(长)整数
%x(%1x)	十六进制(长)整数	十六进制(长)整数
%f	十进制浮点数	十进制浮点数
%lf		十进制双精度浮点数
%g	同 %f, 除去不必要的 0	
%c	单个字符	单个字符
%s	字符串	字符串