



C 语言教程

何以南 编著

312
V11

电子科技大学出版社

TP312
HYN/1

《电脑报》职业教育教材丛书之八

C 语言教程

何以南 编著

电子科技大学出版社

3029446

[川]新登字 016 号

责任编辑 黎和生
封面设计 李光宇
版式设计 谢应成

JS364/L3

C 语言教程

何以南 编著

电子科技大学出版社出版

(成都建设北路二段四号) 邮编 610054

电脑报社照排部排版

重庆日报印刷厂印刷

四川省新华书店经销

*

开本 787×1092 1/16 印张 11.25 字数 270 千字

版次 1995 年 5 月第一版 印次 1995 年 5 月第一次印刷

ISBN 7-81043-374-1/TP·139

定价: 8.50 元

前 言

随着科学技术与现代社会的发展,电子计算机的应用越来越广泛,一个普及计算机知识的高潮正在兴起。普及计算机知识,推动计算机应用的发展,提高全民族文化素质,是当今计算机教育工作者的神圣使命。

近年来,国内计算机职业教育蓬勃兴起,为国家培养了大批计算机应用方面的初级专业人材,同时,也进一步促进了职业教育自身的高速发展,逐渐走向专业化、正规化。为了适应计算机职业教育发展的需要,重庆市教委职教处,重庆市教科所职教室及电脑报社组织具有丰富教学经验的特级教师、高级教师和计算机专家,编写了这套《电脑报》计算机职业教育教材丛书。

本丛书严格按照计算机职业教育的教学大纲要求,根据职业教育注重实际操作技能的特点,从实际出发,介绍了如何使用计算机的方法和与此有关的必要的理论知识。本丛书的内容包括:计算机的基本原理、磁盘操作系统、文字处理方法、计算机语言、数据库管理、基本工具软件、电子排版技术、网络基本原理、计算机英语等。丛书的叙述方法为:深入浅出,循序渐进,通俗易懂,注重实践,每章附有小结和习题,并根据课程要求,配有与授课内容相宜的上机操作手册,作为学生上机练习的指导读物,以加深理解并掌握使用计算机的技能。

该丛书现为十册,分别是:《计算机基础教程》、《磁盘操作系统》、《BASIC 语言教程》、《中文信息处理技术》、《FOXBASE 数据库原理与应用》、《基本工具软件及其应用》、《电子排版技术基础》、《C 语言教程》、《NOVELL 网使用与维护》、《计算机专业英语》,可作为计算机职业高中、中专、技校及各类培训班的教材,也可供电脑爱好者作为自学读本。

随着计算机应用的不断发展,对计算机职业教育的要求越来越高,我们还将根据职业教育的发展需要,推出其他内容的教材,希望广大读者向我们提出建议,反映要求,我们将努力满足广大读者的愿望。

《C 语言教程》是本套丛书的第 8 册,由何以南编著,何宗琦副教授审稿。

《电脑报》职业教育教材丛书编委会

1995 年 5 月

内 容 简 介

本书全面系统地介绍了C语言的基本语法规则和编程技巧。作者结合自己多年从事C语言教学的经验,根据C语言的特点,把基本概念、语法规则、程序设计技巧和上机操作方法有机地结合在一起,全书条理清晰、概念明确、例题详尽,以深入浅出的语言迅速把读者带进C语言大门。

本书适合职业学校及大专院校作为教材,也可作为电脑爱好者的学习辅导书。

《电脑报》职业教育教材丛书编委会名单

主 任	牟维坤
副主任	包锦安 陈宗周
	(以下按姓氏笔画排列)
委 员	向才毅 何以南 何宗琦
	魏嗣富 谭元颖

目 录

第一章 C语言概述	1
第一节 C语言发展与特点	1
第二节 C语言的程序的构成	2
第三节 格式化输入输出函数 scanf 和 printf	4
第四节 C语言程序的编辑、编译、连接和执行	8
本章小结	9
习题一	9
实验一	10
第二章 C语言数据类型、运算符和表达式	11
第一节 标识符与变量	11
第二节 基本数据类型和常量	12
第三节 数据类型转换	15
第四节 运算符表达式	17
本章小结	26
习题二	27
实验二	29
第三章 C语言程序设计	31
第一节 复合语句	31
第二节 条件语句	32
第三节 开关分支(switch)语句	36
第四节 循环语句	39
第五节 break continue 和 goto 语句	46
第六节 应用举例	51
本章小结	53
习题三	54
实验三	55
第四章 数组	57
第一节 一维数组	57
第二节 二维数组	60
第三节 字符数组与字符串	63
本章小结	68
习题四	68
实验四	69
第五章 函数	72
第一节 函数的定义和调用	73
第二节 变量的作用域	78
第三节 编译预处理	84

本章小结	88
习题五	88
实验五	92
第六章 指针	93
第一节 指针	93
第二节 指针与数组	97
第三节 指针作函数参数	103
第四节 指针与字符串	104
第五节 指针与函数	108
第六节 指针数组和命令行参数	111
本章小结	114
习题六	115
实验六	117
第七章 结构体与共用体	118
第一节 结构体类型及其变量的定义和引用	118
第二节 结构体数组	123
第三节 结构体与函数和结构体与指针	127
第四节 位段	133
第五节 共用体和类型定义	135
本章小结	138
习题七	139
实验七	139
第八章 文件	141
第一节 文件的概念和文件类型指针	141
第二节 文件的使用	143
第三节 文件的随机读写与重定向	157
本章小结	160
习题八	160
实验八	160
附录一 Turbo C 2.0 函数列表	162

第一章 C 语言概述

第一节 C 语言的发展与特点

一、C 语言的发展史

C 语言最早是用来描述 UNIX 操作系统及其上层软件的。C 语言和 UNIX 是并肩发展的。剑桥大学和伦敦大学在 ALGOL 语言的基础上于 1963 年设计了 CPL (Combined Programming Language) 语言,其缺点是过于庞大和复杂,难以学习和编程。剑桥的 Martin Richards 对 CPL 语言进行了一番删繁就简的工作,于 1976 年发表了 BCPL 语言。1970 年 UNIX 的研制者 Ken Thompson 应用汇编语言和称为 B 的语言,发表了在 PDP-7 上实现的 UNIX 的初版,B 语言是他对 BCPL 语言的发展和改进。BCPL 语言和 B 语言都过于简单,缺乏通用性。

1972 年,美国 BELL 研究所的 Dennis Ritchie 采用数据类型的方法,对 B 语言进行了充实和完善,推出了一种新型程序设计语言——C 语言。

1983 年美国国家标准局(ANSI)语言标准化委员会对 C 语言问世以来的若干发展,补充和修改订正后,公布了一个 C 语言标准草案(83 ANSI C),1987 年又推出了 87 ANSI 标准 C。

随着 UNIX 系统的广泛流行,C 语言已独立于 UNIX 操作系统而为人们普遍接受,并迅速发展。它已成为微小、小、超小、大、超大和原型机上共同使用的语言。

二、C 语言的特点

C 语言的优点很多,主要体现在以下几个方面。

1. 语言表达能力强

C 语言同时兼有高级语言和低级语言的特点,是介于高级语言和低级语言之间功能很强的中级程序设计语言。它具有算术及逻辑运算功能,能直接处理字符的地址。它能充分反映计算机的性能,很多系统软件和应用软件都是用 C 语言编写的。例如 UNIX 操作系统,除 10% 的程序是用汇编语言编写外其它均是用 C 语言编写的。

2. 模块化结构

C 语言是一种模块化的程序设计语言。C 语言的程序是由一系列函数构成的,这些函数可以是自己设计的,也可以是标准库函数提供的。C 语言程序是由最顶层的函数 main() 按照一定的组织层次调用标准库函数或自己设计的函数而形成。

3. 具有较强的数据类型构造能力

C 语言除了四种标准数据类型(字符型、整型、浮点型和双精度型)外,还具有构造类型:数组、结构、联合、枚举等类型。在这些类型的基础上还可以用 typedef 语句定义新的数据类型名。

4. 语言简洁

在 C 语言中占有 32 个标准的关键字和 45 个操作字,因此,它是一种小型语言,但功能却很强。它的表示方法尽量简单,例如,用 {} 表示开始与结束,用 ++ 表示加 1,-- 表示减 1 等。C 本身没有提供 I/O 设施,许多功能都是通过显式函数调用来实现。

5. 可移植

C 语言之所以广泛流行,这与其编写的程序可移植密不可分。C 语言本身不依赖于硬件系

统,从而使得 C 程序很容易从一种计算机环境上不加修改或稍加修改就可以移植到与该计算机硬件环境不同的其它计算机系统上。

6. 代码质量高

C 语言的一些操作直接对应于实际计算机硬件所执行的操作,在很多方面具有汇编语言的特性,因此能充分反映计算机硬件的功能,所生成的代码率极高。针对同一个问题,用 C 语言来描述,通常其代码效率只比用汇编语言描述低 10%—20%。由于 C 语言编程比汇编语言编程迅速,可读性也较之好,而效率降低不大,且 C 语言具有较好的可移植性,所以 C 语言成了系统软件和应用软件开发的理想工具。

第二节 C 语言程序的构成

在讲 C 语言特点时我们讲过 C 语言程序是由一系列函数组成的。我们先看下面的例子:

例 1.1 在屏幕上显示 "this is a pen".

C 程序为:

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
    printf("This is a pen ");
```

```
}
```

在本程序中,main()称为主函数,main 为函数名,函数名之后是一对圆括号,圆括号内是写函数参数的,本程序的 main 函数没有参数。但圆括号不能省略,它是函数的标志。

main()后面用“{”和“}”括起来的部分叫函数体。在函数体中一般由两部分组成。一个部分是“说明部分”,另一部分是“执行部分”。说明部分用来定义数据类型,执行部分给出做何种操作。本例中没有说明部分,执行部分也只有一个语句,即调用输出的库函数 printf(...).

printf("This is a pen\n");中的分号(;)是语句结束符。C 语言规定语句结束的必须用分号,不能省略。

本例中的“\n”表示换一行,它是 C 语言规定的特殊符号。

程序开头的“/* This is the main program */”是程序的注释,用来帮助阅读和理解程序的。

通过对这个例子的学习,我们可以说:

一、C 语言程序是由一系列函数构成的,C 语言程序本身就是一个函数,即 main()主函数。一个可执行的 C 语言程序总是从 main()函数开始的。

二、main()函数的函数体是从其“{”开始,到“}”结束的部分。

三、函数体中有说明部分和执行部分。

四、执行部分可以是 C 语言语句,也可以是库函数或自己设计的函数调用语句。

五、函数体内的每一个语句均应以分号(;)结束。

六、C 程序中可以有“/*”到“*/”表示的注释,以帮助程序的阅读和理解。在编译时,注释行被忽略掉,并不产生代码行。

由于 C 语言程序的构成是一系列函数,以 main()为主函数,所以 C 语言程序的一般形式

为:

```
全局变量说明
main()
{
    局部变量说明;
    语句序列;
}
sub1()
{
    局部变量说明;
    语句序列;
}
sub2()
{
    局部变量说明;
    语句序列;
}
...
subn()
{
    局部变量说明;
    语句序列;
}
```

全局变量是指在整个程序运行期间均起作用的变量,局部变量是指在本函数内起作用的变量,离开了本函数即被释放,这些我们将在第五章作详细介绍。sub1()……subn()是用户自己定义的函数。

例 1.2

```
#include <stdio.h>
main()
{
    int a,b,c;
    a=2;
    b=10;
    c=a+b;
    printf("a plus b is equal to %d\n",c);
}
```

本程序中,/* a plus b */表示注释,说明本程序是求 a 加 b 的值的。

main()为主函数,没有函数参数“{ ”到“ }”部分为函数体,函数体中 int a,b,c 是说明部分,说明变量 a,b,c 是整型变量。

“a=2”到“printf(“a plus b is equal to %d\n”,c);”部分称为执行部分,分别将 a、b、c 赋了

值 2,10,12。并在屏幕上显示:a plus b is equal to 12。

本例中%d 转换符,表示十进制整数。下节将作详细介绍。

第三节 格式化输入输出函数 printf 和 scanf

C 语言的标准库函数中提供了两个很有用的格式化输入输出函数 scanf 和 printf,它们为编程时实现各种输入输出提供了方便。

一、格式化输出函数 printf

1. 格式化输出函数 printf 的格式

printf(控制字符串,输出项 1,输出项 2);

2. printf 函数的功能

按照控制字符串将输出项进行转换,并按控制字符串中指定的格式在标准输出设备上将输出项输出。

3. 控制字符串

控制字符串中包含两种字符:一种是普通字符,将原样输出;另一种是格式转换符,规定输出项按格式转换符所在位置和格式转换符指定的格式输出。在第二节例 2 中 printf("a Plus b is equal to %d\n",c); "a plus b is equal to %d\n" 为控制字符串,其中 a plus b is equal to 均为普通字符("\n"是转换字符,将在以后介绍),将原样输出,而"%d"是格式转换符,它规定:输出项 C 的值必须转换成十进制整型输出,并在"to"之后输出。

表 1.1 常用格式转换

格式转换符	printf 中输出项相应输出
%d(%ld)	十进制(长)整数
%u(%lu)	十进制无符号(长)整数
%o(%lo)	八进制(长)整数
%x(%lx)	十六进制(长)整数
%f	十进制浮点数
%g	同%f,除去不必要的 0
%c	单个字符
%s	字符串

输出项将按格式转换符规定的格式输出。如下列:

例 1.3

```
#include <stdio.h>
```

```
main()
```

```
{  
    printf("%d\n",25);
```

```

printf("%x\n",25);
printf("%o\n",25);
printf("%f\n",(float)25);
printf("%d %o\n",25,25);
}

```

程序的执行结果是:

```

25
19
31
25.000000
25    31

```

十进制数 25 的十六进制表示是 19,八进制表示是 31。

在 % 与转换符中间可以加入左对齐标号(减号-),加了减号表示左对齐,未加表示右对齐。例如:printf("%-10s","student\n");printf("%10s\n","student");

该语句输出字符串"student"时,前者按左对齐格式输出,后者按右对齐格式输出:

```

student
      student

```

在 % 与转换中间加数字表示域宽,例如,前面的 10 表示字符串占 10 个字符位置。printf("%9.4f\n",23.43678);语句输出为:

```

23.4368

```

表示浮点数 23.43678 占 9 个字符位置,其中小数占 4 位,由于小数部分位数不能精确表示 23.4368,将最后一位之后的那位按四舍五入输出,因此输出的是 23.4368。由于在 % 与转换符之间进行输出,因此输出的是 23.4368。由于在 % 与转换符之间不带负号,故输出是在域宽范围内右对齐的。

例:

```

main()
{
printf("%10s\n","they");
printf("%-10s\n","they");
printf("%9.4f\n",2345678);
printf("%-9.4f\n",2345678);
printf("%0.3f\n",23.45678);
printf("%f\n",23.45678);
}

```

运行结果为:

```

      ↓第10个字符位置
they

```

they
23.4568
23.4568
23.457
23.456780

当没有域宽时,输出 6 位小数。

二、格式输入函数 scanf

1. scanf 函数格式为:

scanf(控制字符串,地址 1,地址 2,...)

2. scanf 函数功能:

scanf 函数是用来输入数据的,它按照控制字符串的要求从键盘上把数据传送到地址所指定的内存空间中。

3. 几点说明:

(1)控制字符串由两部分组成:一部分是各地址所要接收的数据格式转换符,另一部分是地址所要接收数据时的分隔符。在 scanf 函数中所使用的格式转换符与 printf 函数中所使用的格式转换符基本相同。

例如 scanf("%d\t%f\t%c\n",&a,&b,&c);语句中"%d\t%f\t%c\n"是控制字符串,其中……%d,%f,%c 是格式转换符,分别规定 &a,&b,&c 三个地址所接收的数据依次为整型、实型、字符型数据,而\t和\n规定从键盘输入的数据依次以跳格、回车作为输入数据的分隔符。在控制字符串中也可省略分隔符。当控制字符串中省略分隔符时,可以输入空格、回车作为分隔符。

(2)在一个变量名前冠以"&"符号,表示该变量在内存中所分配的首地址,输入的数据通过缓冲区分送到该所在内存的首地址开始的存储单元,简单地说是变量获得了数据。如前面我们所说的&a,&b,&c,表示的是变量a,b,c所在的地址单元。这些地址单元分别是从&a,&b,&c开始的。

例 1.4

```
#include <stdio.h>
```

```
main()
```

```
{  
    int a;  
    float b,c;  
    scanf("%d%f%f",&a,&b,&c);  
    printf("a=%d,b=%f,c=%f\n",a,b,c);  
    scanf("%ds%fs%f",&a,&b,&c);  
    printf("a=%d,b=%f,c=%f\n",a,b,c);  
}
```

```
}
```

运行中输入:

12 4 56

输出为:

a=12,b=34.000000,c=56.000000

又输入:12 S 34 S 56

又输出:a=12, b=34.000000, c=56.000000

第一个 scanf 函数的控制字符省略分隔符,输入时可用空格为分隔符,第二个 scanf 的数据控制字符串指定的 S 作为其输入数据的分隔符。&a,&b,&c 分别表示变量 a,b,c 在内存的首地址。

(3)在输出浮点数时,如不说明输出小数的位数,则一律按 6 位小数输出。可由前例看出。也可在 scanf 函数中指定该数所占的字符位置以及小数部分的位数如:%10f 表示输入的十进制,浮点数至少占 10 个字符位置,%0.3f 表示输入 3 位小数,输出的也按 scanf 函数指定格式输出。

例 1.5

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
    int a,b;
```

```
    float c;
```

```
    scanf("%d-%d-%0.3f",&a,&b,&c);
```

```
    printf("a=%d,b=%d,c=%f\n",a,b,c);
```

```
}
```

运行时输入:

251-32-673756.321

结果是:

a=251,b=32,c=673856.321

例 1.6 圆柱体体积的计算程序

```
main()
```

```
{
```

```
    int r,h;
```

```
    float v;
```

```
    scanf("%d%d",&r,&h);
```

```
    v=3.14159*r*r*h;
```

```
    printf("v=%f\n",v);
```

```
}
```

解此例题。

第四节 C 语言程序的编辑、编译、连接和执行

一个编写好的 C 语言源程序不象 BASIC 源程序那样边解释边执行,它必须经过编辑、编译、连接几个步骤转化成二进制的目标代码,经连接后才能在 DOS 提示符下执行。下面分别对这几个步骤作一简单的介绍。

一、编辑

编辑要完成三个方面的内容:

1. 将源程序输入计算机内存。
2. 将输入的程序用编辑键进行修改。
3. 将修改好的程序存盘。

编辑好的程序是以 ASCII 码形式存储的。

二、编译

编译是将编辑好的程序翻译成二进制的目标代码,生成 obj 为后缀的二进制代码文件。经编译后得到的二进制代码还不能直接运行,还需要进行连接。

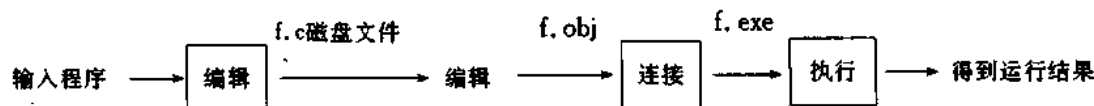
三、连接

因为编译得到的二进制代码是将每一个模块单独编译的,所以必须把经过编译的各个模块的目标代码与 C 语言系统提供的标准模块连接后才能运行。所谓连接就是将各模块的二进制代码与系统标准模块进行连接处理,得到可执行文件,该文件的后缀为 .exe。

四、执行

在操作系统提示符下输入文件名即可运行。

以上四个步骤可用下图表示。



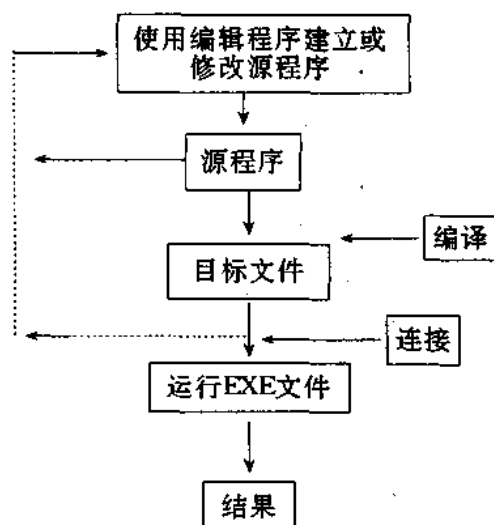
下面在 Turbo C 集成开发环境下说明要运行一个程序进行的操作过程。

1. 在 DOS 提示符下键入 tc, 进入 Turbo C。
2. 进入主 tc 屏, 按 F3 键, 在出现的框中输入文件名, 输入的文件名可以有后缀 .C, 也可缺省, 缺省时系统自动加上后缀 .C。然后按回车键进入编辑状态。这时可输入程序, 修改程序。
3. 当程序输入、修改完毕, 可按 F2 将编辑好的程序存盘。
4. 按 CTRL-F9 进行编译, 连接和执行, 这三步自动连续完成。如果程序有错, 屏幕上会提示错误, 可回到编辑状态进行修改。再重新按 CTRL-F9 进行编译、连接和执行, 直到得到

正确结果。要观察运行结果可按 ALT-F5。如果要退出 TC 返回 DOS 可按 ALT-X。

在源程序处于编辑状态时,当编辑完毕也可以根据菜单提示选择 Compile 菜单(按 ALT-C)中的 COmpile to OBJ 命令,则系统就会产生一个文件名与源程序文件名相同但后缀名为“obj”的目标文件。然后再选择 compile 菜单的 Make EXE file 或 Link EXE file 命令对编译产生的 OBJ 文件进行库程序连接装配,最终生成一个文件与源程序文件名相同、后缀名为“exe”的可执行文件。

用户在 turbo C 上建立程序,进行编辑、编译、连接的过程为:



以上操作也可通过屏幕菜单选择操作来完成。详见其它有关书籍。

本章小结

本章介绍了 C 语言的发展过程以及 C 语言的特点, C 语言程序的构成, 数据的输入输出, 如何进行格式转换和一个 C 程序经过编辑、编译、连接和执行过程而得到结果, 如何在 Turbo c 环境运行程序。

习题一

1. C 语言是如何发展起来的?
2. C 语言有什么特点?
3. C 语言程序的主要构成部分有哪些?
4. 写出下面程序的运行结果。

```
(1)main()  
{  
    printf(" * * * * * \n");  
    printf(" very good;\n");  
}
```

```

printf(" * * * * * \n");
}
(2) main()
{
printf("%d\n",37);
printf("%o\n",37);
printf("%x\n",37);
printf("%c\n",'a');
}

```

5. 编一个程序,从键盘输入 a, b, c, d 四个整数,计算下列式子:

$$(a+b-c) * d$$
并显示计算结果。
6. 运行一个 C 语言程序应经过哪几个步骤,各个步骤的作用是什么?

实验一

实验目的:

1. 掌握在 Turbo c 环境下如何编辑、编译、连接和运行一个 C 语言程序。
2. 了解 C 程序的构成及特点。
3. 正确使用格式转换符和输入输出函数。

实验步骤:

1. 在操作系统提示符下,键入 TC,进入 TC 主屏,按 F3 输入文件名后回车。
2. 输入例 1.1 程序。
3. 按 ctrl-F9 进行编译、连接和运行,并按 F2 存盘。
4. 按 Alt-F5 观看结果。
5. 输入并运行例 1.2,例 1.3,例 1.4,例 1.5,例 1.6,并检查结果。
6. 输入并运行自己编写的习题一中第 5 题的程序。

实验报告:

1. 例 1.1 编辑,编译和连接后你得到了哪些文件? 其后缀分别是什么?
2. 当编译,连接有错误而得不到结果时你是如何处理的?
3. 例 1.6 运行后你输入的是什么? 运行结果是什么?