

TURBO C 语言的 操作使用

何丁海 王盛邦 编



《职业技术培训丛书》
计算机培训教材



621

广东科技出版社

职业技术培训丛书
计算机培训教材

TURBO C 语言的操作使用

何丁海 王盛邦 编

广东科技出版社

粤新登字 04 号

内 容 简 介

本书分为九章，内容由浅入深，引用实例对 C 语言的基本语法，包括数据类型、运算符、语句、函数和程序设计及程序的运行方法，Turbo C (V2. 0) 的集成开发环境的使用作了较完整的介绍。每章均配有练习题，适用于程序语言培训、个人自学和非计算机专业大中专学生的教材或参考书。

图书在版编目 (CIP) 数据

Turbo C 语言的操作使用/何
丁海编著·—广州：广东科技出版
社，1994. 8.

ISBN7—5359—1280—X

I. T…

II. 何…

III. Turbo C 语言，操作

IV. TP31

Turbo C 语言的操作使用

编著者：何丁海 王盛邦

出版发行：广东科技出版社

(广州市环市东路水荫路 11 号)

经 销：广东省新华书店

印 刷：广东新华印刷厂

规 格：787×1092 1/16 印张 14.25 字数 300 千

版 次：1994 年 8 月 第 1 版

1994 年 8 月 第 1 次印刷

印 数：1—10 2000 册

ISBN 7—5359—1280—X

TP·33 定 价：10.20 元

《职业技术培训丛书》

计算机培训教材

编辑委员会

主任委员

丁有豫

副主任委员

何丁海

委员

邓达基

李柏松

余正方

顾问

张益新

段 薇

周国添

(按姓氏笔划排列)

编 者 的 话

日新月异的现代科学技术对国民经济各个领域的发展产生了深刻的影响，计算机的迅猛发展使计算机应用进入了各行各业乃至千家万户之中。学习和掌握计算机的有关知识，已成为时代的需要，成为现代文明的标志。在不远的将来，不会应用计算机将沦为“现代文盲”。江泽民总书记号召广大干部要跟上时代步伐，学习现代科学技术基础知识，其中一项重要内容就是要各级干部掌握和学会使用计算机。

目前，各类计算机培训班如雨后春笋般涌现，但着眼于实用性和操作性，适用于短期培训班的普及性计算机教材不多。为适应这种形势的需要，我们组织编写了这套计算机培训教材。这套教材的特点属普及性教材，内容深入浅出，突出实用性和操作技能训练。课时安排设置适合短期培训要求，每章都附有大量练习题。是各类计算机短期培训班、工人上岗培训、干部计算机培训、大、中专非计算机专业学生课程的理想教材，也适用于广大计算机应用人员和个人自学。这套教材得到劳动部职业技能开发司的肯定，李亨业司长并为这套教材作序，广东省劳动局培训处推荐。

计算机培训教材编辑委员会

1994. 4

序

经济建设呼唤人才！时代培育人才造就人才！技术工人也是人才，而且是社会主义现代化建设不可缺少的人才！培养和造就千百万觉悟高、纪律严、技能精、作风硬的技术工人，是开发人力资源，发展和完善劳动力市场的一项重要任务。

在改革开放、引进技术、发展经济的大潮中，我国的职业技术培训工作得到迅速的恢复和发展，成绩显著。广东的职业技术培训工作同全国一样发展很快，特别是通过贯彻国务院批准、劳动部颁布的《工人考核条例》、广东省人民政府颁发的职业技术培训、考核的有关规定，实行“办学社会化、考核规范化证书统一化”，建立起职业技术培训点3 000多个，基本形成多渠道、多工种、多层次的培训网络。为适应十四大提出的建立社会主义市场经济体制的需要，加速劳动力市场的培育，他们逐步实行职业技能鉴定社会化管理，培训和考核分开，通过技能鉴定站的统一考核，提高培训质量，保证职业技能资格证书的权威性，使持证进入劳动力市场的劳动者名副其实达到国家制定的有关职业技能标准的要求。而统一考核标准，统一考核大纲，统一考核命题并编写与之配套的培训教材，是技能鉴定社会化、管理科学规范化，使之公开、公正、公平的重要条件。为此，广东省劳动局从实际出发组织编写了职业技能培训丛书，为社会上当前比较热门的工种培训提供教材服务。这些丛书按照各工种技术等级标准应知应会的要求，在精讲专业基本理论知识的同时，突出操作技能训练内容，通俗易懂，深入浅出，适合短期培训用，是职业技能培训的好教材。

随着职业技能培训事业的发展，我们需要更多适应职业技能的培训教材，我希望有更多更好的培训教材问世。



1993. 11. 29

前 言

Turbo C 是在 IBM-PC 及其兼容机上实现的一个高效、优化的 C 语言编译程序,它提供的集成开发环境,集 C 程序的编辑、编译、连接、调试和运行于一体。

C 语言由于它的语言结构好、移植性好、应用性强,表示方法简洁等优点,倍受软件工程技术人员的喜爱。许多流行的系统软件如 UNIX 操作系统, dBASE、FoxBASE 数据库管理系统等用它编写。

本书共分九章,第一、二章介绍 Turbo C 的基本操作方法,第三至第六章介绍 C 语言的基本语法、第七至第九章介绍 Turbo C 的应用。此外,还有附录 A、B、C、D,介绍 Turbo C 常用的一些操作。本书侧重从如何操作使用 Turbo C 语言的角度论述,是读者学习、掌握 Turbo C 语言的入门教材和参考资料。

本书第一至第七章和附录由何丁海编写,第八、九章由王盛邦编写。广东省计算机培训教材编委会对本书作了全面地审稿,作者对此表示感谢。

由于我们水平所限,欢迎读者对本书不当之处提出宝贵意见。

何丁海 王盛邦

1993. 12 于中大计算中心

目 录

前 言

第一章 Turbo C 的初步认识	(1)
§ 1. 1 C 语言的由来及其基本特征	(1)
1. 1. 1 C 语言的由来	(1)
1. 1. 2 PC 机上的 C 语言和 Turbo C	(1)
1. 1. 3 C 语言的特点	(1)
§ 1. 2 C 语言程序示例	(2)
1. 2. 1 一个 C 程序的例子	(2)
1. 2. 2 C 程序的一般形式	(3)
§ 1. 3 C 语言的词法和关键字	(4)
1. 3. 1 标识符	(4)
1. 3. 2 关键字	(4)
§ 1. 4 格式化输出和输入函数初步	(5)
1. 4. 1 输出函数 printf()	(5)
1. 4. 2 输入函数 scanf()	(7)
第一章 操作练习和思考题	(9)
第二章 Turbo C 集成开发环境的使用方法	(10)
§ 2. 1 Turbo C 的安装和启动	(10)
2. 1. 1 Turbo C 2.0 系统文件	(10)
2. 1. 2 在软盘或硬盘上运行 Turbo C	(14)
§ 2. 2 Turbo C 的编辑命令	(16)
2. 2. 1 光标移动命令	(16)
2. 2. 2 插入和删除命令	(16)
2. 2. 3 块操作命令	(16)
2. 2. 4 查找定界符命令	(17)
2. 2. 5 其他编辑命令	(17)
§ 2. 3 Turbo C 的功能菜单	(17)
2. 3. 1 TC 主屏幕和热键	(18)
2. 3. 2 菜单功能	(20)
§ 2. 4 程序的上机过程	(25)
2. 4. 1 C 程序运行所经过的三个文件	(25)
2. 4. 2 程序的上机操作步骤	(25)
第二章 操作练习和思考题	(29)
第三章 数据类型和运算符	(30)
§ 3. 1 基本的数据类型	(30)

3. 1. 1	整型数据	(30)
3. 1. 2	浮点型数据	(30)
3. 1. 3	字符型数据	(32)
§ 3. 2	数据类型的扩充	(32)
3. 2. 1	基本整型的扩充	(32)
3. 2. 2	long 与 short 的区别	(33)
§ 3. 3	变量的类型说明	(34)
3. 3. 1	变量说明的一般形式	(35)
3. 3. 2	变量说明注意的问题	(35)
3. 3. 3	数据类型说明的重新定义	(36)
3. 3. 4	枚举类型 (enum)	(36)
§ 3. 4	运算符	(36)
3. 4. 1	算术运算符	(36)
3. 4. 2	关系运算符和逻辑运算符	(37)
3. 4. 3	位运算符和移位运算符	(38)
3. 4. 4	表达式	(40)
3. 4. 5	赋值运算符	(41)
3. 4. 6	加 1 和减 1 运算符	(42)
3. 4. 7	逗号运算符和条件运算符	(43)
3. 4. 8	指针运算符和地址运算符	(44)
3. 4. 9	强制类型转换运算符	(45)
3. 4. 10	sizeof() 运算符	(45)
3. 4. 11	运算符的优先级别	(46)
第三章	操作练习和思考题	(47)
第四章	程序语句	(48)
§ 4. 1	表达式语句、复合语句和空语句	(48)
4. 1. 1	表达式语句	(48)
4. 1. 2	复合语句和空语句	(48)
§ 4. 2	条件语句和多分支语句	(50)
4. 2. 1	条件语句	(50)
4. 2. 2	多分支选择语句	(53)
§ 4. 3	循环语句	(56)
4. 3. 1	while 语句	(56)
4. 3. 2	for 语句	(57)
4. 3. 3	do-while 语句	(59)
4. 3. 4	break、continue 和 goto 语句	(59)
第四章	操作练习和思考题	(63)
第五章	自定义函数、变量的存贮类别 和预处理命令	(66)
§ 5. 1	函数的定义	(66)
§ 5. 2	函数的调用	(67)

5. 2. 1	函数调用的返回及返回值	(68)
5. 2. 2	函数的调用方式	(69)
§ 5. 3	变量的作用域	(74)
5. 3. 1	自动存贮变量	(74)
5. 3. 2	静态存贮变量	(75)
5. 3. 3	外部存贮变量	(78)
5. 3. 4	寄存器存贮变量	(81)
5. 3. 5	变量的初始化	(82)
5. 3. 6	静态变量的应用——伪随机数的产生	(82)
§ 5. 4	预处理命令——程序的插入和常量的置换	(83)
5. 4. 1	文件包含——程序的插入	(83)
5. 4. 2	宏替换命令——常量的置换	(85)
5. 4. 3	宏替换的取消	(88)
5. 4. 4	条件编译	(88)
第五章	操作练习和思考题	(90)
第六章	组合数据和指针的用法	(96)
§ 6. 1	数组的定义及其使用	(96)
6. 1. 1	一维数组	(96)
6. 1. 2	多维数组	(99)
§ 6. 2	结构与联合	(100)
6. 2. 1	结构说明和结构变量的初始化	(100)
6. 2. 2	结构变量的引用	(102)
6. 2. 3	结构数组	(103)
6. 2. 4	联合	(105)
§ 6. 3	指针的用法	(107)
6. 3. 1	指针的运算	(108)
6. 3. 2	指针与数组	(109)
6. 3. 3	指针应用于函数	(113)
6. 3. 4	指针应用于结构	(115)
6. 3. 5	main()的参数	(117)
第六章	操作练习与思考题	(118)
第七章	库函数的用法	(124)
§ 7. 1	输入/输出库函数	(125)
7. 1. 1	字符及字符串输入输出函数	(125)
7. 1. 2	文件输入输出函数	(127)
§ 7. 2	字符串和字符库函数	(135)
7. 2. 1	字符串库函数	(135)
7. 2. 2	字符库函数	(140)
§ 7. 3	数学库函数	(143)
7. 3. 1	三角函数和双曲线函数	(143)
7. 3. 2	指数和对数库函数	(144)

7. 3. 3 其他数学库函数	(145)
§ 7. 4 时间、日期和系统连接的库函数	(147)
7. 4. 1 时间、日期函数	(147)
7. 4. 2 DOS 与 BIOS 接口函数	(151)
§ 7. 5 目录函数、内存管理函数和程序控制库函数	(155)
7. 5. 1 目录函数	(155)
7. 5. 2 内存管理函数	(157)
7. 5. 3 程序运行控制函数	(163)
§ 7. 6 其他库函数	(164)
7. 6. 1 数学类函数	(165)
7. 6. 2 数和字符串转换函数	(166)
7. 6. 3 快速分类及数制转换函数	(167)
7. 6. 4 错误信息显示及执行 DOS 命令函数	(168)
第八章 C 与 dBASE, FoxBASE 的数据交换	(171)
§ 8. 1 dbf 的库结构	(171)
§ 8. 2 C 与 dBASE、FoxBASE 共享文本文件	(172)
8. 2. 1 将 dbf 文件转换为文本文件	(172)
8. 2. 2 由 Turbo C 产生的文本文件转换为 dbf 文件	(175)
§ 8. 3 Turbo C 读取 dbf 文件	(177)
第八章 操作练习和思考题	(183)
第九章 图形函数和其它应用	(184)
§ 9. 1 图形功能	(184)
§ 9. 2 在西文下汉字的使用	(190)
§ 9. 3 使用在线汇编	(197)
§ 9. 4 设计音乐界面	(199)
第九章 操作练习和思考题	(201)
附录 A 程序编译的错误信息	(202)
附录 B 命令行程序 TCC.EXE 的使用	(204)
附录 C 工程文件的建立	(207)
附录 D 充分利用系统的帮助 (Help) 文件	(209)
参考文献	(212)

第一章 Turbo C 的初步认识

本章概要地介绍 Turbo C 语言, 包括 C 语言的由来及其基本特征, C 语言程序示例和 Turbo C 的关键字、词法以及格式化输入/输出函数。

§ 1. 1 C 语言的由来及其基本特征

1. 1. 1 C 语言的由来

1970 年, 美国 Bell 实验室正式推出至今仍流行的操作系统 UNIX。UNIX 是用汇编语言写的, 为解决 UNIX 本身的可读性和移植性问题, K. Thompson 和 D. M. Ritchie 等寻求用高级语言改写 UNIX。他们先在 BCPL 语言的基础上形成了 B 语言, 但因 B 的数据类型单调等原因, B 语言没有流行起来。D. M. Ritchie 改进了 B, 加入了丰富的数据类型, 增强了控制结构, 从而形成了 C 语言。1973 年, K. Thompson 和 D. M. Ritchie 用 C 语言改写了 UNIX 并扩充了 UNIX 的功能。所以, C 和 UNIX 是紧密联系一起的。目前 C 和 UNIX

都相当流行, 图 1. 1. 1 展示了 C 语言的产生过程, 由图可知, C 语言吸收了 B 等几种高级语言的优点, 它属于 ALGOL 语言的体系结构。C++ 是一种面向对象的语言, 是近年来在 C 语言基础上发展起来的新一代高级语言。

ALGOL60

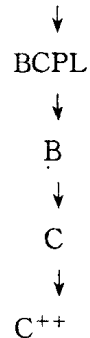


图 1. 1. 1

1. 1. 2 PC 机上的 C 语言和 Turbo C

自从 PC 机及其操作系统 MS-DOS 流行起来后, 于八十年代中, C 语言被移植到 PC 机上。目前我国较为流行的 PC 机 C 语言版本有 Lattice C、High C、Zortech C 及 C++, Quick C 等, 但主要还是 Borland C 和 Microsoft C 两个系列版本。其中又以 Borland 的 Turbo C 较为流行。

1983 年, ANSI (美国国家标准协会) 为 C 语言制定了 ANSI 标准。Turbo C 是完全按照 ANSI 的 C 语言标准实施的。Turbo C 为 C 语言程序的编辑、编译、连接、调试和运行等提供一个集成开发环境; 此外, 按传统运行程序的方法为 C 提供一个命令行编译程序版本, 以满足不同用户的要求。

1. 1. 3 C 语言的特点

C 语言的基本特点是:

(1) C 语言和 Pascal 语言一样, 很强调程序的结构化、简洁和易读性。此外, C 还很

强调程序的模块化，一个C程序可分成若干独立的模块，每个模块为一个函数。

(2) C具有低级语言（汇编语言）的效率，这是因为C允许对位、字节和地址这三个计算机最基本工作单元进行操作。

(3) C语言体系比较小，例如标准C的关键字只有32个，而PC机上的BASIC语言，其关键字共159个。

(4) C具有丰富的数据类型，几乎可定义任意类型的数据。同时，C的运算符多，可提高程序的运算能力。

(5) C语言编写的程序可移植性强，这是C最重要的一个特点。

C有其不足的地方，诸如有的运算符易产生副作用、类型转换随便、没有动态数组界的检查等。这些缺点与C的优点相比，是微不足道的。

§ 1. 2 C语言程序示例

1. 2. 1 一个C程序的例子

假定已知 $a=11$, $b=12$, $c=13$, $d=14$ 。现求 $s=(a+b)*c/d$ ，且显示输出 s 的结果。用C语言编写的程序C_{1_2_1.c}如下：

```
1. #include <stdio.h>
2. main( ) /* count s=(a+b)*c/d */
3. {
4.     int a =11, b=12;
5.     float c, d, s;
6.     c=13; d=14;
7.     s=(a+b)*c/d;
8.     printf("s=%f\n",s);
9. }
```

C_{1_2_1.c}称为C的一个源程序，说明如下：

①C程序本身无行号，但编译出错时会告诉出错语句的行号。这里的行号是为了说明问题而加上的。

②main()为C语言的主函数，它是一个C程序的主体部分，一个可运行的C程序有且仅有一个main()函数；/*与*/之间的内容为注释，本例为：计算 $s=(a+b)*c/d$ ，它对程序只起说明作用，不影响程序的执行效果。

③花括号{和}是函数体的界限符，这里“{”表示函数体的开始符，“}”为函数体的结束符。{和}也可作为复合语句的界限符。

④第4行说明a、b为整型变量且同时分别赋初值11、12。

⑤第5行说明c、d、s为单精度浮点变量。

⑥第6行分别对c、d赋值为13、14。

⑦第7行先计算表达式 $(a+b) * c/d$ ，然后将结果赋给变量s。

⑧第8行的printf()为格式输出函数，它是C系统提供的一个库函数。

⑨每个语句后面加上“;”表示语句结束符，C程序一个程序行可多个语句，但每个语句末必须为“;”。

最后，第一行的#include为预处理命令(见第五章)，stdio.h为标准I/O头部文件，#include <stdio.h>的作用是将printf()等标准I/O库函数嵌入C源程序。

C1_2_1.c源程序经编译，连接后生成一个.exe文件，如执行得：

s=21.357142

1. 2. 2 C程序的一般形式

一个C程序可由一个或多个函数组成，一个可执行的C程序中的主函数main()可出现在其它函数的前后或者函数之间，不管main()出现在程序的什么地方，程序的执行总是从main()的第一个花括号“{”开始，至最后一个“}”结束。主函数可调用其它函数，其它函数也可互相调用，且可递归调用(即自己调用自己)。

一个C程序可以文件形式存放在磁盘上，称为C源程序文件，其扩展名为.C。

图1. 2. 1为C程序的一般形式

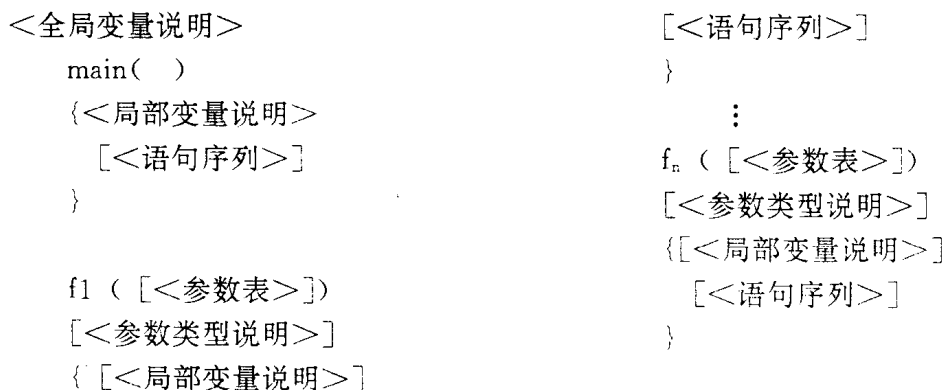


图1. 2. 1 C程序的一般形式

有关全局变量，局部变量和函数请参见第五章。

一个程序由多个函数组成时，可将这个程序的所有函数以一个程序文件存放，也可分开存放几个文件中，对于大的C程序，提倡把一个程序按其不同的子功能，分开若干个函数编写，同时提倡把不同函数存放于不同文件中。这有利于程序的编辑、修改、调试。现举两个特殊的C程序的例子。

(1) 最简单的C程序

```
f( )  
{  
}
```

这个程序仅一个函数，而这个函数既无参数也无语句，但它是一个C程序，调用此函数时，不做任何事情，即返回调用它的函数。

(2) 显示一句话的程序

```
main( )  
{  
    printf("How do you do ! \n");  
}
```

这个程序在屏幕上显示:

How do you do !

§ 1. 3 C 语言的词法和关键字

在 C 中,组成语句的语法单位是单词,单词包括:标识符、常量、字符串、运算符、分隔符和关键字。其中分隔符指空格、制表符、换行符和注释等。常量、字符串和运算符在后面的章节介绍。本节就标识符和关键字分述如下。

1. 3. 1 标识符

标识符用来表示变量、函数、数组等由用户自定义函数的名称。

标识符是以字母或下横线开头,由字母、数字和下横线等组成的字符串,字符串的长度为1~32个B。Turbo C 允许使用 \$ 字符作标识符的字符,但 \$ 为非标准字符,建议一般不用。

下面的字符串是正确的标识符:

abc, a123, high_line, a, B, ala, _bq, a838

下面的字符串不能用作标识符:

3ab	(以数字开头)
abc!	(!非字母、数字和下横线)
ab.123	(.非字母、数字和下横线)
a_b	(_非字母、数字和下横线)
x&y	(&非字母、数字和下横线)

值得注意的是大、小写字母用作标识符时是有区别的,例如:abc, Abc, aBc 是三个不同的标识符。另外,不能用关键字和已定义的函数名作为标识符。

1. 3. 2 关键字

关键字又称保留字,它是 C 语言命令(语句)的关键词,是语句的特殊成分,一般出现在语句的开头,表明该语句的性质、作用等。C 语言的关键字使用小写字母。如

if、for 是,而 IF、FOR 则不是由 ANSI 标准推荐的 C 语言关键字。

表1. 3. 1

Turbo C 关键字

由 ANSI 标准推荐的32个关键字			
<u>auto</u>	double	int	struct
<u>break</u>	else	long	switch
<u>case</u>	enum	register	typedef
<u>char</u>	extern	return	union
const	float	short	unsigned
<u>continue</u>	for	signed	void
<u>default</u>	goto	sizeof	volatile
<u>do</u>	if	static	while
Turbo C 扩充的关键字			
asm	_cs	_ds	_es
_ss	cdecl	far	huge
interrupt	near	pascal	

Turbo C 扩充的关键字:

asm 在 C 程序中定义汇编命令, 例 asm mov al, 1a

far (远程)、near(近程)、huge(特大)、_cs, _ds, _es, _ss 为寻址标识符, 例: int _es * ptr; ptr 为放附加段的16位偏移量。

cdecl 一般用于函数修饰符, 即用于函数和指向函数的指针。

pascal 与 cdecl 类似

interrupt (中断)用于函数修饰符

【例1. 3. 1】 C_{1_2_1.c}程序中的语句

```
int a, b;
```

int 为关键字, 用于说明整型变量, a、b 为变量名。

§ 1. 4 格式化输出和输入函数初步

C 程序中数据的输出和输入通常用格式化输出函数 printf() 和输入函数 scanf()。这两个函数定义在头部文件 stdio.h 中, 程序中使用这两个函数前要用预处理命令

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

才能将 printf() 或 scanf() 嵌入源程序中。

1. 4. 1 输出函数 printf()

(1) 举例

程序 C_{1_4_1.c}

```

#include <stdio.h>                                /* 将 printf( ) 嵌入源程序 */
main( )
{
    int a=123;                                     /* 说明 a 为整型变量, 且赋初值为123 */
    float f=1.123456;                             /* 说明 f 为浮点型变量, 且赋初值为
                                                    1.123456 */
    char c='a';                                    /* 说明 c 为字符型变量, 且赋初值 'a' */
    char *s="abc";                                 /* s 为指针, *s 取字符串"abc"(参见 § 6.3
                                                    节) */
    printf("a=%d %d\n",a,321);                    /* 显示输出: a=123及321且换行 */
    printf("%f %.5f",f,1.123456);                /* 显示输出: 1.123456和1.12346不换行 */
    printf(" %c A=%s\n",c,s);                     /* 显示输出:  a A=abc */
}

```

程序C1_4_1.c执行后在屏幕上显示输出:

```

a=123      321
1.123456  1.12346  a  A=abc

```

上述 printf() 中的 %d、%f、%c、%s 为格式转换字符, 分别用于转换控制输出为整型数、浮点型数、单个字符、字符串等; \n 用于控制换行。

① 可控制输出小数的位数

用 %f 转换时, 定义输出小数位为6位; 用 %.nf (n 为正整数) 控制输出为 n 位小数。超过输出位数时, 四舍五入。

例如: printf("%f %7f", 1.1234567, -1.1234567);

输出: 1.123457 -1.1234567

② 可控制输出长度

例如: printf("%5d %05d", 123, 321);

输出: 123 00321

注意: 当给出的控制长度小于要输出数据长度时不起作用。例: printf("%6.3f", 12.3456);

输出: 12.346

③ 按指数形式输出

例如: printf("f1=%f f2=%e", 12.3, 12.3);

输出: f1=12.300000 f2=1.230000e+01

④ 用 l 控制输出

例如: long int a=1234567890; (a 为长整型, 参见表3, 2, 1)

printf("a=%ld\n", a);

输出:

```
a=1234567890
```

(2) printf() 的一般格式

printf("转换控制字符串", <输出参数表>);