



C语言实例解析 精粹

曹衍龙 编著
林瑞仲
徐 慧

精选近200个实例

提供全部实例源代码

汇集C语言各种编程技巧

前言

当今的计算机软件设计中,无论开发技术如何发展,C语言作为一种基本的程序语言,仍然是程序开发人员必须掌握的基本功。掌握了C语言,不但可以对结构化的编程有全面的了解,而且能够深入理解操作系统的运作方式、内存管理分配方式以及硬件编程控制方法等。

目前流行的许多开发工具,包括微软的 Visual C++和 Visual C++ .NET, Borland 公司的 C++ Builder 等开发工具都还遵循着标准 C 语言的基本语法。在很多嵌入式系统的软件设计中,甚至都采用 C 语言来进行开发。

为什么写本书

对于 C 语言的学习者来说,最重要的是具备程序设计的能力。初学者往往能读懂别人编写的代码,而自己编写程序时却无所适从,往往不清楚通过程序语言的控制结构如何将简单的计算步骤串联起来完成一项复杂的计算。提高程序设计能力的一个重要途径是学习别人编写的程序,从中掌握解决问题的核心方法和关键步骤,循序渐进,直至自己能够找出算法并编写程序。本书正是为了满足 C 语言学习者的这种需求而策划的,这也是书名“C 语言实例解析精粹”的由来。

本书特色

书中以近 200 个实例的编程求解为线索,突出对例题的分析、对知识点的归纳、对求解方法的引申,力图通过对例题的详细分析,帮助学习者快速提高 C 语言的“程序设计能力”。

主要内容

基础篇——介绍了 C 语言编程的基础知识,包括第一个程序的建立,基本数据类型的转换,数组、函数、指针、文件、结构、联合的使用等。

数据结构篇——介绍了冒泡排序、堆排序、归并排序等各种排序算法,顺序表、双链表、二叉树、图的建立和操作等。

数值计算和趣味数学篇——数值计算部分包括多项式求值,线性方程求解以及矩阵运算等。趣味数学部分主要介绍了一些经典问题的求解,包括绘

制余弦曲线，计算高次方数的尾数，求解阿姆斯特朗数，歌德巴赫猜想，素数幻方，爱因斯坦的数学题，三色球问题等。

图形篇——介绍如何使用 Turbo C 提供的图形函数，绘制基本的直线、圆弧，设置屏幕颜色、线条类型、填充类型；绘制直线类图形、金刚石、飘带、肾形、心脏形、沙丘等图案；绘制正多边形、递归三角形、抛物样条曲线等；实现图形变换，VGA 编程，分形图，动画设计等。

系统篇——介绍如何通过 Turbo C 中的系统调用函数，编写屏幕窗口程序、获取系统各类信息程序以及硬件参数读取程序等。

常见试题解答篇——介绍一些常见的 C 语言考试试题的解答方法，比如水果拼盘问题、计算方差、统计符合特定条件的数、字符串倒置、部分排序、产品销售记录处理、求解三角方程、统计选票、数字移位等。

游戏篇——介绍了 DOS 环境下的 C 语言游戏编程，包括商人过河游戏、吃数游戏、解救人质游戏、打字训练游戏、双人竞走游戏、迷宫探险游戏、迷你撞球游戏、模拟扫雷游戏、推箱子游戏、五子棋游戏。

综合实例篇——介绍了综合 CAD 系统、功能强大的文本编辑器、图书管理系统和进销存管理系统。

书中所有实例均在 Turbo C 2.0 或 Borland C++ 3.1 编译器中调试通过。如果某些程序需要显示汉字，可以通过 Visual C++ 或 Visual C++ .NET 编译运行；如果在 DOS 下运行，需要先运行汉字环境，比如 UCDOS。具体的使用方法参见书中的“光盘使用说明”以及光盘中的相关文档。

本书在编写过程中，参考了《The C Programming Language》等书籍和 Internet 上的相关资源，在此对相关的作者和机构深表谢意。

技术支持

本书主要由曹衍龙、林瑞仲编写，参加写作的人员还有吴越、徐慧、续瑞瑞、张静、程立、吴阳等。在写作过程中，我们力求精益求精，但难免存在一些不足之处，恳请读者批评指正。如果您在使用本书时遇到问题，可以发 E-mail 至 zjulinruizhong@yahoo.com.cn 和 tangqian@ptpress.com.cn 与我们联系。

编 者
2005 年 3 月

光盘说明

为了方便读者学习,本书附带了一张光盘,下面对光盘内容及使用方法进行简要的介绍。

光盘的运行环境

硬件环境: CPU的主频在 500MHz 以上、内存在 128MB 以上。

软件平台: 操作系统为 Windows 98/Me/NT/2000/XP (推荐使用 Windows 2000/XP), 调试环境为 Turbo C 2.0 (部分需要显示中文的程序, 调试环境为 Visual C++ 6.0 及其以上版本)。

光盘主要内容

1. **【Code】** 目录, 包括实例 1~190 的所有源代码、可执行程序、相应的数据文件、使用说明 (readme) 等。
2. **【Code】→【VC】** 目录, 包含采用 Visual C++ 编译的实例程序的 Visual C++ 项目文件和可执行程序。
3. **【Tools】** 目录, 包括 Turbo C 2.0 编译器。

光盘的使用方法及注意事项

将相应的实例源代码, 比如 1.c 文件 (在 **【001】** 目录下) 拷贝到 **【C:\tc】** 目录下, 去掉其只读属性 (否则可能无法正常使用这些源程序)。

提示: 可以直接双击 **【Code】** 目录下的 MYCOPY.BAT 文件, 将 190 个实例目录下的所有文件都拷贝到 **【C:\tc】** 目录下。这时, 只有在 TC 编译器的安装目录为 **【C:\tc】** 时才有效, 需要使用者特别注意。

在 MS-DOS 方式下运行 TC.exe, 打开 Turbo C 编译器, 按 **【F3】** 键或者执行 “File→Load” 菜单命令, 打开 1.c 文件, 如图 1 和图 2 所示。

按 **【Ctrl+F9】** 键, 或者执行 “Run→Run” 菜单命令, 编译运行该程序, 如图 3 所示。若有相应的 .dat 文件, 应将 .dat 文件一起拷贝到 **【C:\tc】** 目录下。

查看程序运行结果, 可以按 **【Alt+F5】** 键, 或者执行 “Run→User screen” 菜单命令, 如图 4 和图 5 所示, 若返回 Turbo C 环境, 按回车键即可。

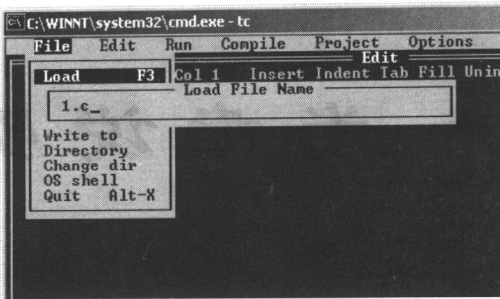


图 1

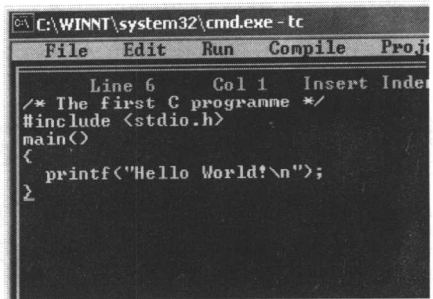


图 2

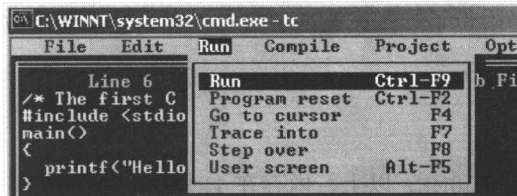


图 3

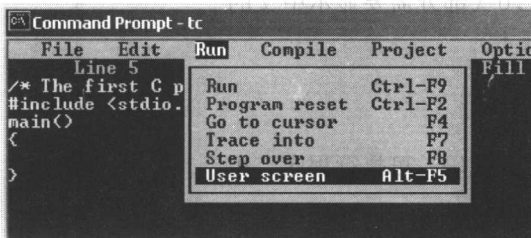


图 4

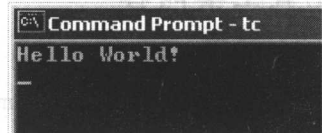


图 5

运行 Visual C++ 项目的方法是，将整个实例目录拷贝到硬盘某个目录下，用右键单击该目录，选择【属性】命令，将该目录及其子目录、文件的【只读】属性取消，在 Visual Studio 6.0 中双击打开该目录下的 .dsw 文件，按【F7】键，生成 .exe 文件，即可在【Debug】目录下运行可执行程序。

技术支持

读者在光盘使用过程中，如果发现差错或者遇到问题，请发 E-mail 至 linread@163.com 和 tangqian@ptpress.com 与我们联系。

目 录

第一部分 基础篇

实例 1 第一个 C 程序	2	实例 21 通过指针比较整数大小	44
实例 2 求整数之积	6	实例 22 指向数组的指针	48
实例 3 比较实数大小	8	实例 23 寻找指定元素的指针	50
实例 4 字符的输出	10	实例 24 寻找相同元素的指针	52
实例 5 显示变量所占字节数	11	实例 25 阿拉伯数字转换为罗马数字	53
实例 6 自增/自减运算	13	实例 26 字符替换	56
实例 7 数列求和	14	实例 27 从键盘读入实数	57
实例 8 乘法口诀表	17	实例 28 字符行排版	59
实例 9 猜数字游戏	19	实例 29 字符排列	60
实例 10 模拟 ATM (自动柜员机) 界面	22	实例 30 判断字符串是否回文	62
实例 11 用一维数组统计学生成绩	24	实例 31 通讯录的输入输出	63
实例 12 用二维数组实现矩阵转置	26	实例 32 扑克牌的结构表示	68
实例 13 求解二维数组的最大/最小元素	29	实例 33 用“结构”统计学生成绩	69
实例 14 利用数组求前 n 个质数	31	实例 34 报数游戏	72
实例 15 编制万年历	33	实例 35 模拟社会关系	73
实例 16 对数组元素排序	36	实例 36 统计文件的字符数	74
实例 17 任意进制数的转换	37	实例 37 同时显示两个文件的内容	80
实例 18 判断回文数	39	实例 38 简单的文本编辑器	81
实例 19 求数组前 n 元素之和	41	实例 39 文件的字数统计程序	82
实例 20 求解钢材切割的最佳订单	42	实例 40 学生成绩管理程序	85

第二部分 数据结构篇

实例 41 插入排序	96	实例 42 希尔排序	100
------------------	----	------------------	-----

实例 43	冒泡排序	102	实例 52	二叉树遍历	130
实例 44	快速排序	105	实例 53	浮点数转换为字符串	132
实例 45	选择排序	109	实例 54	汉诺塔问题	133
实例 46	堆排序	111	实例 55	哈夫曼编码	135
实例 47	归并排序	115	实例 56	图的深度优先遍历	138
实例 48	基数排序	119	实例 57	图的广度优先遍历	139
实例 49	顺序表插入和删除	123	实例 58	求解最优交通路径	141
实例 50	链表操作	126	实例 59	八皇后问题	143
实例 51	双链表	129	实例 60	骑士巡游	145

第三部分 数值计算与趣味数学篇

实例 61	绘制余弦曲线和直线的迭加	150	实例 80	求 π 的近似值	173
实例 62	计算高次方数的尾数	151	实例 81	奇数平方的有趣性质	175
实例 63	钓鱼还是晒网	151	实例 82	角谷猜想	176
实例 64	怎样存钱以获取最大利息	154	实例 83	四方定理	177
实例 65	阿姆斯特朗数	155	实例 84	卡布列克常数	178
实例 66	亲密数	156	实例 85	尼科彻斯定理	179
实例 67	自守数	157	实例 86	扑克牌自动发牌	180
实例 68	具有 $abcd=(ab+cd)^2$ 性质的数	158	实例 87	常胜将军	181
实例 69	验证歌德巴赫猜想	159	实例 88	搬山游戏	182
实例 70	素数幻方	161	实例 89	兔子产子 (菲波那契数列)	183
实例 71	百钱百鸡问题	163	实例 90	数字移动	184
实例 72	爱因斯坦的数学题	164	实例 91	多项式乘法	186
实例 73	三色球问题	165	实例 92	产生随机数	189
实例 74	马克思手稿中的数学题	166	实例 93	堆栈四则运算	190
实例 75	配对新郎和新娘	167	实例 94	递归整数四则运算	196
实例 76	约瑟夫问题	168	实例 95	复平面作图	199
实例 77	邮票组合	169	实例 96	绘制彩色抛物线	200
实例 78	分糖果	170	实例 97	绘制正态分布曲线	203
实例 79	波瓦松的分酒趣题	172	实例 98	求解非线性方程	206

实例 99	实矩阵乘法运算.....	209	实例 103	求定积分.....	220
实例 100	求解线性方程.....	211	实例 104	求满足特异条件的数列.....	221
实例 101	n 阶方阵求逆.....	215	实例 105	超长正整数的加法.....	222
实例 102	复矩阵乘法.....	219			

第四部分 图形篇

实例 106	绘制直线.....	226	实例 124	正方形螺旋拼块图案.....	251
实例 107	绘制圆.....	230	实例 125	图形法绘制圆.....	252
实例 108	绘制圆弧.....	231	实例 126	递归法绘制三角形图案.....	254
实例 109	绘制椭圆.....	232	实例 127	图形法绘制椭圆.....	255
实例 110	设置背景色和前景色.....	233	实例 128	抛物样条曲线.....	257
实例 111	设置线条类型.....	235	实例 129	Mandelbrot 分形图案.....	259
实例 112	设置填充类型和填充颜色.....	237	实例 130	绘制布朗运动曲线.....	261
实例 113	图形文本的输出.....	238	实例 131	艺术清屏.....	262
实例 114	金刚石图案.....	240	实例 132	矩形区域的颜色填充.....	263
实例 115	飘带图案.....	241	实例 133	VGA256 色模式编程.....	265
实例 116	圆环图案.....	242	实例 134	绘制蓝天图案.....	266
实例 117	肾形图案.....	243	实例 135	屏幕检测程序.....	267
实例 118	心脏形图案.....	244	实例 136	运动的小车动画.....	268
实例 119	渔网图案.....	245	实例 137	动态显示位图.....	269
实例 120	沙丘图案.....	246	实例 138	利用图形页实现动画.....	270
实例 121	设置图形方式下的文本类型.....	246	实例 139	图形时钟.....	271
实例 122	绘制正多边形.....	248	实例 140	音乐动画.....	274
实例 123	正六边形螺旋图案.....	249			

第五部分 系统篇

实例 141	读取 DOS 系统中的国家信息.....	278	实例 145	读取磁盘文件.....	284
实例 142	修改环境变量.....	279	实例 146	删除目录树.....	286
实例 143	显示系统文件表.....	280	实例 147	定义文本模式.....	287
实例 144	显示目录内容.....	282	实例 148	设计立体窗口.....	290

实例 149 彩色弹出菜单.....	292	实例 153 备份/恢复硬盘分区表.....	297
实例 150 读取 CMOS 信息.....	293	实例 154 设计口令程序.....	298
实例 151 获取 BIOS 设备列表.....	294	实例 155 程序自我保护.....	300
实例 152 锁住硬盘.....	295		

第六部分 常见试题解答篇

实例 156 水果拼盘.....	304	实例 167 求解三角方程.....	320
实例 157 小孩吃梨.....	305	实例 168 新完全平方数.....	321
实例 158 删除字符串中的特定字符.....	306	实例 169 三重回文数.....	323
实例 159 求解符号方程.....	307	实例 170 奇数方差.....	324
实例 160 计算方差.....	308	实例 171 统计选票.....	326
实例 161 求取符合特定要求的素数.....	309	实例 172 同时整除.....	328
实例 162 统计符合特定条件的数.....	310	实例 173 字符左右排序.....	329
实例 163 字符串倒置.....	312	实例 174 符号算式求解.....	331
实例 164 部分排序.....	314	实例 175 数字移位.....	333
实例 165 产品销售记录处理.....	316	实例 176 统计最高成绩.....	334
实例 166 特定要求的字符编码.....	318		

第七部分 游戏篇

实例 177 商人过河游戏.....	338	实例 182 迷宫探险游戏.....	349
实例 178 吃数游戏.....	340	实例 183 迷你撞球游戏.....	351
实例 179 解救人质游戏.....	341	实例 184 模拟扫雷游戏.....	353
实例 180 打字训练游戏.....	344	实例 185 推箱子游戏.....	357
实例 181 双人竞走游戏.....	346	实例 186 五子棋游戏.....	359

第八部分 综合实例篇

实例 187 综合 CAD 系统.....	362	实例 189 图书管理系统.....	381
实例 188 功能强大的文本编辑器.....	368	实例 190 进销存管理系统.....	385



第一部分 基础篇

精彩导读

- 第一个 C 程序
- 数列求和
- 模拟 ATM 界面
- 编制万年历
- 任意进制数的转换
- 文件的字数统计

实例 1 第一个 C 程序

实例说明

本实例将建立第一个 C 程序，通过使用 Borland 公司的 Turbo C 2.0 软件新建一个 C 程序，在该程序中添加代码，使其能够输出“Hello World!”的问候语。本实例重点介绍如何创建、编译、调试和运行 C 程序。

程序运行效果如图 1-1 所示。

C 语言程序的编译器有很多，如 Turbo C 2.0、Turbo C 3.0、Borland C++ 3.1、Visual C++ 6.0 和 Visual C++.NET 等开发环境都可以编译运行 C 程序。其中最常用的是 Turbo C 2.0（简称 TC）。本书配套光盘上的【Turbo C 2.0】目录下有该编译器的自解压文件 tc.exe。要在硬盘上安装 TC 编译器，可按如下步骤进行：

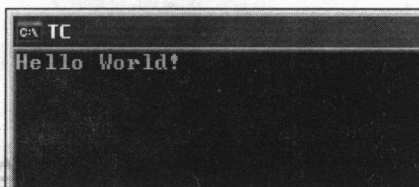


图 1-1 第一个 C 程序运行效果

【1】将配套光盘【Turbo C 2.0】目录下的 tc.exe 文件拷贝到硬盘上，比如 C 盘根目录下。

【2】双击硬盘上的 tc.exe 文件，出现如图 1-2 所示的自解压文件向导。

【3】采用默认的安装目录【C:】，单击【安装】按钮，完成安装。

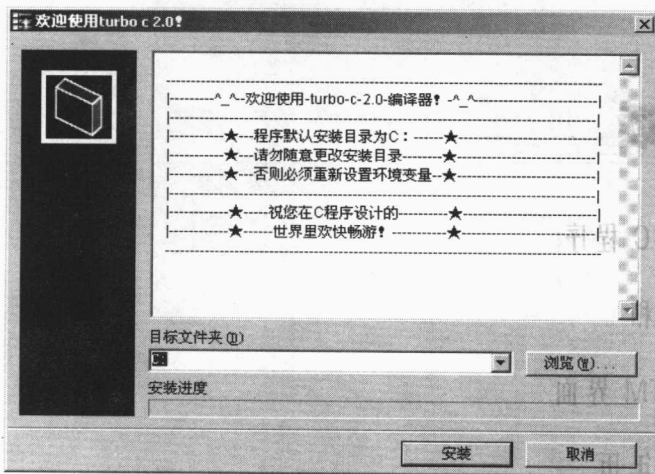


图 1-2 Turbo C 2.0 的安装向导

实例解析

安装好 TC 编译器之后，就可以按照如下步骤创建第一个 C 程序。

【1】双击 TC 安装目录（比如【C:\TC】目录）下的 tc.exe 程序，打开 TC 开发环境。

【2】此时 TC 编译器会自动创建一个 NONAME.C 的 C 程序，如图 1-3 所示。

【3】在该文件中添加【程序代码】中所示的内容。

【4】使用【Alt+F】组合键激活 File 菜单，通过上下光标键选择“Save”命令，或用快捷键“F2”打开如图 1-4 所示的保存文件对话框，输入保存路径和文件名，按回车键，完成保存。

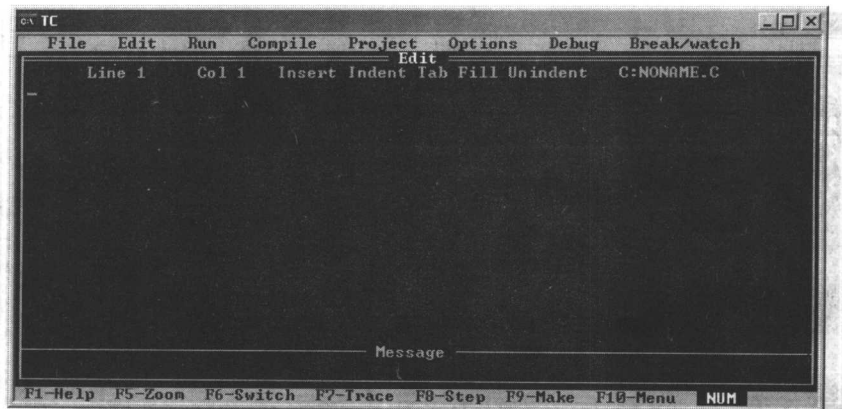


图 1-3 TC 开发环境

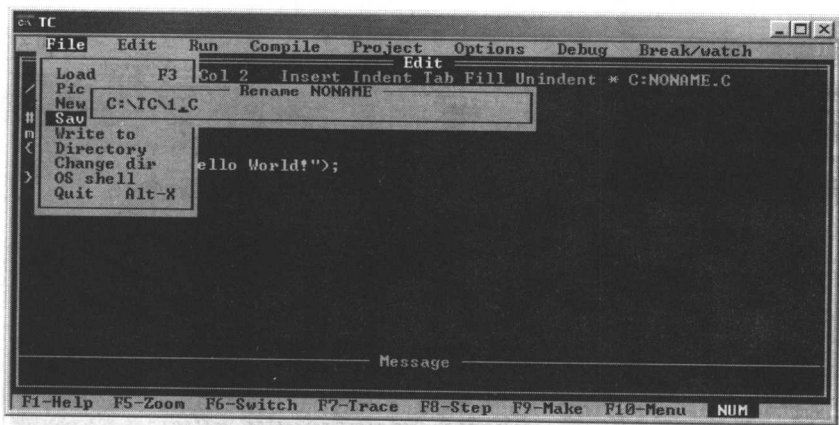


图 1-4 保存程序文件

【5】执行菜单命令“Compile|Compile to OBJ”（见图 1-5），可将程序编译为 OBJ 对象文件，此时编译器将对程序的语法规义进行检查，若有错误或警告，将给出提示。如图 1-6 所示是编译成功的例子，图 1-7 所示是将 printf 语句后面的用于表示语句结束的分号“;”去掉后出现的错误信息。

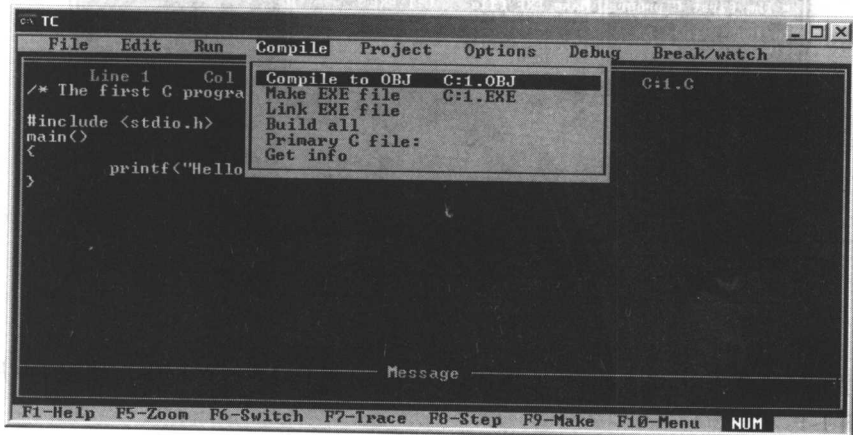


图 1-5 编译程序

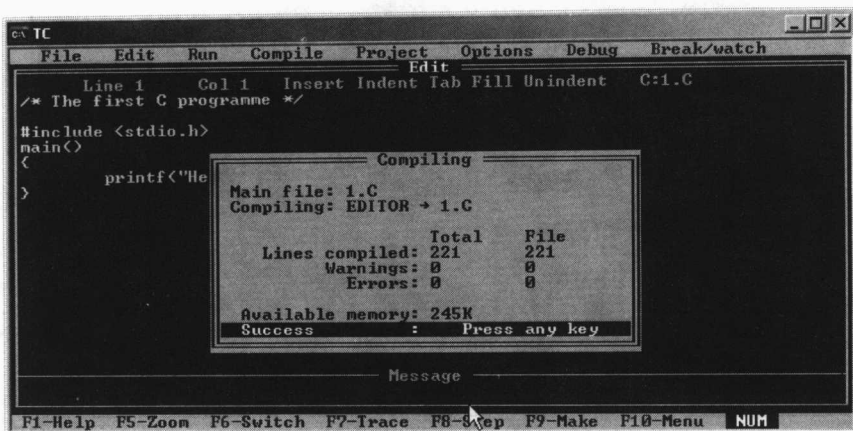


图 1-6 编译成功的信息

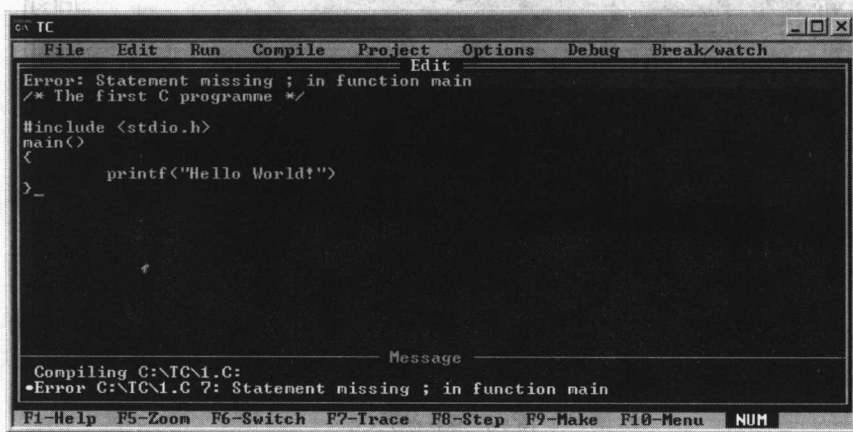


图 1-7 编译时报错的信息

【6】执行菜单命令“Compile | Make EXE File”，生成可执行文件，如图 1-8 所示。

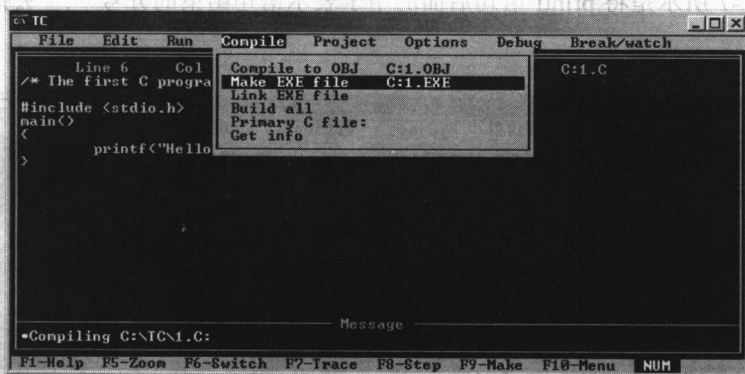


图 1-8 生成可执行文件

【7】执行菜单命令“Run | Run”，或按快捷键【Ctrl + F9】，运行程序，如图 1-9 所示。程序运行结果见图 1-1。

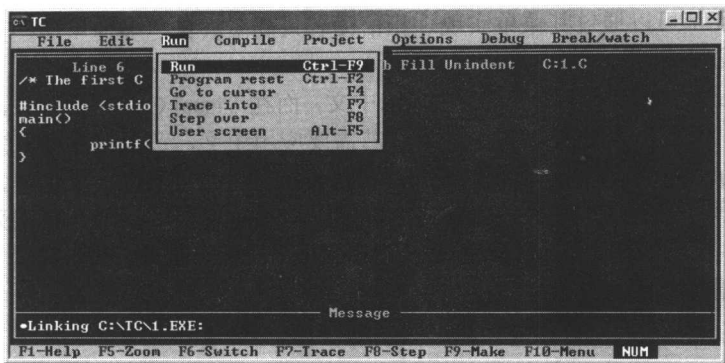


图 1-9 运行程序

【8】要查看程序运行结果，可以执行菜单命令“File | OS Shell”切换到命令行下查看结果，如图 1-10 所示。在命令行下键入“exit”，可以返回 TC 开发环境。

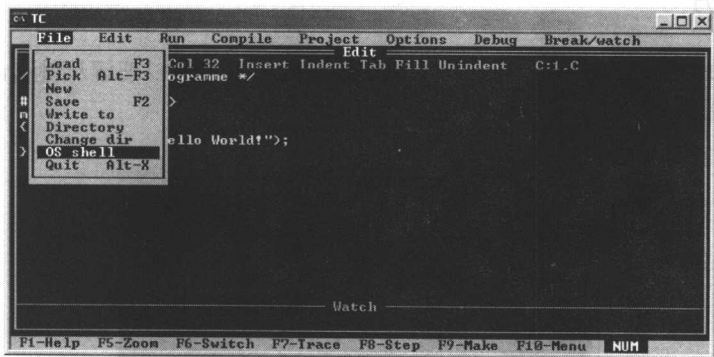


图 1-10 查看运行结果



程序代码

【程序 1】 第一个 C 程序

```
/* The first C programme */
#include <stdio.h>          /* 包含标准输入输出头文件 */
main()                    /* 主函数 */
{
    printf("Hello World!\n"); /* 打印输出信息 */
}
```



归纳注释

#include 语句用于包含头文件 stdio.h，即定义标准输入输出函数的头文件。

每一个 C 程序必须有且只有一个 main() 主函数，这样程序运行时才可以找到入口。

本程序通过 printf 函数向标准的显示终端打印输出字符“Hello World!”。该程序仅打印这些字符，因此光标此时还停在同一行。通常程序打印输出一行后，需要回车换行，这里可以通过加入控制符“\n”来实现，即 printf(“Hello World!\n”);。

TC 编译器使用的其他问题，可以通过运行 TC 安装目录下的 Readme.exe 帮助文件来了解。

基本上所有的功能都可以通过其菜单实现。相信读者只要多尝试，多使用这些菜单命令，很快就可以掌握 TC 开发环境。

值得注意的是，TC 开发环境中，无法显示中文，而在 Visual C++ 6.0 及 Visual C++.NET 等编译器中支持中文，且运行时也能够显示。读者若需要应用中文，可采用这些开发环境。比如上面的 printf 语句改为“printf(“世界，您好!”);”，则显示在命令行下即为“世界，您好!”。本书的实例程序中的代码尽量不使用中文，以便与 TC 编译器兼容，而其后的注释则采用中文，这样虽然在 TC 编译器中无法看到中文（看到的是乱码），但用记事本等其他文本阅读工具打开，则可以看到中文，以便增加程序的可读性。

实例 2 求整数之积

实例说明

从键盘输入两个整数，输出它们的积。通过本实例，读者可以理解从键盘读取输入的数据以及输出整型变量等方法。程序运行结果如图 2-1 所示。

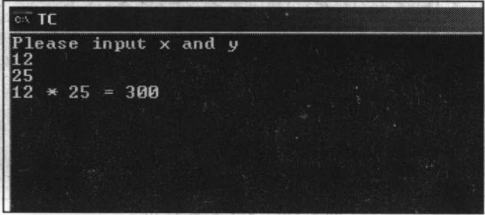


图 2-1 实例 2 程序运行结果

实例解析

C 语言提供的数据结构是以数据类型形式出现的。C 语言的数据类型可分为基本类型、构造类型、指针类型 3 大类。其中基本类型包括整型（变量声明关键字为 int），字符型（变量声明关键字为 char），实型（即浮点型，分为单精度浮点型 float 和双精度浮点型 double float），枚举类型（变量声明关键字为 enum）。构造类型包括数组、结构体和共用体 3 种。

整型、字符型和实型都有常量和变量之分。比如 12、0、-3 为整型常量，4.6、-1.23 为实型常量，'a'、'D' 为字符常量。常量一般从其字面形式即可判别，也可以用一个标识符代表一个常量，比如用预定义#define PRICE 30，这样可提高程序的可读性和可维护性。

值可以改变的量称为变量。变量名只能由字母、数字和下划线组成，且第一个字符必须为字母或下划线。整型变量可分为基本型（用 int 表示）、短整型（用 short int 或 short 表示）、长整型（用 long int 或 long 表示）和无符号型（包括无符号整型、无符号短整型、无符号长整型，分别用 unsigned int、unsigned short 和 unsigned long 表示）。

各种整型变量在不同的计算机机型上存放的内存字节数不同，典型的 IBM PC 中所占的位（bit）和数的范围如表 2-1 所示。整型变量的定义为：

```
int a,b; /* 定义变量 a, b 为整型 */
unsigned short c,d; /* 定义变量 c, d 为无符号短整型 */
```



```
long e, f; /* 定义变量 e, f 为长整型 */
```

表 2-1 整型变量所占位数和数的范围

数据类型	所占位数	数的范围	
int	16	-32768~32768	即 $-2^{15} \sim (2^{15}-1)$
Short [int]	16	-32768~32768	即 $-2^{15} \sim (2^{15}-1)$
Long [int]	32	-2147483648~2147483647	即 $-2^{31} \sim (2^{31}-1)$
unsigned [int]	16	0~65535	即 $0 \sim (2^{16}-1)$
unsigned short	16	0~65535	即 $0 \sim (2^{16}-1)$
unsigned long	32	0~4294967295	即 $0 \sim (2^{32}-1)$

在本例中，设两个整数分别为 x、y，它们的乘积为 m；程序首先调用 printf() 函数，提示用户输入数据，然后调用 scanf() 函数，输入变量 x 和 y 的值，接着求 x 与 y 的积 m，最后输出结果。上述过程用算法描述如下：

【算法】 读入两个整数，输出它们的积

```
{
    提示用户输入数据；
    输入变量 x 和 y 的值；
    计算乘积；
    输出乘积；
}
```



程序代码

【程序 2】 求整数之积

```
/* Input two numbers, output the product */
#include <stdio.h>
main()
{
    int x, y, m; /* 定义整型变量 x, y, m */
    printf("Please input x and y\n"); /* 输出提示信息 */
    scanf("%d%d", &x, &y); /* 读入两个乘数，赋给 x, y 变量 */
    m=x*y; /* 计算两个乘数的积，赋给变量 m */
    printf("%d * %d = %d\n", x, y, m); /* 输出结果 */
}
```



归纳注释

本实例程序实现的是两个整数的简单乘积，同样的，也可以通过修改，实现简单的整数四则运算。整型变量包括整型、短整型、长整型和无符号整型，在 printf() 和 scanf() 函数中的格式说明都可以是 %d。

printf() 输出函数（关键字中的 f 就是表示 format，格式化的意思）用于输出变量的值时，调用格式为“printf(格式控制，输出表列);”，“格式控制”是用双引号括起来的字符串，包括格式字符串和普通字符。格式字符串是以 % 开头的，在 % 后跟各种格式字符，以说明输出数据的类型、形式、长度、小数位数等。如“%d”表示按十进制整型输出，“%ld”表示按十进制长整型输出等。

格式字符串。在 Turbo C 中，格式字符串的一般形式为：[标志][输出最小宽度][精度][长度]类型。其中方括号[]中的项为可选项。各项的意义如下：①类型字符用以表示输出数据的类型，d

——以十进制形式输出带符号整数(正数不输出符号);o——以八进制形式输出无符号整数(不输出前缀 O);x——以十六进制形式输出无符号整数(不输出前缀 OX);u——以十进制形式输出无符号整数;f——以小数形式输出单、双精度实数;e——以指数形式输出单、双精度实数;g——以 %f 中较短的输出宽度输出单、双精度实数;c——输出单个字符;s——输出字符串。②标志字符为-、+、#、空格4种,-——结果左对齐,右边填充空格;+——输出符号(正号或负号),输出值为正时冠以空格,为负时冠以负号;#——对 c、s、d、u 类无影响;对 o 类,在输出时加前缀 0;对 x 类,在输出时加前缀 0x;对 e、g、f 类当结果有小数时才给出小数点。③输出最小宽度:用十进制整数来表示输出的最少位数。若实际位数多于定义的宽度,则按实际位数输出,若实际位数少于定义的宽度则补以空格或 0。④精度格式符以“.”开头,后跟十进制整数。如果输出数字,则表示小数的位数;如果输出的是字符,则表示输出字符的个数;若实际位数大于所定义的精度数,则截去超过的部分。⑤长度格式符为 h、l 两种,h 表示按短整型量输出,l 表示按长整型量输出。

scanf()为输入函数,即按用户指定的格式从键盘上把数据输入到指定的变量之中。scanf 函数的一般形式为“scanf(“格式控制字符串”,地址表列);”其中,格式控制字符串的作用与 printf 函数相同,但不能显示普通字符串,也就是不能显示提示字符串。地址表列中给出各变量的地址。地址是由地址运算符“&”后跟变量名组成的。例如,“&a,&b”分别表示变量 a 和变量 b 的地址。这个地址就是编译器在内存中给 a 和 b 变量分配的地址。

实例 3 比较实数大小

实例说明

从键盘输入两个实数,输出它们中比较大的数。通过本实例,可以学习如何从键盘读取输入的实数,如何实现实数的大小比较,以及实型变量的基本概念等。程序运行结果如图 3-1 所示。

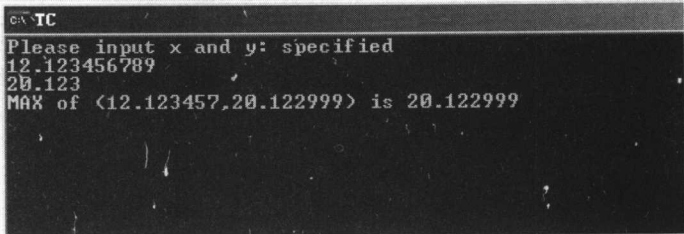


图 3-1 实例 3 程序运行结果

实例解析

实型变量分为单精度(float 型)和双精度(double 型)两类。对于每一个实型变量在使用前都应定义。例如:

```
float x,y ;          /* 指定 x、y 为单精度实数 */
double z ;           /* 指定 z 为双精度实数 */
```

在一般系统中,一个 float 型数据在内存中占用 4 个字节(32 位),一个 double 型数据占 8 个字节。单精度实数提供 7 位有效数字,双精度实数提供 15~16 位有效数字,数值的范围随机器系