

● 高等学校教材

C语言程序设计 实验及习题解答

刘达明 王宁 陈昌志 聂永萍 代永亮 编著



NLIC2970903709



高等教育出版社
HIGHER EDUCATION PRESS

高等学校教材

C 语言程序设计实验及 习题解答

C Yuyan Chengxu Sheji Shiyān ji Xiti Jieda

刘达明 王宁 陈昌志 聂永萍 代永亮 编著



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书是与《C 语言程序设计》(刘达明等编著)配合使用的实验与习题解答,依据教学要求给出了主教材中相应知识点的实验内容,以及各章节的习题参考答案。

全书分 4 部分,第一部分“C 语言程序上机指南”包括 3 章,主要介绍 C 语言程序上机步骤、使用 Turbo C 运行程序和使用 Visual C++运行程序;第二部分“上机实践内容”包括 3 章,主要介绍实验目的和要求、课内上机实验内容、课程设计实验内容;第三部分给出主教材《C 语言程序设计》全部习题的参考答案,以及计算机等级考试模拟练习题参考答案;第四部分为附录,包括程序设计风格、程序错误类型和调试、ASCII 码表、C 运算符。

本书体系新颖、层次清晰、逻辑性强、内容丰富,特别注重可读性和可用性,既介绍了 C 语言程序集成开发环境,又按照知识点覆盖原则设计了基础实验、综合实验、扩展实验 3 个层次的实验内容,还设计了 4 个综合性的课程设计内容,使实验内容环环相扣,层层推进,语言通俗易懂,由浅入深。本书不仅可以作为主教材的参考书,也可以作为其他 C 语言教材的配套实验教程,既适合高等学校师生使用,又适合自学参考。

图书在版编目(CIP)数据

C 语言程序设计实验及习题解答 / 刘达明等编著. — 北京:高等教育出版社, 2013.2

ISBN 978-7-04-036798-0

I. ①C… II. ①刘… III. ①C 语言-程序设计—高等学校-教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 003109 号

策划编辑 刘茜
责任校对 刘春萍

责任编辑 刘茜
责任印制 田甜

封面设计 王洋

版式设计 马敬茹

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100120
印 刷 北京宏伟双华印刷有限公司
开 本 787mm×1092mm 1/16
印 张 12.75
字 数 300 千字
购书热线 010-58581118

咨询电话 400-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>
版 次 2013 年 2 月第 1 版
印 次 2013 年 2 月第 1 次印刷
定 价 20.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换
版权所有 侵权必究
物 料 号 36798-00

前 言

C 语言是国内外使用非常广泛的一种计算机程序设计语言，C 语言以其功能丰富、表达能力强、描述算法方便、系统底层容易控制、语法结构简洁、使用灵活方便、应用面广、目标程序效率高、可移植性好而深得广大程序开发人员的青睐，特别是在硬件产品和底层程序开发中应用更加广泛。目前社会上的很多单位在招聘软件开发人员时也将 C 语言作为最基本的知识点进行考核。

“C 语言程序设计”是一门实践性很强的课程，既要掌握 ANSI C 语言的基本知识，还必须通过大量的编程训练，全面掌握 C 语言的精髓。通过学习，学生不仅要掌握 C 语言程序设计基本知识，培养进行程序设计的基本能力，更重要的是还要在实践中逐步掌握程序设计的思想和方法。

全书共分 4 部分。第一部分“C 语言程序上机指南”包括：第 1 章 C 语言程序上机步骤，主要介绍 C 语言程序的编辑、编译、链接、运行等上机过程；第 2 章使用 Turbo C 运行程序，主要介绍在 Turbo C 集成开发环境下编辑、编译、链接、运行 C 语言程序等的上机方法；第 3 章使用 Visual C++ 运行程序，主要介绍在 Visual C++ 集成开发环境下编辑、编译、链接、运行 C 语言程序等的上机方法。第二部分“上机实践内容”包括：第 4 章实验目的和要求，总体介绍实验目的、实验要求、实验内容的安排原则；第 5 章课内上机实验内容，按照《C 语言程序设计》主教材知识点设计了 12 个实验，每个实验包括实验目的、相关知识点、实验内容，而实验内容又分为 3 个层次：基础实验、综合实验、扩展实验；第 6 章课程设计实验内容，设计了 4 个实验，主要以大型综合实验（课程设计）形式给出，每个实验包括实验目的、相关知识点、实验内容，而实验内容又分为两个层次：大型实验、设计实验。第三部分“《C 语言程序设计》习题参考答案”包括主教材 13 章全部习题、计算机等级考试模拟练习题的参考答案。第四部分“附录”包括 4 个附录：程序设计风格、程序错误类型和调试、ASCII 码表、C 运算符。本书给出的实验程序、习题参考程序等均在 Visual C++ 6.0 下调试通过。

本书由刘达明、王宁、陈昌志、聂永萍、代永亮编著。刘达明编写第 1、2 章，附录 D，第三部分习题 6~9、12 参考答案；王宁编写第 3 章、第 5 章和第 6 章，附录 A 和 B，第三部分习题 11 参考答案；陈昌志编写第 4 章，附录 C，第三部分习题 1、13 参考答案；聂永萍编写第三部分习题 2~5、10、计算机等级考试模拟练习题参考答案；代永亮编写第 5 章实验一部分内容，并调试了本书给出的实验和习题程序。全书由刘达明统稿和校稿，王宁和代永亮承担了部分校稿工作。在编写过程中，瞿中、甘玲、郑先锋、蒋贵全、冯潇、张莉萍、刘亮、唐晓军、

II 前言

薛峙、霍敏霞、向碧群等给出了宝贵意见和建议，在此表示感谢。感谢本书所列参考文献的作者。对使用本书以及给本书提出宝贵意见和建议的全体老师，在此也一并表示感谢。

由于作者水平有限，加之时间仓促，不足之处在所难免，恳请读者批评指正。如需要联系作者，或想索取本书实验和习题源代码，可发信息至 liudaming@cqupt.edu.cn 或 liudaming88@163.com。

作者于 2012 年夏

目 录

第一部分 C 语言程序上机指南

第 1 章 C 语言程序上机步骤	1
第 2 章 使用 Turbo C 运行程序	3
2.1 Turbo C 的安装和启动	3
2.2 运行程序	5
2.3 Turbo C 菜单	8
2.4 窗口操作	9
2.5 程序调试	9
第 3 章 使用 Visual C++运行程序	11
3.1 安装和启动 VC++	12
3.2 输入和编辑源程序	12
3.3 编译链接执行程序	14
3.4 关闭程序工作区	15
3.5 建立和运行包含多个文件程序的方法	16
3.6 程序调试	20

第二部分 上机实践内容

第 4 章 实验目的和要求	23
4.1 上机实验目的	23
4.2 实验要求	24
4.3 实验内容的安排原则	25
第 5 章 课内上机实验内容	26
实验一 C 程序运行环境	26
实验二 数据类型、运算符和表达式	29
实验三 顺序结构程序设计	33
实验四 选择结构程序设计	37
实验五 循环结构程序设计	41
实验六 数组	44
实验七 函数	51
实验八 模块化程序设计	54
实验九 指针 (1)	59

II 目录

实验十 指针 (2)	62
实验十一 结构体、共用体和位运算	65
实验十二 文件	71
第 6 章 课程设计实验内容	75
实验十三 综合实验 (1)	75
实验十四 综合实验 (2)	78
实验十五 综合实验 (3)	89
实验十六 综合实验 (4)	97

第三部分 《C 语言程序设计》 习题参考答案

习题 1 C 语言概述	104
习题 2 简单 C 程序设计	105
习题 3 选择结构程序设计	106
习题 4 循环结构程序设计	109
习题 5 模块化程序设计	113
习题 6 初识指针	117
习题 7 一维数组及其指针	121
习题 8 二维数组及其指针	126
习题 9 字符数组及其指针	130
习题 10 构造数据类型	134
习题 11 文件处理程序设计	138
习题 12 指针的高级应用	143
习题 13 位运算	151
计算机等级考试模拟练习题	152

第四部分 附 录

附录 A 程序设计风格	163
附录 B 程序错误类型和调试	173
附录 C ASCII 码表	192
附录 D C 运算符	193
参考文献	194

第一部分

C 语言程序上机指南

C 编译系统不属于 C 语言本身，它是由 C 系统开发商提供给用户使用的。不同的软件开发商提供了不同的 C 编译系统，但其功能大同小异，都可以用来对用户的源程序进行编译、链接，最终生成可执行的运行文件。C 编译系统的功能大都是以集成开发环境（IDE）形式执行的，它把程序的编辑、编译（含预编译处理）、链接和运行等操作集中到一个界面上，功能丰富，使用方便，直观易用。

20 世纪 90 年代，Turbo C 2.0 使用比较普遍。Turbo C 2.0 是一个 C 语言程序集成开发环境，使用菜单或快捷方式进行操作，而不能使用鼠标进行操作。由于在 Visual C++ 或 Turbo C++ 等 Windows 下的 C 语言集成开发环境中可以方便地使用鼠标进行操作，因此 Visual C++ 和 Turbo C++ 逐渐成为主流平台。

在教学中，一般程序的规模不大，功能相对简单，调试过程不会太复杂，对集成环境的功能要求不是很高，主要考虑使用简单方便。因此，本书着重介绍在 Windows 环境下使用的 Turbo C（简称为 TC）和 Visual C++（简称为 VC++）集成开发环境。读者在学习 C 语言程序设计时可以任选一种，也可以尝试选用其他的 C 编译系统。

第 1 章 C 语言程序上机步骤

按照 C 语言语法规则而编写的 C 程序称为源程序。源程序在计算机内是用 ASCII 码方式存储的，其扩展名一般为“.c”。C 源程序可以通过 C 语言集成开发环境进行编辑，也可以使用诸如记事本等其他软件来编辑。由于 C 源程序不是计算机能够识别的二进制机器指令，因此 ASCII 码的源程序需要先被翻译成机器指令，然后才能被计算机正常运行。

C 源程序翻译过程包括编译与链接两个步骤。首先对源程序进行编译处理（含预处理），把每一条 C 语句翻译成机器指令，生成由机器指令组成的目标程序。然后再把不是自己编写的“库函数”或其他公共函数模块等链接到目标程序中，生成计算机可以执行的文件。

C 程序的编辑、编译、链接、运行等步骤如图 1-1 所示。虚线表示当某一步骤出现错误时的修改路线。运行时，无论是出现编译错误、链接错误，还是运行结果不正确，都需要修改源程序，并将其重新编译、链接和运行，直至将程序调试正确为止。

在上机过程中，根据出错现象找出错误并改正称为程序调试。要在学习程序设计的过程中

逐步培养调试程序的能力，要在上机过程中不断摸索总结，这种能力是经验的不断积累和升华。

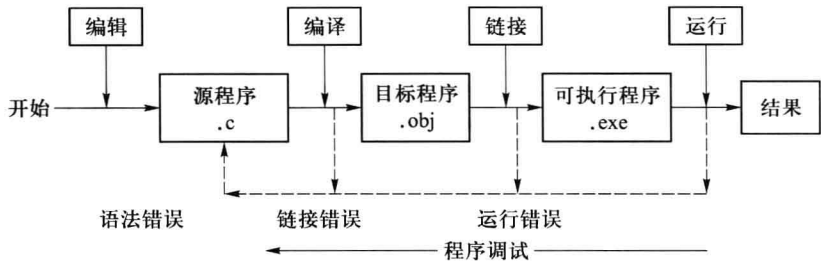


图 1-1 C 语言程序的调试、运行步骤

程序中的错误大致可分为以下 3 类。

- (1) 编译程序时检查出来的语法错误。
- (2) 链接时出现的错误。
- (3) 程序在执行过程中产生的错误。

通常在编程者违反了 C 语言的语法规则，如保留字（关键字）输入错误、大括号不匹配、语句少分号等情况下会发生编译错误。一般在未定义或未指明要链接的函数，或者函数调用不匹配等情况下会引起链接错误，对系统库函数的调用必须通过 `include` 来说明。

对于编译链接错误，C 编译系统会提供出错信息，包括出错位置（行号）、出错提示信息。有时系统会提示很多错误信息，但并不表示真的有这么多个错误，往往是由于前面的一两个错误导致的，所以在纠正了几个错误后不妨再编译链接一次，然后根据最新的出错信息继续纠正。

有些程序通过了编译链接，并能够在计算机上运行，但得到的结果不正确，这类错误最难查找。一部分可能是程序书写错误带来的错误，例如，应该使用变量 `x` 的地方写成了变量 `y`，虽然没有语法错误，但意思完全错了；一部分可能是程序的算法不正确，解题思路不对；还有一些程序有时计算结果正确，有时不正确，这往往是由于编程时在算法中对各种情况考虑不周所致。解决运行错误的首要步骤就是定位错误，即找到出错的位置，才能予以纠正。通常先设法确定错误的大致位置，然后通过 C 语言提供的调试工具找出真正的错误。

为了确定错误的大致位置，可以先把程序分成几大块，并在每一块的结束位置，手工计算一个或几个阶段性结果，然后用调试的方式运行程序，到每一块结束时，检查程序运行的实际结果与手工计算是否一致，通过这些阶段性结果来确定各块是否正确。对于出错的程序块，可逐条仔细检查各语句，找出错误所在。如果出错块程序较长，难以一下子找出错误，可以进一步把该块细分成更小的块，按照上述步骤作进一步检查。在确定了大致出错位置后，如果无法直接看出错误，可以单步运行相关位置的几条语句，并逐条检查，一定能找出错误的语句。

当运行结果有时正确有时不正确时，最好多选几组典型的输入数据进行测试，除了普通的数据外，还应包含一些边界数据和不正确的数据。例如，确定正常的输入数据范围后，分别以最小值、最大值、比最小值小的值和比最大值大的值，多方面运行检查自己的程序。

本实验教程分别以 TC 和 VC++ 为上机平台，对 C 语言程序的编译、链接和调试、运行进

行简单介绍。建议一开始学习 C 语言程序上机操作时把注意力放在程序的编译、链接和运行上，待熟练掌握后，再重点学习程序的调试方法。

第 2 章 使用 Turbo C 运行程序

2.1 Turbo C 的安装和启动

Turbo C 是一个常用的、最基本的 C 语言工具，一般简称为 TC。它为 C 语言开发提供了操作便利的集成环境。源程序的输入、修改、调试及运行都可以在 TC 集成环境下完成，非常方便有效。TC 3.0 系统非常小巧，但功能齐全。它除了具有 TC 2.0 支持 DOS 环境的特点外，还能够在操作中使用鼠标，可以通过键盘操纵菜单或快捷键，也可以使用鼠标确定具体位置，这也成为使用 TC 3.0 强于 TC 2.0 及其之前版本的一大原因。

由于 TC 支持 DOS 环境，TC 系统的安装十分方便。如果有安装盘，可以按照提示一步步完成安装；如果没有安装盘，可以从其他计算机中直接复制已安装好的系统。

如果要在 TC 环境中编辑源程序，首先要找到 TC 系统的安装（复制）目录，找到 TC.exe 应用程序并双击它来启动 TC 系统；也可以用鼠标双击快捷的 TC 图标（如图 2-1（a）所示），本书以 Turbo C++ 3.0 为例进行演示，其界面如图 2-1（b）所示。

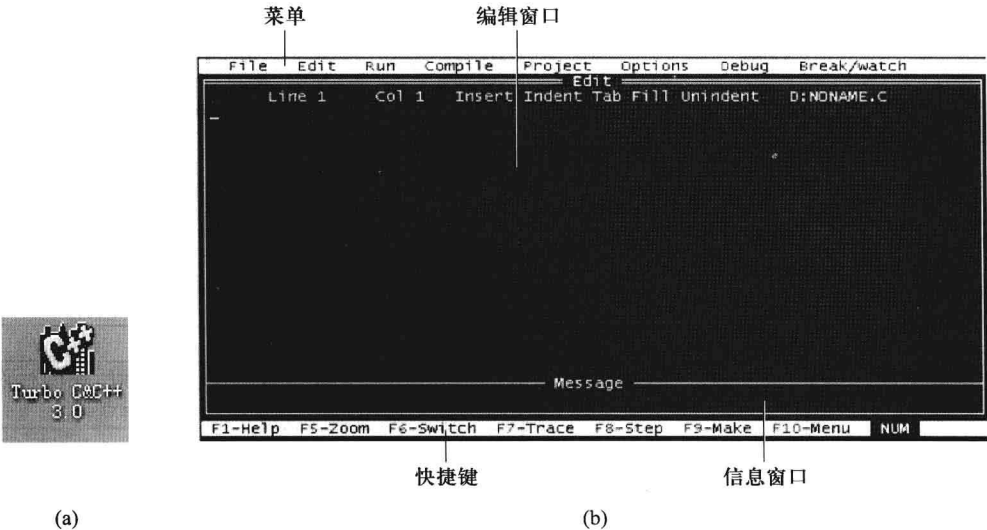


图 2-1 TC 快捷图标和编程界面

在图 2-1 中，菜单中包含了进行所有操作的命令；编辑窗口是用于输入、修改程序的区域；信息窗口用于显示程序编译、链接和运行过程中的错误信息或有关提示信息；快捷键用于给出常用操作的快捷键提示信息。

表 2-1 中给出了菜单栏各个下拉菜单的具体含义，括号中是代表操作的快捷键。

表 2-1 TC 中菜单栏各下拉菜单功能中英文对照表

File（文件）		Window（窗口）	
New	新建	Message	信息窗口
Open（F3）	打开	Output	输出窗口
Save（F2）	保存	Watch	监视窗口
Save as	另存为	User screen	用户屏幕
Save all	保存全部	Register	寄存器
Change dir	改变目录	Project	项目
Print	打印	Project notes	项目记录
Dos shell	返回 DOS	List all	罗列所有窗口
Quit（Alt+X）	退出		
Search（查找）		Edit（编辑）	
Find	查找	Undo	撤销
Replace	替换	Redo	重做
Search again	继续查找	Cut	剪切
Go to line number	转到某行	Copy	复制
Previous error	查看前一个错误	Paste	粘贴
Next error	查看下一个错误	Clear	清除
Locate function	查找功能	Copy example	复制副本
		Show clipboard	查看剪贴板
Compile（编译）		Run（运行）	
Compile	编译	Run（Ctrl+F9）	运行
Make	生成目标文件	Program reset	程序重启
Link	链接	Go to cursor	运行到光标处
Build all	建立所有文件	Trace into	跟踪进入
Information	程序信息	Step over	单步执行
Remove messages	删除信息	Arguments	命令行参数
Project（项目）			
Open project	打开项目	Inspect	检查
Close project	关闭项目	Evaluate/modify	评定/修改
Add item	添加项目	Call stack	检查堆栈
Delete options	删除选项	Watch	监视
Local options	本地选项	Toggle breakpoint	对光标所在的行设置或清除断点
Include files	包含文件	Breakpoints	断点
Window（窗口）		Help（帮助）	
Size/move	大小/移动	Contents	帮助文档
Zoom	分区控制	Index	索引
Tile	窗口平铺	Topic search	主题搜索
Cascade	级联窗口	Previous topic	上一个主题
Next	下一个窗口	Help on help	在线帮助
Close	关闭	About	关于 Turbo C 3.0
Close all	关闭全部		

续表

Options (选项)		Options (选项)	
Application	应用程序	Directories	路径
Compiler	编译器	Environment	环境
Transfer	转换	Save	存储配置
Make	生成		
Linker	链接器		
Librarian	管理		
Debugger	调试器		

注：在 Run 菜单中：

- (1) “运行到光标处”选项在调试程序时使用，选择该项可使程序运行到光标所在行，光标所在行必须为一条可执行语句，否则会提示错误。
- (2) “跟踪进入”选项表示在执行一条调用其他用户定义的子函数时，若用 Trace into 选项，则执行长条将跟踪到该子函数内部去执行。
- (3) “单步执行”选项表示执行当前函数的下一条语句，即使用户函数调用，执行长条也不会跟踪进函数内部。

2.2 运行程序

要运行一个 C 程序，必须经过输入源程序、修改错误、编译、链接和运行等几个步骤。

1. 编辑

启动 TC 后直接打开编辑窗口，或者按 Alt+F 快捷键操作 File 菜单，如图 2-1 所示。可以使用 ↑、↓、←、→4 个方向键和回车键（在 Turbo C 3.0 中也可以直接使用鼠标）打开 File 菜单中的 New 命令新建一个文件，如图 2-2 所示。然后在此编辑窗口中编辑源代码，如图 2-3 所示的一个是用 C 语言编写的输出 “hello world !” 字符序列的简单程序。

如果要对程序内容进行修改，同样可以使用方向键（在 Turbo C 3.0 中直接使用鼠标）移动光标到所需位置，然后删除错误，输入正确的内容。编辑完成之后使用 File 下拉菜单中的 Save 命令保存编辑的源程序，默认的保存路径为安装 TC 时设定的文件路径，若要保存在另外的文件夹里，则单击 File 下拉菜单中的 Save as 命令，然后在弹出的对话框中输入要保存源程序的路径和文件名（C 源文件扩展名为 “.c”），输入完后单击 OK 按钮即可保存，如图 2-4 所示。

2. 编译、链接和运行

编译、链接和运行操作可以分步进行（先按 Alt+F9 键检查有无编译错误并生成.obj 文件，再执行 Link 菜单检查有无链接错误并生成可执行文件.exe，最后按 Ctrl+F9 键运行并查看运行结果）。

也可直接按 Ctrl+F9 键，同时完成对编辑窗口中的程序的编译、链接和运行 3 个步骤。此时有可能发生以下几种情况。

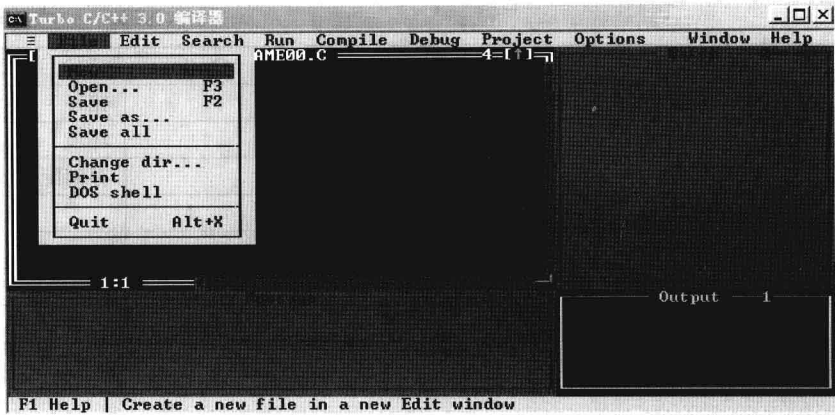


图 2-2 在 TC 3.0 中新建文件

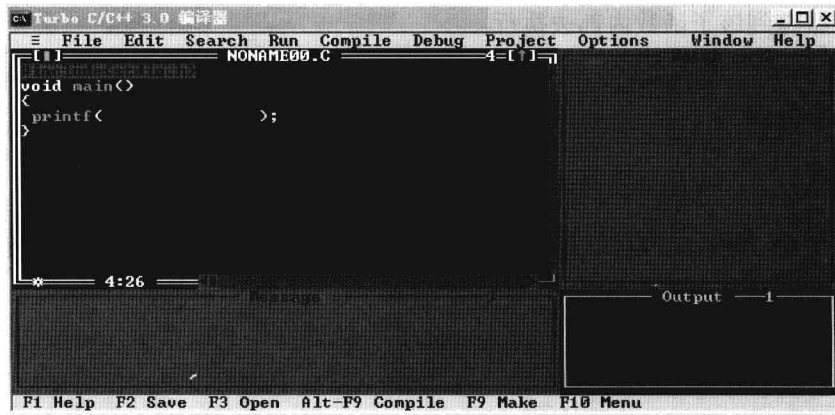


图 2-3 在编辑窗口中编辑源程序

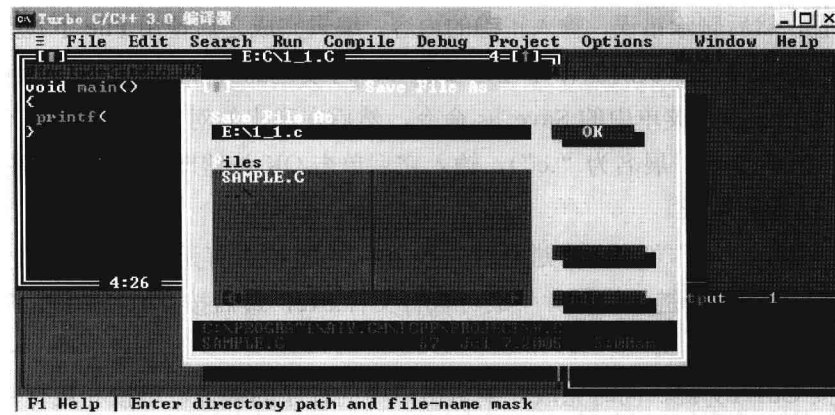


图 2-4 源程序的“另存为”对话框

- (1) 如果程序没有错误，将直接运行程序。
- (2) 如果存在编译错误，信息窗口中将显示错误信息，并终止链接与执行步骤。
- (3) 如果编译通过，但存在链接错误，信息窗口中将显示错误信息，并终止执行步骤。

任何错误都必须纠正后，重新按 **Ctrl+F9** 键运行。如果还有错，则继续修改，直到能正确执行为止。

即使通过了编译和链接，也并不说明程序就没有错误了，解题思路或非格式的语句错误，都会导致无法得到正确的结果，甚至程序无法正确执行。如果程序在执行过程中无法结束（死机），可以按 **Ctrl+Break** 键强制结束。为了避免这种情况发生，建议初学者采用分步操作的方式进行编译、链接、运行。

如果程序没有编译、链接错误，就将出现如图 2-5 所示的界面，并在 **Output** 窗口中显示出程序的运行结果。

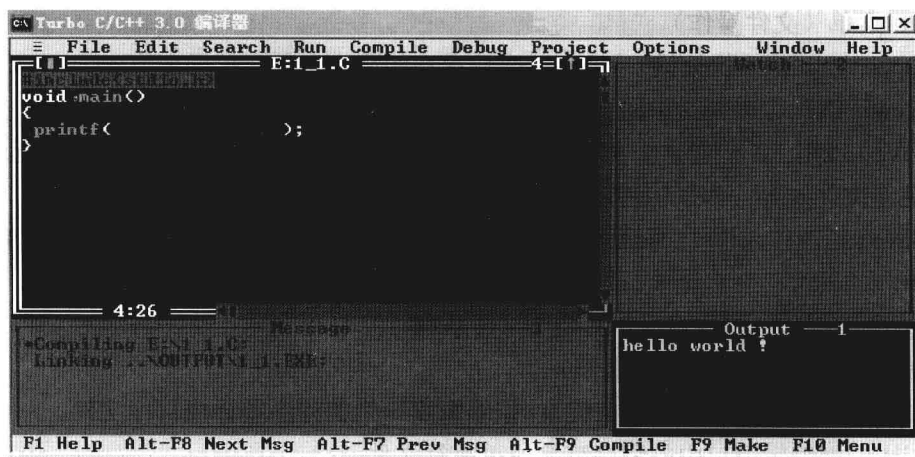


图 2-5 最终运行结果

3. 输入数据

如果程序中有 **scanf()** 语句，则屏幕上将出现一个黑底的输入窗口，等待输入数据。当输入一个数据后，按空格键或回车键，再输入下一个数据，直到输入完所有数据为止。如果 **scanf()** 语句格式规定使用逗号或非%开始的格式字符分隔，则在各数据之间还要按照原样输入逗号等字符。建议初学者在 **scanf()** 语句中尽量不要使用除了格式输入符之外的其他普通字符，以免出错。

4. 查看结果

如果在编辑界面上没有设置显示输出窗口，在输入完数据后，系统自动关闭输入窗口，运行程序后，回到编辑窗口。如果此时运行结果未显示在屏幕上，可以按 **Alt+F5** 键查看运行结果，再按任意键关闭输出窗口。

对于没有输入要求的程序，按 **Ctrl+F9** 键运行，屏幕上可能没有什么变化，但这并不表示程序未被运行，这时按 **Alt+F5** 键就能看到结果。

通过上述步骤，读者可以快速掌握 C 语言的上机过程，运行自己的程序。

5. 运行环境设置

第一次上机时, 如果一个简单正确的程序无法编译、链接、运行, 需要先检查一下运行环境是否设置正确。具体参看 Turbo C 菜单中的 Options 菜单介绍。

2.3 Turbo C 菜单

前面介绍的是上机过程中最基本的一些操作。为了对 TC 的功能有一个全面的了解, 这里逐一介绍 TC 的各主要菜单。

TC 中提供了完善的帮助信息, 按 F1 键屏幕上将弹出帮助窗口, 但其中的信息都是英文的。

TC 的快捷键通常是组合键, 如 Ctrl+F9 表示要同时按下两个键。对于三键的组合键, 其前两个键要同时按下, 然后再快速按下第三个键, 如 Ctrl+K+B 键表示先同时按下 Ctrl 键和 K 键, 然后再按 B 键, 按完三键同时快速松手。

1. File 菜单 (文件操作)

(1) Open (或用 F3 快捷键): 调入一个已存在的程序文件 (扩展名为.c)。

(2) New: 清除编辑窗口中的程序, 以输入一个新程序。

(3) Save (或用 F2 快捷键): 把编辑窗口中的程序保存到文件中。如果该程序已经保存过, 该操作将更新文件内容; 如果该程序是新输入的, 需要进一步输入路径和文件名。

(4) Save as: 把当前程序写到另外命名的文件上, 相当于 Windows “文件” 菜单中的 “另存为” 命令。

(5) Quit (或用 Alt+X 快捷键): 退出 TC 系统。

2. Compile 菜单 (编译链接操作)

(1) Compile (或用 Alt+F9 快捷键): 把编辑窗口中的程序编译成目标文件。

(2) Link: 把编辑窗口中的程序链接成可执行文件。

(3) Make (或用 F9 快捷键): 将编辑窗口中的程序进行编译、链接, 生成可执行文件。

3. Run 菜单 (运行操作)

Run (或用 Ctrl+F9 快捷键): 执行编辑窗口中的程序。如果该程序最近未被编译链接过, 将先自动编译链接, 然后再执行。

4. Options 菜单 (建立工作环境)

(1) 工作环境目录的设置: 如果使用安装盘安装, 该工作环境目录会自动设置好。如果是通过系统复制的, 则需要对 Options 菜单中的 Directories 菜单项进行设置。假设 TC 所在目录为 “D:\TC”, 在打开的目录窗口中应输入:

Include directories: D:\TC\INCLUDE

Library directories: D:\TC\LIB

确保程序链接时能从这两个位置找到系统包含文件和系统库文件, 如图 2-6 所示。

(2) 命令行参数输入: 在学习了 “指针” 后, 会用到命令行参数。可以执行 Options 菜单中的 Arguments 命令, 输入命令行参数 (不包括可执行文件名, 各参数间用空格分隔), 按回车键结束输入, 按 Esc 键隐去菜单, 然后按 Ctrl+F9 键运行程序, 参数便能被主函数接受。

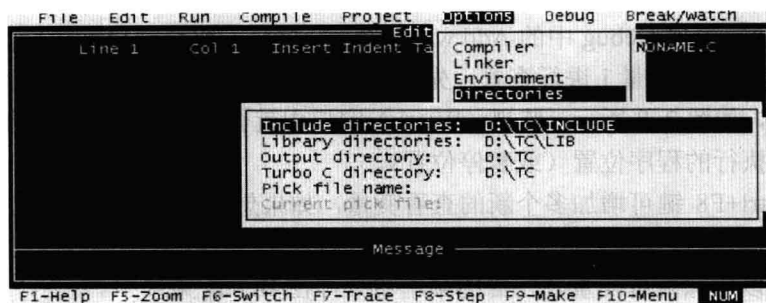


图 2-6 TC 工作目录设置

2.4 窗口操作

在如图 2-1 所示的 TC 界面中，屏幕上半部分是编辑窗口，下半部分是信息窗口。如果想把编辑窗口扩大到整屏，可按 F5 键，这时信息窗口将被遮住，再按 F5 键又可以恢复成上下两个窗口。如果编辑窗口被扩大到整屏，而又想看一下信息窗口，可使用 F6 键进行窗口切换。如果在程序执行时又打开了监视窗口，按 F6 键可以在 3 个窗口间进行切换，切换过程是按一个方向循环。

在 Windows 中运行 TC，所打开的窗口往往较小，边框线也不对。按 Alt+Enter 键可以将窗口最大化，成为仿真 DOS 界面。再按 Alt+Enter 键又会恢复成较小的窗口。

2.5 程序调试

TC 提供了必要的调试手段和工具，下面按照使用过程予以介绍。

1. 让程序执行到中途暂停以便观察阶段性结果

(1) 方法一：使程序执行到光标所在的那一行暂停。

① 把光标移动到需暂停的行上。

② 按 F4 键或执行 Run 菜单中的 Go to cursor 命令。当程序执行到该行将会暂停。如果把光标移动到后面的某个位置，再按 F4 键，程序将从当前的暂停点继续执行到新的光标位置，第二次暂停。

(2) 方法二：把光标所在的那一行设置成断点，然后按 Ctrl+F9 键执行，当程序执行到该行时将会暂停。

① 把光标移动到需暂停的行上。

② 按 Ctrl+F8 键或执行菜单 Debug 中的 Toggle breakpoint 命令将该行设置成断点。

③ 按 Ctrl+F9 键执行。

不管是通过光标位置还是断点设置，其所在的程序行必须是程序执行的必经之路，亦即不应该是分支结构中的语句，因为该语句在程序执行过程中受到判断条件的限制，有可能因条件不满足而不能执行，这时程序将一直执行到结束位置或下一个断点位置。

2. 设置需观察的结果变量

按照上面的操作，可使程序执行到指定位置时暂停，其目的是为了查看有关的中间结果。

按 Ctrl+F7 键或执行菜单 Debug 中的 Add watch 命令，屏幕上将会弹出小窗口供输入查看变量，如图 2-7 所示，如输入变量 i 进行查看。先把光标移动到第 5 行，然后按 F4 键执行，程序运行到第 5 行暂停，如图 2-8 所示，监视 (Watch) 窗口中就会显示查看变量 i 的当前值。绿色光条表示当前将被执行的程序位置 (或暂停位置)。

多次使用 Ctrl+F8 键可增加多个新的查看变量，如果想改变查看变量的名称或删除查看变量，可以按 F6 键，使查看窗口成为操作窗口，然后按回车键 Enter，可以改变查看变量，按删除键 Delete 键可以删除查看变量。相应的菜单命令都在 Debug 菜单中。

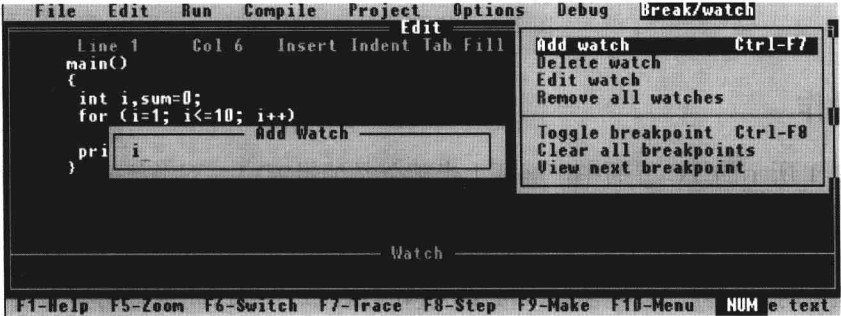


图 2-7 输入查看变量

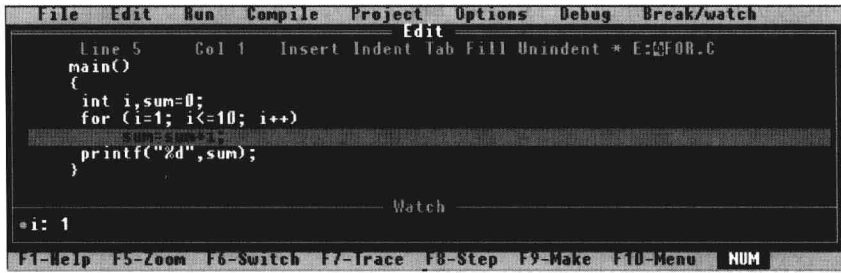


图 2-8 查看中间结果

3. 单步执行

当程序执行到某个位置时发现结果已经不正确了，说明在此之前肯定有错误存在。如果能确定一小段程序可能有错，先按上面步骤暂停在该小段程序的头一行，再输入若干个查看变量，然后单步执行，即一次执行一行语句，逐行检查下来，看看到底是哪一行造成结果出现错误，从而确定错误的语句并予以纠正。

单步执行可按 F8 键或执行菜单 Run 中的 Step over 命令。如果遇到自定义函数调用，想进入函数进行单步执行，可按 F7 键或执行菜单 Run 中的 Trace into 命令。对不是函数调用的语句来说，F7 键与 F8 键的作用相同。

4. 断点的使用

使用断点也可以使程序暂停。但一旦设置了断点，不管是否还需要调试程序，每次执行程序都会在断点上暂停。因此，调试结束后应取消所定义的断点。方法是先把光标定位在断点所在行，再按 Ctrl+F8 键执行菜单 Debug 中的 Toggle breakpoint 命令，该命令是一个开关，第一