



PRACTICE
GUIDANCE
FOR
C PROGRAMMING

PRACTICE
GUIDANCE
FOR
C PROGRAMMING

C 语言程序设计 实训指南

王 勇 李淑珍 余学兵 主编

PRACTICE
GUIDANCE
FOR
C PROGRAMMING



ZHEJIANG UNIVERSITY PRESS

浙江大學出版社

C 语言程序设计实训指南

——全国计算机等级考试(新大纲)

主 编 王 勇 李淑珍 余学兵
主 审 程从从

浙 江 大 学 出 版 社

图书在版编目(CIP)数据

C 语言程序设计实训指南 / 王勇, 李淑珍, 余学兵主编. — 杭州: 浙江大学出版社, 2006. 10
ISBN 7-308-04942-6

I. C... II. ①... ②李... ③余... III. C 语言 - 程序设计 - 高等学校 - 教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 111626 号

C 语言程序设计实训指南

王 勇 李淑珍 余学兵 主编

责任编辑 徐宝澍

出版发行 浙江大学出版社

(杭州天目山路 148 号 邮政编码 310028)

(E-mail: zupress@mail.hz.zj.cn)

(网址: <http://www.zjupress.com>)

排 版 浙江大学出版社电脑排版中心

印 刷 浙江省煤田地制局制图印刷厂

开 本 787mm × 1092mm 1/16

印 张 20.75

字 数 544 千字

版 印 次 2006 年 10 月第 1 版 2006 年 10 月第 1 次印刷

书 号 ISBN 7-308-04942-6/TP · 313

定 价 38.00 元

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社新世纪事业部电话 (0571) 88273985

C 语言程序设计实训指南编委会

主 编	王 勇	李淑珍	余学兵
副主编	黄雷鸣	肖守柏	徐军莉
编 委	王庆延	姚 华	邓 斌
	吴迎军	熊小梅	王 勇
	李淑珍	肖守柏	余学兵
	徐军莉	黄雷鸣	罗 琳
主 审	程从从		

前 言

C 语言程序设计实训指南——全国计算机等级考试(新大纲)是我们编写的《C 语言程序设计教程》配套教材。我们认真学习了新大纲,收集和分析历年考试试题,结合 C 语言基本知识,将三者融合在一起,以 C 语言程序设计知识为基础,紧扣考试大纲要求,精选试题,加强上机实验操作和解题技巧等方面训练。编写本实训教材,其目的就是帮助考生顺利参加考试,取得较好成绩。

本书内容共有 11 章,第 1 章 C 语言程序上机调试过程,第 2 章 C 语言基础,第 3 章数据类型及运算,第 4 章基础结构及其实现,第 5 章数组,第 6 章函数,第 7 章指针,第 8 章结构体与共同体,第 9 章文件,第 10 章二级 C 语言上机试题精选与分析,第 11 章公共基础知识,模拟试题及附录。

具体编写中,从第 2 章到第 9 章,编写的格式为:本章知识重点,考试要点,本章教材习题参考答案,本章典型考题与解析,本章指导实训等五部分内容。为了帮助考生参加上机应试考试,特编写了二级 C 语言上机试题精解与分析,讲授了考题解析、达标训练和应试技巧作为第 10 章的内容,第 11 章是按大纲要求编写的公共基础知识内容。

第 1 章、第 2 章、第 3 章由邓斌、吴迎军编写,第 4 章由李淑珍、罗琳编写,第 5 章由王勇、黄雷鸣编写,第 6 章由余学兵、徐军莉编写,第 7 章由熊小梅编写,第 8 章、第 9 章由王勇、黄雷鸣、肖守柏编写,第 10 章由李淑珍编写,第 11 章由王庆延、王勇、黄雷鸣编写,模拟试题、附录由邓斌、吴迎军编写。全书由王庆延教授统稿,由程从从教授主审。

本书内容丰富,基础知识完整,实用性、操作性很强,指出了学生应试技巧,对学生参加 C 语言考试有极大的帮助。

虽然作者们都是从事 C 语言程序设计教学的老师,具有较丰富的教学经验,但由于时间短、任务重,难免有错误之处,敬请同行和读者批评指正。

编 者

2006 年 7 月

目 录

第1章 C语言程序上机调试过程	(1)
1.1 Turbo C 集成开发环境	(1)
1.1.1 Turbo C 2.0 的安装	(1)
1.1.2 Turbo C 2.0 集成化操作界面	(1)
1.1.3 C语言程序上机调试步骤	(3)
1.1.4 上机操作实例	(5)
1.2 程序动态调试方法	(5)
1.2.1 按步执行方法	(5)
1.2.2 设置断点方法	(7)
第2章 C语言基础	(10)
2.1 考试要点	(10)
2.1.1 保留字、标识符	(10)
2.1.2 C语言程序的结构	(11)
2.2 教材习题参考答案	(11)
2.3 上机指导	(13)
第3章 数据类型及运算	(16)
3.1 考试要点	(16)
3.1.1 C语言的数据类型	(16)
3.1.2 C语言的基本数据类型	(17)
3.1.3 常量	(17)
3.1.4 变量	(17)
3.1.5 数据类型间的转换	(18)
3.1.6 运算符和表达式	(18)
3.1.7 位运算	(20)
3.2 教材习题参考答案	(21)
3.3 典型考题与解析	(24)

3.3.1 选择题	(24)
3.3.2 填空题	(26)
3.4 上机指导	(27)
第4章 基本结构及其实现	(30)
4.1 考试要点	(30)
4.1.1 输入输出库函数	(30)
4.1.2 选择分支结构	(31)
4.1.3 多分支结构	(34)
4.1.4 while 循环结构	(34)
4.1.5 do-while 循环结构	(35)
4.1.6 for 循环结构	(35)
4.1.7 辅助语句 break 与 continue	(36)
4.2 教材习题参考答案	(36)
4.3 典型考题与解析	(52)
4.3.1 选择题	(52)
4.3.2 填空题	(56)
4.4 上机指导	(58)
第5章 数组	(60)
5.1 考试要点	(60)
5.1.1 一维数组	(60)
5.1.2 二维数组	(61)
5.1.3 字符数组	(62)
5.2 教材习题参考答案	(64)
5.3 典型考题与解析	(71)
5.3.1 选择题	(71)
5.3.2 填空题	(73)
5.3.3 改错题	(75)
5.3.4 写出程序运行结果	(77)
5.4 上机指导	(78)
第6章 函数	(81)
6.1 考试要点	(81)
6.1.1 函数的定义和调用	(81)
6.1.2 变量的存储属性	(83)
6.1.3 内部函数与外部函数	(84)
6.1.4 编译预处理	(85)
6.2 教材习题参考答案	(86)
6.3 典型考题与解析	(96)
6.3.1 选择题	(96)
6.3.2 填空题	(100)

6.3.3	改错题	(101)
6.3.4	写出程序运行结果	(105)
6.3.5	编程题	(105)
6.4	上机指导	(107)
第7章	指针	(109)
7.1	考试要点	(109)
7.1.1	指针变量与地址的概念	(109)
7.1.2	指针变量的定义与使用	(109)
7.1.3	指针运算	(110)
7.1.4	指针与一维数组	(111)
7.1.5	指针与二维数组	(111)
7.1.6	指针与字符串	(112)
7.1.7	指针数组	(113)
7.1.8	二级指针	(113)
7.1.9	函数指针	(113)
7.1.10	指针函数	(114)
7.2	教材习题参考答案	(114)
7.3	典型考题与解析	(126)
7.3.1	单选题	(126)
7.3.2	填空题	(131)
7.3.3	改错题	(134)
7.3.4	程序分析题	(136)
7.4	上机指导	(137)
第8章	结构体与共用体	(142)
8.1	考试要点	(142)
8.1.1	结构体	(142)
8.1.2	共用体	(144)
8.1.3	枚举型	(145)
8.1.4	用户自定义类型	(145)
8.2	教材习题参考答案	(146)
8.3	典型考题与解析	(151)
8.3.1	选择题	(151)
8.3.2	填空题	(154)
8.3.3	改错题	(156)
8.3.4	写出程序运行结果	(157)
8.4	上机指导	(158)
第9章	文件	(162)
9.1	考试要点	(162)
9.1.1	文件概述	(162)

9.1.2 文件操作	(163)
9.2 教材习题参考答案	(165)
9.3 典型考题与解析	(169)
9.3.1 选择题	(169)
9.3.2 填空题	(170)
9.3.3 改错题	(172)
9.3.4 写出程序运行结果	(173)
9.4 上机指导	(174)
第10章 二级C语言上机试题精解与分析	(176)
10.1 上机试题的应试技巧	(176)
10.2 程序填空题精选与分析	(178)
10.3 程序修改(改错)题精选与分析	(184)
10.4 程序编程题精选与分析	(196)
10.5 上机考试模拟试题及参考答案	(207)
第11章 公共基础知识	(230)
11.1 基本数据结构与算法	(230)
11.1.1 数据表示与数据处理	(230)
11.1.2 几个基本术语	(232)
11.1.3 算法分析	(234)
11.2 线性表	(236)
11.2.1 线性表的定义	(236)
11.2.2 线性表的基本操作	(237)
11.2.3 线性表的顺序存储	(237)
11.2.4 线性表的链式存储结构	(240)
11.3 栈和队列	(244)
11.3.1 栈的逻辑结构及基本运算	(244)
11.3.2 队列的逻辑结构及基本运算	(246)
11.4 树形结构	(247)
11.4.1 树的基本概念	(247)
11.4.2 二叉树	(248)
11.4.3 二叉树的遍历	(250)
11.5 顺序查找与二分法查找算法	(251)
11.5.1 顺序查找	(251)
11.5.2 二分法查找	(252)
11.6 排序	(253)
11.6.1 排序的基本概念	(253)
11.7 软件工程概述	(254)
11.7.1 软件的发展与软件“危机”	(254)
11.7.2 软件生存周期	(255)

11.8	结构化分析方法	(256)
11.8.1	自顶向下逐步分解的分析策略	(257)
11.8.2	数据流图	(257)
11.8.3	数据字典	(258)
11.9	总体设计与详细设计	(258)
11.10	软件测试	(259)
11.10.1	软件测试(程序测试)	(260)
11.10.2	黑盒测试与白盒测试	(260)
11.11	程序设计基础	(263)
11.11.1	结构化程序设计	(263)
11.12	程序设计风格	(264)
11.12.1	源程序文档化	(264)
11.12.2	数据说明	(265)
11.12.3	语句结构	(265)
11.12.4	输入和输出	(265)
11.13	面向对象程序设计概念	(265)
11.13.1	程序设计基本模式	(266)
11.13.2	程序设计模式的转换	(266)
11.13.3	面向对象的概念	(267)
11.13.4	面向对象的特征	(269)
11.13.5	面向对象的要素	(270)
11.14	数据库基础	(271)
11.14.1	数据库基本概念	(271)
11.14.2	数据库系统的模式结构	(272)
11.15	数据库设计基础	(276)
11.15.1	信息世界的基本概念	(276)
11.15.2	实体联系模型	(277)
11.15.3	关系模型与关系数据库	(279)
11.15.4	数据库设计步骤	(284)
11.16	试题解析	(287)
附录 I	全国计算机二级 C 语言笔试模拟试题	(290)
附录 II	C 语言常用语法提要	(312)
附录 III	C 语言出错中英文对照表	(316)

第 1 章

C 语言程序上机调试过程

本章知识重点:

- 了解 Turbo C 2.0 的集成环境
- 掌握在 Turbo C 2.0 环境中编辑、编译、连接和运行简单的 C 语言程序
- 掌握在 Turbo C 2.0 环境中使用按步执行的方法和断点的使用方法

1.1 Turbo C 集成开发环境



第一章 C 语言程序上机调试过程

Turbo C 2.0 可运行于 X86T 系列兼容计算机。Turbo C 占用资源少,几乎在所有的微机上都可以使用 Turbo C。Turbo C 是基于 DOS 平台的 C 编译系统,只要求在计算机安装 DOS2.0 以上的版本。

1.1.1 Turbo C 2.0 的安装

使用 Turbo C 之前,必须先将 Turbo C 系统装入用户的磁盘上。用户可以指定安装目录,系统默认的目录为 C:\tc。Turbo C 2.0 的安装非常简单,只要在命令行方式下键入 Turbo C 目录中的 INSTALL,或者在 Windows 环境下双击 INSTALL.EXE 图标即可。安装完成后,在 TC 目录下主要有以下的文件和子目录:

C:\tc 其中包含 tc.exe、tcc.exe、make.exe 等可执行文件。

C:\tc\include 其中包含 stdio.h、math.h 等各种头文件。

C:\tc\lib 其中包含 maths.lib、math1.lib 等各种库函数文件。

1.1.2 Turbo C 2.0 集成化操作界面

1. 运行 Turbo C 2.0

(1)由 DOS 平台进入 Turbo C

在系统默认的安装目录下,使用下面的 DOS 命令进入 Turbo C。

C:\>cd tc (回车) (将当前目录改为 C 盘下的 tc 子目录)

C:\>tc (回车) (执行 tc.exe 文件)

(2)由 Windows 平台进入 Turbo C

方法一:单击“开始”→“运行”菜单,在弹出的对话框中键入 command 后,单击“确定”按钮,进入 DOS 方式。用上述的命令方式即可进入 Turbo C 环境。

方法二:在“我的电脑”或“资源管理器”中找到相应的文件夹 TC,双击其中的 tc.exe,即可进入 Turbo C 环境。

2. 操作界面

Turbo C 2.0 操作界面如图 1.1 所示。

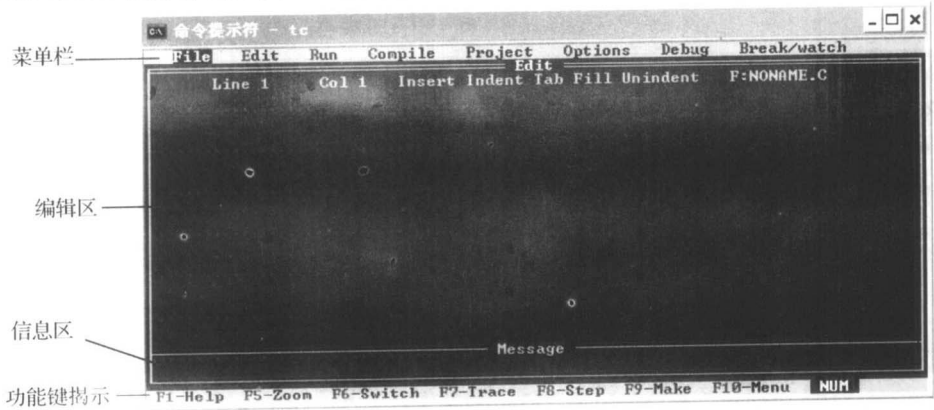


图 1.1

其中,菜单栏包括 File(文件)、Edit(编辑)、Run(运行)、Compile(编译)、Project(项目)、Options(选项)、Break/watch(断点/监视)。每一个主菜单还有相应的子菜单,通过菜单可实现相应的操作。

编辑区的作用是对 Turbo C 源代码进行输入和编辑。其上方有 Line 1 和 Col 1,表示当前光标的位置。在该行的最右边显示当前正编辑的源程序文件名(默认的文件名是 NOMANE.C)。

信息区,用英文 Message 表示,用于显示程序调试过程中的一些信息。

3. Turbo C 2.0 操作热键

所谓的热键就是能够立即完成某一功能的键。Turbo C 2.0 操作热键如表 1-1 所示。

表 1-1 Turbo C 2.0 操作热键

热键	功能	热键	功能
F1	显示当前位置的帮助信息	Alt + F3	选取文件
F2	保存编辑程序里的当前文件	Alt + F5	切换 TC 主屏幕和用户屏幕
F3	装入一个文件	Alt + F7	定位上一个错误
F4	程序运行到光标所在行	Alt + F8	定位下一个错误
F5	放大/缩小活动窗口	Alt + F9	将源程序编译成 .OBJ 文件

表 1-1(续)

热键	功能	热键	功能
F6	切换活动窗口	Alt + C	进入 Compile 菜单
F7	调试模式下运行程序,跟踪进函数内部	Alt + D	进入 Debug 菜单
F8	调试模式下运行程序,跳过函数调用	Alt + E	进入 Edit 菜单
F9	执行 make	Alt + F	进入 File 菜单
F10	切换菜单和活动窗口	Alt + H	进入 Help 菜单
Ctrl + F1	调用与函数相关的帮助信息	Alt + O	进入 Options 菜单
Ctrl + F2	复位运行的程序	Alt + P	进入 Project 菜单
Ctrl + F4	计算表达式	Alt + R	进入 Run 菜单
Ctrl + F7	增加一个监视表达式	Alt + S	进入 Search 菜单
Ctrl + F8	置断点 On 或 Off	Alt + W	进入 Windows 菜单
Ctrl + F9	运行程序	Alt + X	退出 TC, 返回到 DOS。
Alt + F1	调出所参考的最后一个帮助屏幕		

1.1.3 C 语言程序上机调试步骤

1. 编辑

如图 1.2 所示,按下 Alt + F 组合键,选择子菜单 New,按下回车键进入编辑状态(或者按下 F10 键激活 File 菜单,按回车键打开其子菜单)。此时光标定位在窗口左上角即第一行、第一列。编辑过程中,可用“Ins”键转换插入/改写状态和“Del”键删除字符。

2. 保存

按下 Alt + F 组合键或按下 F10 键,然后回车,选择 Save 项,如图 1.3 所示。

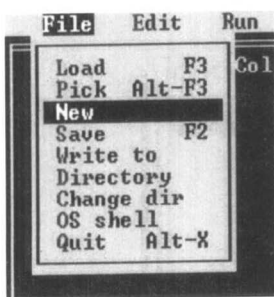


图 1.2

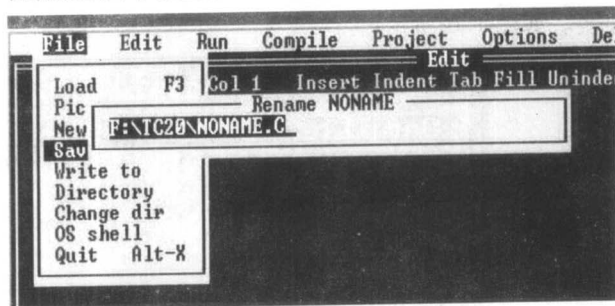


图 1.3

用户可以按自己的习惯指定文件名,然后,此源文件名存在用户工作目录中。如果用户想要改变工作目录,可以选择如图 1.4 所示 Change dir 项。

3. 打开文件

按下 Alt + F 组合键,选择 Load 项,这时屏幕上会出现一个 *.c 的“Load File Name”(打开文件名)对话框,如图 1.5 所示。用户在此处输入要打开的文件名,按下回车键后可进入编辑状态。

4. 编译和连接

源文件在编辑好了以后,应对源程序进行编译(compile)、连接(link),生成可执行文件。

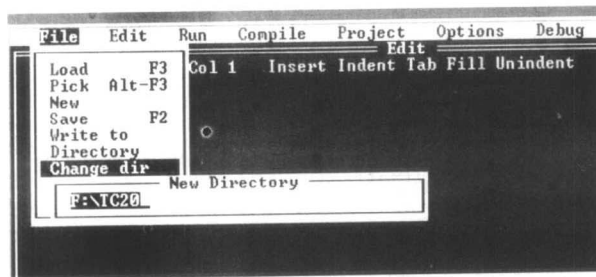


图 1.4

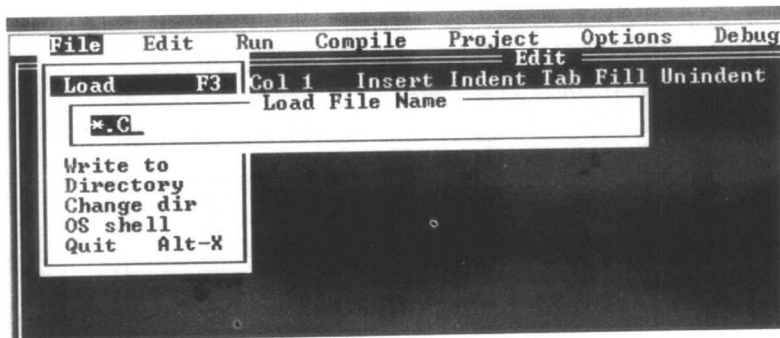


图 1.5

可以使用以下方法之一。

(1) 按下 Alt + C 组合键, 打开 Compile 菜单, 将光标移到 Compile to OBJ, 按下回车键就可进行编译。编译完成后, 再选择 Compile 中的 Link EXE file 项, 按下回车键, 就可以执行连接了。或者打开 Compile 菜单, 从中选择 Make EXE file 项, 按下回车键后一次完成编译和连接。如图 1.6 所示。

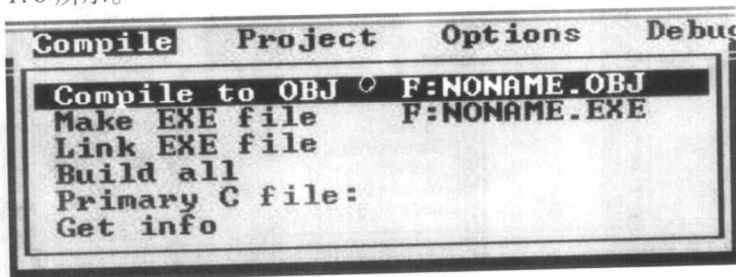


图 1.6

(2) 按 F9 键可一次完成编译和连接。

5. 运行

(1) 在 DOS 环境下输入所生成的 .exe 文件的路径和文件名后, 按下回车键即可运行。

(2) TC 集成环境下运行。按 F10 键, 选择 Run 菜单中的 Run 命令, 按下回车键可运行。

(3) 直接按下 Ctrl + F9 组合键即可运行。此方法是将编译、连接和运行一起完成。按下 Alt + F5 组合键可观看运行结果, 按任一键可返回编辑状态。

1.1.4 上机操作实例

编写程序,在屏幕上输出一串字符:This is a c program.

1. 启动 Turbo C2.0。
2. 按下 Alt + F, 打开 File 菜单, 从中选择 New 项。
3. 在编辑区光标所在处开始输入以下源代码:

```
main( )
{
    printf ( " This is a c program. \n" );
}
```

4. 按下 Alt + F, 打开 File 菜单, 从中选择 Save, 输入一个新文件名 aaa. c, 回车确认。
5. 按下 Ctrl + F9 键运行。
6. 按下 Alt + F5 键查看运行结果。

1.2 程序动态调试方法



1.2.1 按步执行方法

程序的编译和连接没有错误不等于运行时一定正确。编译系统能检查出语法错误, 但无法检查出逻辑错误。这里介绍程序动态调试方法——按步执行方法。这种方法的特点是程序一次执行一行。每执行一行后, 就停下来, 用户可以检查此时各有关变量和表达式的值, 以便发现问题所在。

功能键介绍如下:

F7 键。按步执行程序, 每按一次, 执行一句语句。

Ctrl + F7 组合键。观察变量和表达式的值。

以下面程序为例说明按步执行方法的使用。

```
#include <stdio.h>
main( )
{
    float a,b,c;
    a = 2.5;
    b = 3.6;
    c = a + b;
    printf ( "c = %d\n", c );
}
```

输出结果为:c=0

这个源程序没有任何语法错误,能顺利通过编译和连接。

这个结果显然不对。这时可以采用按步执行的方法,检查每一步的正确性。

(1)重新开始运行,按 F7 键,可以看到编辑窗口中源程序的主函数 main() 处用高亮显示,表示准备进入 main() 函数。同时可以看到屏幕下部的 Message 窗口变成了 Watch 窗口,它的功能是观察数据用的。如图 1.7 所示。

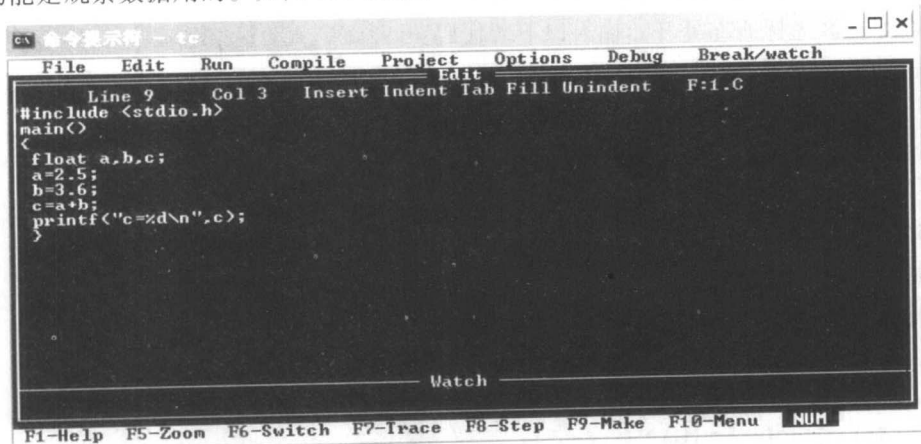


图 1.7

(2)再次按下 F7 键,亮条移到程序的第 5 行,表示已进入 main() 函数和左花括号,但并未执行本行语句,只是表明下一步要执行此行语句。

按回车键后,屏幕切换到编辑窗口,亮条移到第 6 行,表示第 5 行语句已执行完毕。再按一次 F7 键,亮条移到第 7 行,表示第 6 行中的表达式语句执行完毕。

(4)此时,可以检查一下有关变量的值是否正确。按下 Ctrl + F7 组合键,弹出一个 Add Watch 对话框。如果想看变量 a 的值,就在此框中输入 a,如图 1.8 所示,按下回车键后,该对话框消失,在屏幕下部的 Watch 窗口显示出 a 的当前值 2.5。再按下 Ctrl + F7 组合键,输入 b,按下回车键,同上方法输入 c,可看到变量的值,如图 1.9 所示。

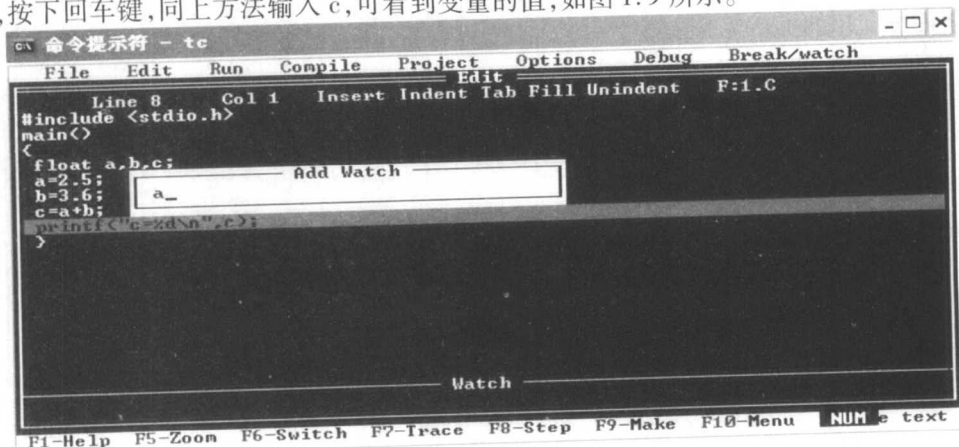


图 1.8

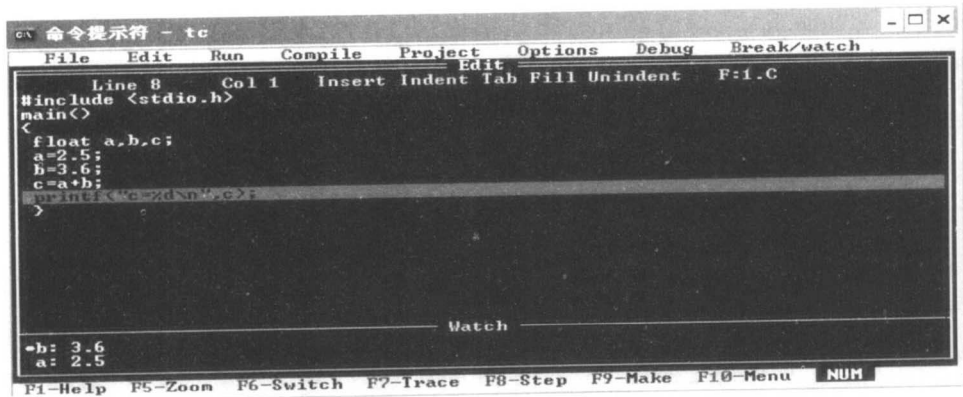


图 1.9

(5)再一次按下 F7 键,亮条移到第 8 行,表示第 7 行中的 $c = a + b$ 表达式已执行完毕。此时,按 Ctrl + F7 组合键,查看变量 c 的值,在 Watch 框内输入 c 的值,则其中显示出 c 的值是 6.1,而不是 0。由于 printf 语句执行完毕,应当在用户屏幕上有输出,按 Alt + F5 组合键观察屏幕上的显示,发现输出结果为 $c = 0$ 。至此,可确定为 printf 语句有问题。经检查,发现输出格式符用的是输出整数的格式符 %d。将 %d 改为 %f,再运行程序,结果为正确。

1.2.2 设置断点方法

对于一个较长的程序,常用的调试方法是在程序中设置若干个断点,程序运行到断点时暂停,用户可以检查此时有关变量或表达式的值。如果未发现错误,就使程序继续执行到下一个断点,如此一段一段的检查。实质上这种方法是把一个程序分割成几个分区,再逐区检查有无问题,若为有问题分区,再在该分区内设若干个断点,将其分成几个小区,然后再寻找有错的小区。这样不断缩小找错范围,直至找到出错点。

在按步执行方法或设置断点方法找错过程中,还可以使用 TC 的菜单 Debug 下的子菜单 Evaluate 提供的调试工具,等同于按 Ctrl + F4 组合键。它不仅可以看到变量或表达式的值,还可以修改它们的值,以帮助用户调试程序。

功能键介绍如下:

Ctrl + F8 组合键。设置/取消断点。

Ctrl + F4 组合键。查看/修改变量的值。

以如下程序为例说明断点设置的方法。

【例 1-1】 求解一元二次方程。

分析:对于一个一元二次方程 $ax^2 + bx + c = 0$,其根有两个: x_1 和 x_2 。

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$
 C 语言程序可以将其写成如下表达式:

$$x_1 = (-b + \text{sqrt}(b * b - 4 * a * c)) / 2$$

$$x_2 = (-b - \text{sqrt}(b * b - 4 * a * c)) / 2$$

程序如下:

#include <math.h>