



# 程序设计 实验教程

第二版

夏宝岚 夏 耘 张慕蓉/编著



华东理工大学出版社  
EAST CHINA UNIVERSITY OF SCIENCE AND TECHNOLOGY PRESS

# C 程序设计实验教程

(第二版)

夏宝岚 夏 耘 张慕蓉 编著



华东理工大学出版社

图书在版编目(CIP)数据

C 程序设计实验教程/夏宝岚,夏耘,张慕蓉编著.

2 版—上海:华东理工大学出版社,2004.1

ISBN 7-5628-1468-6

I. C… II. ①夏… ②夏… ③张… 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(2003)第 106515 号

C 程序设计实验教程

(第二版)

夏宝岚 夏耘 张慕蓉 编著

出版	华东理工大学出版社	开本	787×1092	1/16
社址	上海市梅陇路 130 号	印张	10.75	
邮编	200237 电话(021)64250306	字数	253 千字	
网址	www.hdlgpress.com.cn	版次	2004 年 1 月第 2 版	
发行	新华书店上海发行所	印次	2004 年 1 月第 1 次	
印刷	上海崇明裕安印刷厂	印数	1-6550 册	

ISBN 7-5628-1468-6/TP·123

定价:16.00 元



## 内 容 简 介

本书全篇由四个部分组成。第一部分为基础知识,对 C 程序的开发环境以及如何 Turbo C 2.0 编译系统上编辑、编译、链接、运行 C 程序的全过程作了详细而清晰的介绍,作为学习和使用 C 语言的基础。第二部分为实验操作,共安排了 13 个实验,其内容、节奏与“C 程序设计”课程的理论教学完全同步。第三部分为应用技巧,介绍了初学者在编程中常犯的近 30 种错误及其排错方法,调试程序的几点技巧以及提高程序运行效率的几点建议,这部分内容对提高学习者的排错、纠错能力和程序设计水平有极大的帮助。最后一部分为附录,其中附录 A 列出了详尽的编译出错信息,并同时给出了造成出错的可能原因以及排除的方法;附录 C 给出了全书的实验参考答案,为学生的独立自学提供了极大的帮助与启发。

本书既适合作高等院校“C 程序设计”课程配套的实验指导书,又可用于《C 程序设计》的参考书,同时也可用作初学者学习相关知识和技能的启蒙教材。

# 前言

C 程序设计语言作为第二代面向过程的程序设计语言中的佼佼者，在国内得到了广泛的应用，已被社会所公认。尤其是全国的各大高校已将“C 程序设计”这门课程作为非计算机专业学生的必修课，作为大学生获取程序设计能力的首选课程。但是相对于其他一些常用的程序设计语言来说，C 语言具有内容多、概念复杂、不易掌握的特点，总有相当一部分学生在学到结构、指针、链表等难度较高的内容时会感到不易理解，有学习困难。尤其对毫无程序设计基础的初学者来说，在整个学习过程中总有一种无所适从的感觉，害怕编程。为了帮助他们尽快了解 C 语言，掌握 C 语言的学习方法，提高程序设计的能力，作者编写了本书。

本书的第一个特点是：在给出每个实验具体的操作内容之前，作者都全面、透彻地阐述了与本次实验有关的基本概念和知识要点。学习者阅读以后，将会感到不仅对顺利完成本次实验操作大有帮助，而且也起到了对本章节所学的理论知识进行全面、系统地复习与回顾的作用。第二个特点是：作者突破了传统的实验题型的框架，并非就事论事地给出几道枯燥的、毫无启发性的题目，让学习者自己去编制程序，而是采用了丰富多样的形式，其中有：给出现成的程序让学习者模仿，给出一个简单的程序让学习者去扩充功能，给出一个有错的程序让学习者纠错改错，给出一个功能不全的程序让学习者填空补缺，给出一道题目和程序说明让学习者独立编程，以及考察学习者综合运用能力的不同档次的习题。第三个特点是：在实验题的选择与编排上遵循了深入浅出、由简至繁的原则，可使读者在阅读和练习时能循序渐进、轻松自如。除此之外，本教程还对 Turbo C 的集成环境，程序的编辑、调试、编译、运行等一系列操作方法，作了较全面的介绍。因此本教程既是一本理想的实验指导书，又是一本有价值的《C 程序设计》的参考书。

本书在编写过程中得到了华东理工大学信息学院副院长宋国新、计算机系主任邵志清、计算机教学实验中心主任龚正良、网络中心主任黄建华等诸位教授以及留美博士龚骏同学的热情指导与帮助，在此一并表示衷心的感谢。

编者

2003 年 12 月

# 目录

## 1 基础知识

- 1.1 C 程序的开发环境 ..... (1)
  - 1.1.1 Turbo C ..... (1)
  - 1.1.2 Visual C + + ..... (1)
- 1.2 怎样使用 Turbo C 2.0 编译系统 ..... (2)
  - 1.2.1 Turbo C 的启动与退出 ..... (2)
  - 1.2.2 TC 主屏幕的组成 ..... (3)
  - 1.2.3 TC 主菜单 ..... (4)
  - 1.2.4 系统状态的切换 ..... (9)
- 1.3 初次在 Turbo C 2.0 系统上编程 ..... (10)
  - 1.3.1 编辑、编译、运行 C 程序的操作步骤 ..... (10)
  - 1.3.2 从 TC 集成环境暂时返回 DOS 操作系统 ..... (13)
  - 1.3.3 Turbo C 2.0 常用的编辑操作 ..... (13)
  - 1.3.4 Turbo C 2.0 常用的热键 ..... (13)
- 1.4 在 Visual C + + 系统上编程 ..... (16)
  - 1.4.1 Visual C + + 6.0 窗口的组成 ..... (16)
  - 1.4.2 编辑、编译、运行 C + + 程序的操作步骤 ..... (17)
  - 1.4.3 如何在 VC 集成环境下编辑、编译、运行 TC 程序 ..... (19)

## 2 实验操作

- 2.1 Turbo C 操作初步(实验一) ..... (22)
- 2.2 设计并运行简单的 C 程序(实验二) ..... (25)
- 2.3 输入与输出(实验三) ..... (29)
- 2.4 选择结构程序设计(实验四) ..... (33)
- 2.5 循环结构程序设计(实验五) ..... (40)
- 2.6 数组(实验六) ..... (48)
- 2.7 函数(实验七) ..... (54)
- 2.8 编译预处理(实验八) ..... (65)
- 2.9 指针(实验九) ..... (70)
- 2.10 结构(实验十) ..... (78)
- 2.11 文件(实验十一) ..... (88)

---

2.12 图形处理(实验十二)	(95)
2.13 综合练习(实验十三)	(99)
<b>3 应用技巧</b>	
3.1 初学者编程中常犯的错误	(107)
3.2 调试程序的几点技巧	(114)
3.2.1 程序的单步执行	(114)
3.2.2 观察程序执行过程中变量和表达式值的变化	(115)
3.2.3 修改数据对象的值	(115)
3.2.4 设置断点	(115)
3.2.5 执行到光标所在行	(116)
3.3 有关程序效率的几个问题	(116)
3.3.1 ++ 和 -- 运算	(116)
3.3.2 使用寄存器变量	(116)
3.3.3 使用指针处理数组下标	(116)
3.3.4 合理使用函数	(117)
<b>附录</b>	
附录 A 编译出错信息	(118)
附录 B 常用字符与 ASCII 码对照表	(133)
附录 C 实验参考答案	(134)

# 1 基础知识

本章主要介绍 C 程序的开发环境,介绍如何使用 Turbo C 2.0 编译系统以及在 Turbo C 2.0 系统上编程包括哪些步骤?并初次尝试编程。

## 1.1 C 程序的开发环境

C 语言的标准已为大多数 C 和 C++ 开发环境所兼容。在今天,我们可以使用相当多的工具来开发自己的 C 程序。例如,以下将要介绍的 Turbo C 和 Visual C++ 就是当今最流行的 C 程序开发工具。

### 1.1.1 Turbo C

Turbo C 编译系统是近几十年来最为流行的 C 程序开发环境。它以自身的小巧、易用和高效征服了绝大多数的 C 程序开发者,经历了时间与实践的考验,证实是一款优秀的 C 程序开发工具。

Turbo C 2.0 是基于 DOS 操作系统的,欲在磁盘上建立 Turbo C 系统,首先在硬盘上建立一个名字为 TC 的子目录,把 Turbo C 2.0 系统的全部文件都装入该子目录下即可运行该集成环境,但是为了便于管理和数据存取,在 TC 子目录下再建两个名字分别为 include 和 lib 的二级子目录,用于存储嵌入文件和库文件,这样用户就可方便地从集成环境版本转换到命令行版本。

### 1.1.2 Visual C++

近些年来,随着 C++ 语言的崛起,Visual C++ 开发环境也逐渐流行起来,它在开发程序的易用性方面又上了一个台阶,为我们又提供了一个基于 Windows 操作系统的 C 程序开发环境。

Visual C++ 是基于 Windows 和 Windows NT 32 位应用程序的可视化开发工具,用它能够开发出规模更大、功能更复杂的 32 位 Windows 应用程序,包括数据库应用程序、Internet 应用程序和 Active X 应用程序等。

在此必须指出,Visual C++ 开发环境内容相当多,功能十分繁杂,使用者必须具有一定的计算机基础和程序设计基础;与此相比,Turbo C 具有小巧、简单、易学、编译速度快、目标程序效率高的优点,对于初学者来说,从 Turbo C 入手学习 C 程序设计是一种明智的选择。所以本书将主要对 Turbo C 2.0 编译系统作详细介绍,对 Visual C++ 仅作最简单的使用介绍。



## 1.2 怎样使用 Turbo C 2.0 编译系统

### 1.2.1 Turbo C 的启动与退出

#### 1) 启动

如果已经对 tc.exe 文件所在的路径作了设置,则在 DOS 状态下直接输入 TC 并按回车键即可启动 Turbo C。

例如:

c:\> tc <enter>

直接在 C 盘根目录下启动 TC

H:\STD\980366> tc <enter>

在网络盘的二级子目录 980366 下启动 TC

c:\> d:\tc\tc <enter>

以给出路径全称的方法启动 TC

这里的<enter>表示按一次回车键。

TC 启动以后,屏幕显示 TC 主屏幕和版本信息(见图 1.1),按任意键即关闭之。

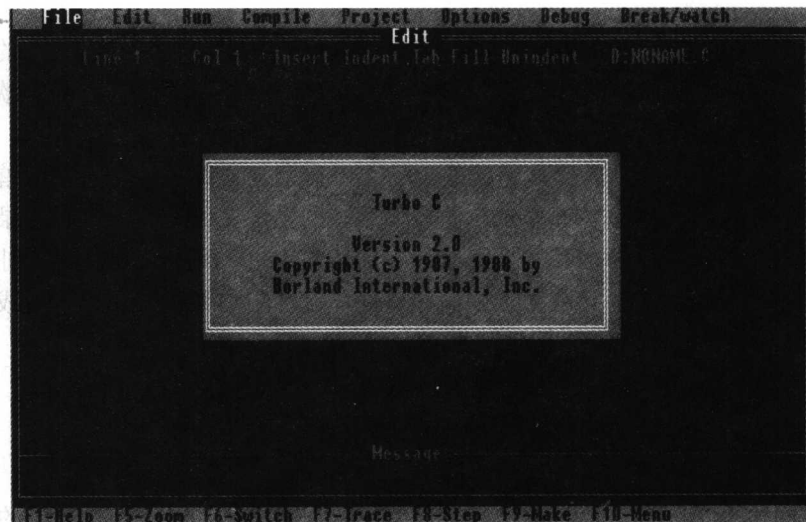


图 1.1 TC 主屏幕

#### 2) 退出 TC

打开 File 菜单,选择 Quit 命令或直接按 Alt-X 热键即可退出 TC 集成环境,返回操作系统。如果当前编辑的源程序文件尚未存盘,在退出之前,系统将打开如图 1.2 所示的对话框请求保存确认,此时可选择按 Y 或 N 键以确认存盘或放弃。

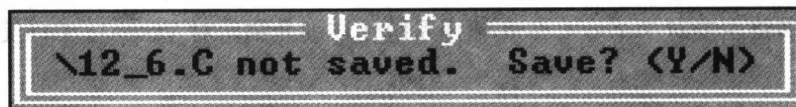


图 1.2 保存确认对话框



### 1.2.2 TC 主屏幕的组成

Turbo C 2.0 集成环境的主屏幕由上至下分成 4 个部分，它们是：主菜单、编辑窗口、消息窗口和热键提示行，各部分的功能如下。

#### 1) 主菜单

为用户提供了用于文件操作的一切命令。

#### 2) 编辑窗口

用于输入和编辑源程序，按 F10 功能键或按 ALT-E 热键或单击 EDIT 菜单并按回车键均可激活编辑光标进行源程序的输入和编辑，此时窗口顶部将显示与当前编辑状态有关的信息行，形式如下所示：

```
line×× col×× Insert Indent Tab Fill Unindent ××.C
```

line	××	当前光标所在行；
col	××	当前光标所在列；
Insert		插入状态，按此键可以在插入状态与替换状态之间切换；
Indent		首行自动缩进。即输入的新行自动与上一行的行首对齐，使用 Ctrl-O-I 热键可控制 on 和 off 的切换；
Tab		制表模式。on 时，按一次向右跳 8 格，使用 Ctrl-O-T 热键可控制 on 和 off 的切换；
Fill		填充模式。当 Tab 为 on 时，每行的开始处自动填入适当的制表符和空格，使用 Ctrl-O-F 热键可控制 on 和 off 的切换；
Unindent		对齐模式，按一次退格键（backspace 键），使当前行对齐上一行的第一个非空字符，使用 Ctrl-O-U 热键可控制 on 和 off 的切换；
××.C		当前编辑文件所在的路径和名字。

#### 3) 消息窗口

用于显示系统提示的信息，例如，编译时显示编译出错信息，调试程序时显示监视表达式的值等等。

#### 4) 热键提示行

为用户提供了一组快速执行菜单命令的功能键。热键提示行的内容随活动光标所处的位置不同而有所不同，当活动光标位于菜单或编辑窗口内时，提示行内容如下：

F1 (help)	打开帮助窗口，获取有关编辑命令的有关信息；
F5 (zoom)	放大或缩小当前活动窗口；
F6 (switch)	光标在编辑窗口与活动窗口之间切换；
F7 (trace)	单步执行程序，跟踪到函数；
F8 (step)	单步执行程序，不跟踪到函数；
F9 (make)	编译、链接生成 .exe 文件；

F10 (menu) 光标在主菜单与编辑窗口之间交替切换。

当活动光标位于监视窗口中时，热键功能如下：

F1 打开 help 窗口，获取如何使用错误跟踪的帮助信息；  
 F5 将消息窗口扩大至全屏幕或还原；  
 F6 激活编辑窗口；  
 INS 向 watch 窗口内增加一个监视表达式；  
 DEL 向 watch 窗口内删除一个监视表达式；  
 ENTER 编辑当前的监视表达式。

### 1.2.3 TC 主菜单

Turbo C 2.0 集成环境的主菜单共有以下 8 个菜单栏目，见表 1.1。

表 1.1 Turbo C 2.0 菜单组成

菜单名称	功能
File	建立、打开、保存文件，管理目录，暂转 DOS 和退出 Turbo C
Edit	建立、编辑源文件
Run	编译、链接和运行程序
Compile	对源程序文件进行编译，生成目标文件和可执行文件并查错报错
Project	生成并维护由多个源程序文件、目标文件组合而成的可执行文件
Options	设置集成环境的配置和工作方式
Debug	设置各种调试选择项，包括检查、修改变量和表达式的值，查找函数，程序编译时在执行代码中插入调试信息等
Break/Watch	设置和删除断点，增加、删除和编辑监视表达式等

每个菜单栏目（除 Edit 外）又都包含一个下拉式子菜单，下面对它们作更详细的介绍。

#### 1) File 菜单

File 菜单包含的下拉式子菜单如图 1.3 所示，菜单中各命令项目的功能如下：

Load 将指定的文件从磁盘调入内存，允许使用类似 DOS 的通配符来进行列表选择，执行该命令会有对话框出现，要求用户输入文件名，如果输入的文件名磁盘上并不存在，就表示建立一个新文件。本命令的热键为 F3。  
 Pick 显示最近编辑过的 8 个 C 文件清单，用户可直接单击其中的某个文件名，将其装入内存。本命令的热键为 ALT-F3。  
 New 选择该命令，将编辑一个新文件，默认的文件名为 NONAME.C。  
 Save 将当前正在编辑的文件存盘，假如此时的文件名为 NONAME.C 的话，系统会提示是否要改名，并显示对话框让你输入文件名。本命令的热键为 F2。  
 Write to 将当前编辑的文件写到新的文件名下或覆盖一个已存在的老文件，执行该

- 命令，屏幕将显示一个空白的对话框，等待输入文件名。
- Directory** 显示指定的目录及文件清单，若直接按 Enter 键，将显示当前盘、当前路径的目录和文件。
- Change dir** 显示当前目录，若再键入不同的驱动器和目录并按回车键，则改变当前目录路径。
- Os shell** 暂时脱离 Turbo C，返回 DOS 系统，若再在 DOS 提示符键入 Exit 命令则又返回 Turbo C，当需要运行 DOS 命令而又不想退出 Turbo C 时，本命令十分有用。
- Quit** 退出 Turbo C，返回 DOS 系统。本命令的热键为 ALT-X。

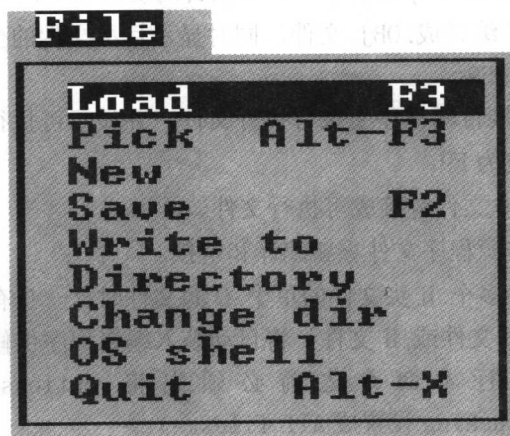


图 1.3 File 菜单

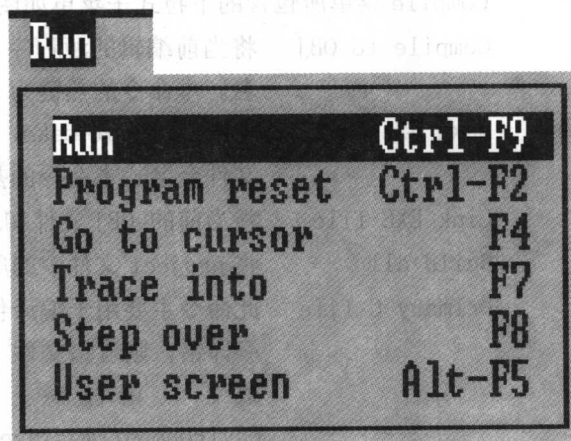


图 1.4 Run 菜单

## 2) Edit 菜单

单击该菜单名，直接激活编辑光标，对当前已调入内存的文件进行全屏幕编辑。

## 3) Run 菜单

Run 菜单所包含的下拉式子菜单如图 1.4 所示，菜单所包含的各命令项目的功能如下：

- Run** 运行用户程序，若执行此命令时还未生成可执行文件，就自动对源程序先编译、链接，然后再运行。本命令的热键为 Ctrl-F9。
- Program reset** 终止程序的调试，释放用户程序占用的内存空间，关闭所有已打开的文件。本命令的热键为 Ctrl-F2。
- Go to cursor** 使程序从执行的亮条处（若没有执行亮条，就从开始处）执行到光标所在行；若光标所在行之前有断点，就执行到断点处；若所给的执行范围内没有可执行代码，本命令将显示警告信息，并提示按 ESC 键返回。本命令主要用于程序调试，其热键为 F4。
- Trace into** 单步执行程序，若遇函数调用，自动进入函数跟踪（但该函数必须是

调试程序可以访问的，也就是函数所在的源文件必须在 options/compiler/code generation/obj debug information 和 debug/source debugging 设置为 ON 时进行编译)，以后的 Trace into 或 Step over 命令就运行在函数的定义里。调试程序离开函数时，再恢复到函数调用的那条语句。本命令的热键为 F7。

- Step over 单步执行程序，若遇函数调用，不进入函数跟踪。本命令的热键为 F8。
- User screen 从 TC 主屏切换到用户屏幕，查看程序的运行结果，按任意键返回 TC 主屏幕。本命令的热键为 ALT-F5。

#### 4) Compile 菜单

Compile 菜单所包含的下拉式子菜单如图 1.5 所示，菜单中各命令项目的功能如下：

- Compile to OBJ 将当前编辑的源程序文件编译成 .OBJ 文件，同时显示目标文件的全名。本命令的热键为 ALT-F9。
- Make EXE file 调用 project-make，直接将源程序生成可执行文件，并显示可执行文件全名。本命令的热键为 F9。
- Link EXE file 将当前的 .OBJ 文件和 .LIB 文件链接成可执行文件。
- Build all 将 project 文件中的所有源程序文件重新编译和链接。
- Primary C file 此命令主要用于编译包含多个 .H 头文件的单个 .C 源文件时，一旦有编译错，则含错文件 (.C 文件或 .H 文件) 将自动装入编辑程序供修改。（请注意，执行本命令之前必须先 will options/Environment/message Tracking 设置成 All files。）
- Get info 执行此命令，屏幕弹出一个窗口，显示上次编译或运行程序的所有信息。包括当前子目录名，源文件名，文件大小，是否使用 EMS，已编译的行数，错误个数等。

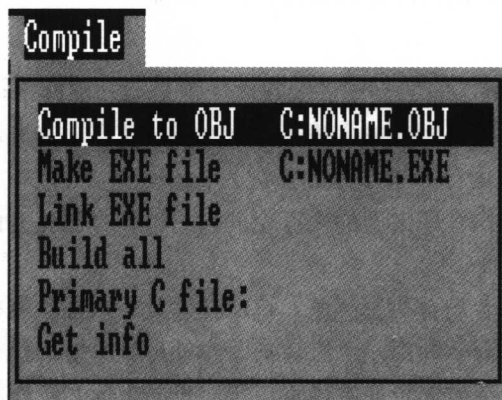


图 1.5 Compile 菜单

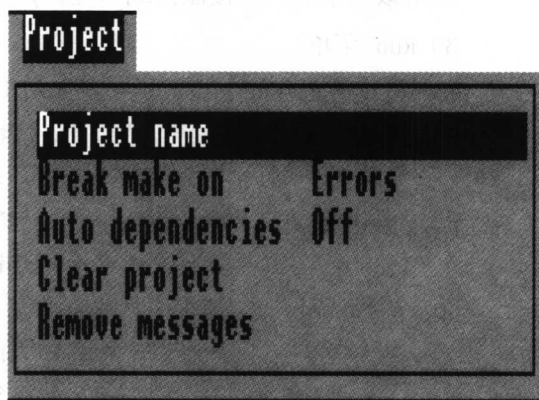


图 1.6 Project 菜单



### 5) Project 菜单

Project 菜单主要用于由多个 C 文件组成的程序。一个相当大的 C 程序放在一个文件中编辑是很不方便的,对程序哪怕作再小的一点改动,整个程序都要重新编译,解决的办法是分而治之,将程序分为若干模块,先分块编辑、编译,再链接组装。在 C 中,该组装文件就称为项目“Project”。

Project 菜单所包含的下拉式子菜单如图 1.6 所示,菜单中各命令项目的功能如下:

Project name	指定一个扩展名为 .prj 的 project 文件名。
Break make on	由用户指定终止编译的缺省条件,通常为:警告时 (Warning)、有错误时 (Errors)、有致命错误时 (Fatal Errors) 或链接之前。
Auto dependencies	这是一个开关命令,置为 ON 时,project-make 将自动检查项目组中的每一个 .obj 文件是否与 .c 文件相一致,若二者的日期或时间不一致,重新编译.c 文件。
Clear project	去除按 project 文件编译的方式。
Remove messages	清除消息窗。

### 6) Options 菜单

Options 菜单所包含的下拉式子菜单如图 1.7 所示,菜单中各命令项目的功能如下:

Compiler	提供一个设置各种编译参数的子菜单,包括 Model (存储模式), Defines (宏定义), Code generation (代码生成), optimization (代码优化) 等 8 条命令。
Linker	提供一个设置各种连接参数的子菜单,包括 Map file (选择 map 文件的类型), Initialize segments (段初始化), Default libraries (缺省库) 等 7 条命令。
Environment	提供一个对集成环境设置各种参数的子菜单,包括 Message Tracking (信息跟踪,即滚动消息窗口的同时,跟踪编辑窗口中的语法错误), Config auto save (自动保存配置), Backup file (备份文件) 等 8 条命令。
Directories	设置 TC 操作路径,包括 include directories (包含文件目录), Library directories (库目录), Output directories (生成的 .obj 和 .exe 文件存储目录,简称输出目录) 等 6 个命令。
Arguments	用户通过该命令输入运行程序命令行的参数(即设置程序运行时需要的参数),与 DOS 状态下使用命令行参数一样。
Save options	将 option 菜单中所有已作的设置保存到配置文件中(缺省的配置文件为 Tcconfig.tc)。以后启动 TC 时自动寻找 Tcconfig.tc 文件,完成各种设置。
Retrieve options	加载以前保存过的配置文件。

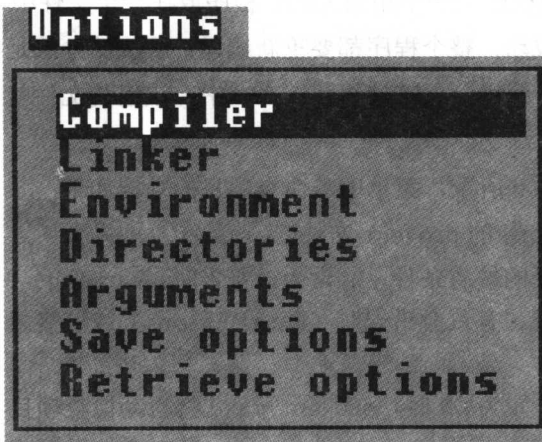


图 1.7 Options 菜单

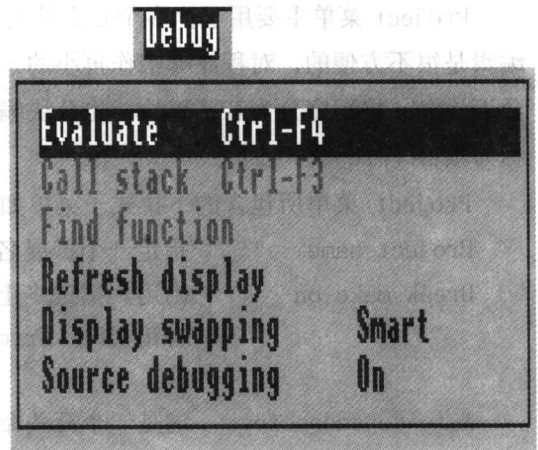


图 1.8 Debug 菜单

### 7) Debug 菜单

Debug 菜单所包含的下拉式子菜单如图 1.8 所示，菜单中各命令项目的功能如下：

- |                  |  |
|------------------|--|
| Evaluate         | 计算并显示变量或表达式的值，但表达式中不允许出现函数调用，用#define 或 typedef 定义的符号和宏以及不在当前执行函数中出现的局部变量或静态变量。本命令的热键为 Ctrl-F4。                          |
| Call stack       | 调用堆栈，以显示到目前为止函数的调用情况。MAIN 函数在栈底，正在运行的函数在栈顶，每项显示了调用的函数名和传递给它的参数值。本命令的热键为 Ctrl-F3。   |
| Find function    | 查找用户程序中的函数，但该程序不应在编译时将 Debug/Source Debugging 和 option /compiler/ code generation/obj debug information 设置为 ON。本命令常用于调试。 |
| Refresh display  | 恢复被重写了的屏幕内容。   |
| Display swapping | 设置显示转换方式。Smart（缺省值）时，若正在执行的代码产生屏幕输出或调用一个函数，就从编辑屏幕切换到用户屏幕，完成输出以后再切换回编辑屏幕；Always 时，执行每一条语句都切换；None 时，任何情况下都不切换屏幕。          |
| Source debugging | 设置源代码调试方式。ON 时，链接的程序能同时用 TC 集成调试程序和独立的 Turbo C 调试程序调试；Standalone 时，就只能用 Turbo C 调试程序调试了；None 时两者都不能用。                    |

### 8) Break/watch 菜单

本菜单用于设置断点和添加、删除监视表达式。其下拉式子菜单如图 1.9 所示，菜单中各命令项目的功能如下：

- |           |                           |
|-----------|---------------------------|
| Add watch | 向监视窗口中输入一个监视表达式，缺省表达式为编辑窗 |
|-----------|---------------------------|

Delete watch	口中光标处的单词，若表达式正确，监视窗口就显示表达式的值。本命令的热键是 Ctrl-F7。
Edit watch	删除监视表达式。注意：只有当监视窗口可见时（也即编辑窗口不能放大时）才能执行本命令。
Remove all watches	编辑监视表达式。执行本命令，调试程序给出一个显示当前监视表达式的拷贝窗口。修改后，按回车键确认，按 ESC 键取消。
Toggle breakpoint	从监视窗口中删除所有的监视表达式。
Clear all breakpoints	设置或取消光标处的断点。
View next breakpoint	清除程序中的所有断点。
	对断点进行光标定位，亦即执行本命令，按断点设置的顺序将光标移到下一个断点处。

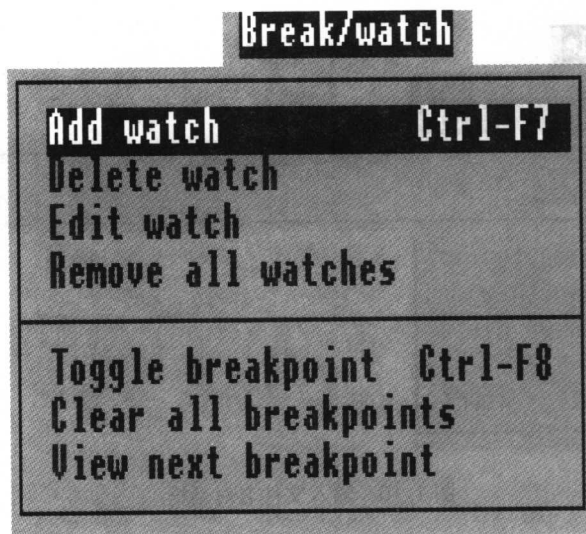


图 1.9 Break/watch 菜单

#### 1.2.4 系统状态的切换

- (1) 按功能键 F10，光标就在菜单和编辑窗口之间切换。
- (2) 当菜单处于打开状态时，按 ESC 键，则关闭该菜单并退回上一级菜单或返回当前活动窗口。



## 1.3 初次在 Turbo C 2.0 系统上编程

对 TC 集成环境有了较全面的了解以后, 这里以一个简单程序的编制过程感性认识一下如何在 Turbo C 2.0 系统上完成程序的编辑、编译、链接、运行以及源程序和目标程序的存储等任务。

### 1.3.1 编辑、编译、运行 C 程序的操作步骤

#### 1) 进入编辑状态

假设 TC 已经启动, 打开 File 菜单, 选择 Load 命令或直接按 F3 功能键, TC 主屏的中央弹出一个“Load File Name”对话框(如图 1.10 所示), 要求输入将要编辑的源程序的文件名, 可以输入文件全称, 例如 ch1\_2.c; 也可以省略扩展名.c, 仅输入主文件名, 例如 ch1\_2, 当输入并按回车键以后, 编辑窗口中的光标即被激活, 进入了编辑状态。

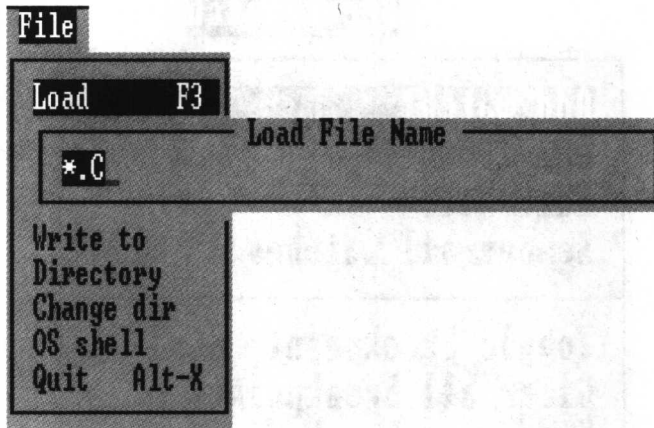


图 1.10 输入文件名对话框

#### 2) 编辑源程序

如果在上面的“Load File Name”对话框中输入的文件名磁盘上已存在, 该文件即被装入编辑窗口, 这表示将一个已经存在的老文件从磁盘调入内存, 你就可以对它进行各种编辑操作了; 若输入的文件名是新的, 则表示创建一个新文件, 此时, 编辑窗口如同一张白纸, 你可以将事先已编好的源程序输入到该窗口内。现假设输入以下简单程序:

```
main()  
{ char a, b;  
  printf("Please input two characters!");  
  scanf("%c%c", &a, &b);  
  printf("The first character is %c\n", a);  
  printf("The second character is %c\n", b);  
}
```