



C语言程序设计

上机指导与习题解答

苑鸿骥 钟文峰 主编
谭晓芳 晏志英 副主编

 人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

C 语言程序设计上机指导与习题解答 / 苑鸿骥, 钟文峰主编.

—北京: 人民邮电出版社, 2006.8

ISBN 7-115-15051-6

I. C... II. ①苑...②钟... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 084594 号

内 容 提 要

本书是《C 语言程序设计》的配套上机指导。

本书包括三部分内容。第一部分为 C 语言上机指南, 详细地介绍目前多数用户广泛应用的 Turbo C 集成环境的上机过程及出错信息。第二部分为上机实验安排, 提出上机实验的要求, 介绍程序调试和测试的初步知识, 具体安排 11 个实验, 便于进行实验教学。第三部分为 C 语言程序设计习题与参考解答, 包括教材《C 语言程序设计》一书的全部习题。对书中一些概念问答题, 由于能在教材中直接找到答案, 本书不另给出解答。对其他类型的习题一律给出参考答案。

本书可以作为高等院校 C 语言课程的教材, 也可以供报考计算机等级考试者和其他自学者参考。

C 语言程序设计上机指导与习题解答

◆ 主 编 苑鸿骥 钟文峰

副 主 编 谭晓芳 晏志英

责任编辑 赵慧君

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京铭成印刷有限公司印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 7.25

字数: 172 千字

2006 年 8 月第 1 版

印数: 5 001—7 000 册

2006 年 8 月北京第 2 次印刷

ISBN 7-115-15051-6/TP • 5580

定价: 18.00 元 (附光盘)

读者服务热线: (010)67170985 印装质量热线: (010)67129223

编者的话

C 语言是近年来国内外广泛使用的计算机语言，是程序开发人员必须掌握的一种语言，近十多年来，C 语言在国内得到了迅速的推广应用，许多高校和计算机培训班相继开设了“C 语言程序设计”课程。

在以往 C 语言的教学中比较强调理论知识的培养而忽视了上机调试的重要性，上机操作是理论联系实际的一个重要环节。在教学过程中应该注意课程的理论部分与实践部分在时间上的分配，考虑如何在有限的学时内学到更多有用的内容。为此我们精心为广大学习“C 语言程序设计”的读者编写了这本《C 语言程序设计上机指导与习题解答》。本书概念清晰、叙述详尽、例题丰富、通俗易懂，非常适合初学者使用。

本书包括三部分。第一部分为 C 语言上机指南，详细地介绍了目前多数用户广泛应用的 Turbo C 集成环境的上机过程及出错信息。第二部分为上机实验安排，提出了上机实验的要求，介绍了程序调试和测试的初步知识，并且具体安排了 11 个实验，便于进行实验教学。第三部分为 C 语言程序设计习题与参考解答，包括了教材《C 语言程序设计》一书的习题解答。对其中的一些概念问答题，由于能在教材中直接找到答案，本书不另给出解答。对其他类型的习题一律给出了参考答案。

本书由南昌理工学院苑鸿骥教授组织编写并主审全书，谭晓芳在审稿过程中做了大量的辅助工作。参加本书编写有：钟文峰、谭晓芳、晏志英、陈黎艳和陆志香。另外，张朋、孙素敏、刘肃平、黄栋和徐炳荣等同志对本书提供了帮助，在此一并表示感谢。

由于编者水平有限，加之时间仓促，书中错误在所难免，敬请广大读者提出宝贵意见和建议。

编 者

2006 年 7 月

目 录

第一部分 C 语言上机指南	1
1.1 Turbo C 的安装	1
1.2 进入 Turbo C 环境	2
1.3 “TC” 窗口	2
1.4 C 语言程序上机步骤	4
1.4.1 单程序文件的编辑、编译、链接与调试	4
1.4.2 大程序文件的编辑、编译、链接与调试	6
1.5 Turbo C 各菜单功能	8
1.5.1 主菜单	8
1.5.2 Turbo C 2.0 的配置文件	13
1.6 UNIX/Linux 环境下的 C 语言开发环境	13
1.6.1 建立 C 语言程序	13
1.6.2 Vi 编辑器的命令	15
1.6.3 编译系统简介	16
1.6.4 C 语言程序的建立	17
1.6.5 调试 C 语言程序	18
1.7 编译出错信息	21
第二部分 上机实验安排	27
2.1 实验指导	27
2.1.1 上机实验的指导思想和要求	27
2.1.2 关于程序的调试和测试	29
2.2 实验内容	30
2.2.1 实验一 C 语言程序的运行环境	30
2.2.2 实验二 数据类型、运算符和表达式	33
2.2.3 实验三 顺序结构程序设计	36
2.2.4 实验四 选择结构程序设计	39
2.2.5 实验五 循环结构程序设计	44
2.2.6 实验六 数组	48
2.2.7 实验七 函数	52
2.2.8 实验八 预处理命令	57
2.2.9 实验九 指针	59
2.2.10 实验十 结构体、共用体和枚举型	66
2.2.11 实验十一 文件	74

2.2.12 实验十二 综合程序设计	78
第三部分 《C 语言程序设计》习题与参考解答	89
3.1 C 语言概述	89
3.2 算法	90
3.3 数据类型、运算符和表达式	91
3.4 基本输入输出和顺序结构程序设计	93
3.5 选择结构程序设计	96
3.6 循环结构程序设计	97
3.7 数组	100
3.8 函数	102
3.9 预处理命令	105
3.10 指针	106
3.11 结构体、共用体和枚举型	108
3.12 文件	109

第一部分

C 语言上机指南

Turbo C (TC) 是由美国 Bland 公司研制生产的, 是一个集程序编辑、编译、链接和调试于一体的 C 语言程序开发软件, 它具有速度快、效率高、功能强、使用灵活方便等优点, 是编制 C 语言程序的最佳工具。

1.1 Turbo C 的安装

Turbo C 的安装步骤如下。

(1) Turbo C 2.0 的安装非常简单, 用户只需要将包含 install.exe 的软盘插入 A 驱动器中, 在 DOS 的“A: \>”下输入: A: \>INSTALL 即可, 此时屏幕上显示 3 种选择。

- ① 在硬盘上创建一个新目录来安装整个 Turbo C 2.0 系统。
- ② 对 Turbo C 1.5 更新版本。这样的安装将保留原来对选择项、颜色和编辑功能键的设置。
- ③ 为只有两个软盘而无硬盘的系统安装 Turbo C 2.0。

这里假设按第一种选择进行安装, 只要在安装过程中按对盘号的提示, 顺序插入各个软盘, 就可以顺利地进行安装, 安装完毕将在 C 盘根目录下建立一个 TC 子目录, TC 下还建立了两个子目录 LIB 和 INCLUDE, LIB 子目录中存放库文件, INCLUDE 子目录中存放所有头文件。

(2) 直接复制 Turbo C 2.0, 在 Windows 环境下建立 TC.EXE 的快捷方式, 双击即可运行 (即在 DOS 下直接运行 TC.EXE), 但在使用过程中会遇到如下问题。

- ① 找不到 stdio.h conio.h 等 include 文件。
- ② 出现 cos.obj 无法连接之类的错误。

这些问题是由于没有设置好路径引起的, 路径的设置方法如下 (如图 1.1 所示)。

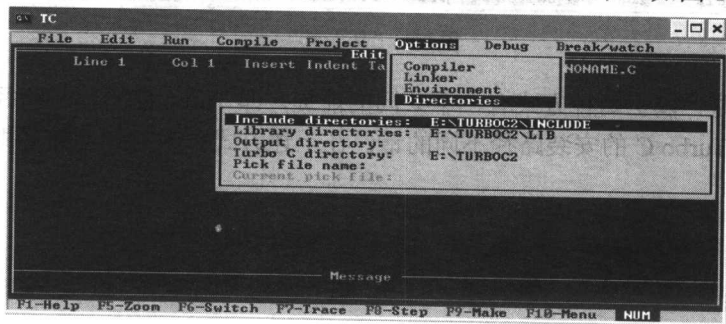


图 1.1 路径设置

在“TC”窗口中选择“Options”/“Directories”命令。

Include directories: [TC 所在的盘符]\TURBOC2\INCLUDE。

Library directories: [TC 所在的盘符]\TURBOC2\LIB。

Output dirextory: 输出目录请自己设置一个工作目录, 以免混在一起。

“Options”菜单下的 Save options: 保存以上的设置, 要求保存在与“Include”文件相同的文件夹下。

1.2 进入 Turbo C 环境

进入 Turbo C 环境有如下两种方式。

1. 从 DOS 环境进入

在 DOS 命令行中输入:

C>CD \TC✓ (指定当前目录为 TC 子目录)

C>TC✓ (进入 Turbo C 环境)

这时进入 Turbo C 环境的主菜单窗口, 屏幕显示如图 1.2 所示。

2. 从 Windows 环境进入

在 Windows 95 以上版本环境中, 如果本机中已安装了 Turbo C, 可以在桌面上建立一个快捷方式, 双击该快捷图标即可进入 C 语言环境。或者选择“开始”/“运行”命令, 在“运行”对话框中输入“C:\TC\TC”, 单击“确定”按钮即可。屏幕显示如图 1.2 所示。

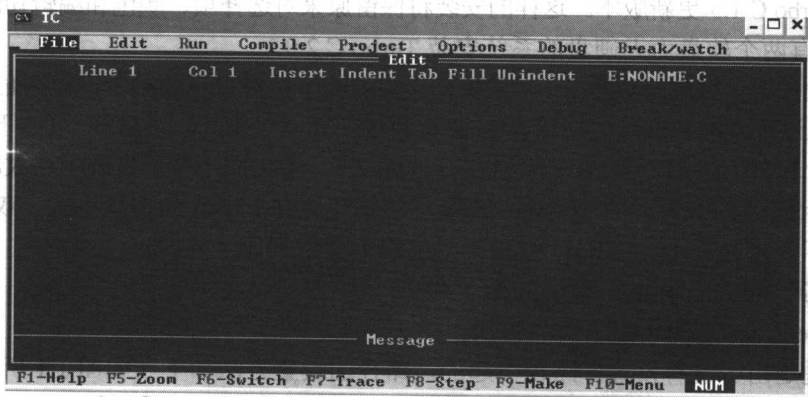


图 1.2 “TC”窗口

需要说明的是: 以上两种方式有一个共同的前提, 即 Turbo C 的安装路径为 C: \TC, 如果用户计算机中 Turbo C 的安装路径不同的话, 在上述方式中改变相应路径即可。

1.3 “TC”窗口

下面介绍“TC”窗口的组成(如图 1.3 所示)。

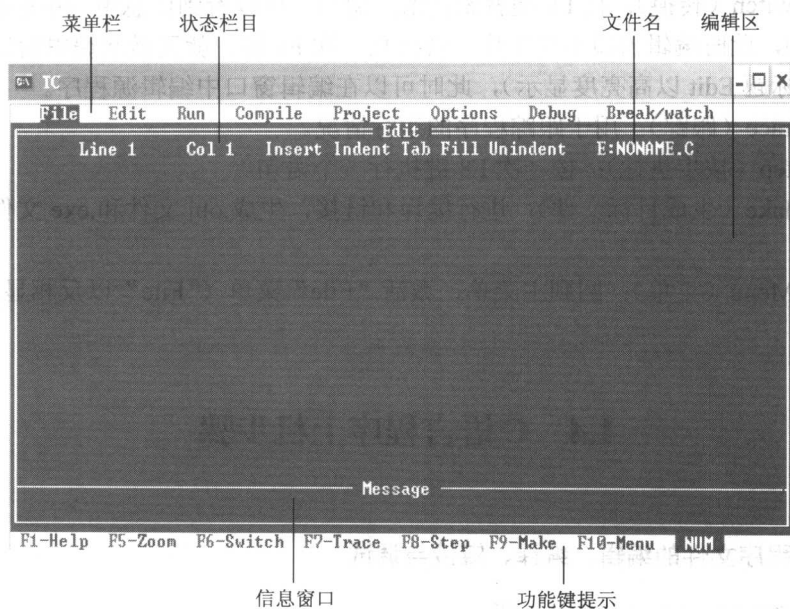


图 1.3 TC 窗口

1. 菜单栏

它包括 8 个主菜单：File（文件）、Edit（编辑）、Run（运行）、Compile（编译）、Project（项目）、Options（选项）、Debug（调试）和 Break/watch（断点/监视），每一主菜单都有其子菜单，分别实现各项操作。

2. 状态栏

(1) Line 1 和 Col 1 表示当前光标的位置在第 1 行和第 1 列，当光标移动时，Line 和 Col 后面的数字也随之改变，它的作用是告诉用户光标当前所在的位置。

(2) Insert 表示现在处于插入状态。可用 Insert 键进行切换。

(3) E: NONAME.C 是系统对新文件自动的命名，如果将文件保存为另一名字，将不再显示 E: NONAME.C，而是显示新命名的名字。

3. 编辑区

编辑区即编辑源程序窗口。

4. 窗口

窗口用于显示编译和连接时的有关信息。在信息窗口上方有“Message”字样作为标志。

5. 功能键提示

一些功能键的作用如下。

(1) F1+Help（帮助）：按 F1 键会显示帮助信息。

(2) F5+Zoom（分区控制）：如果当前在编辑窗口工作，即编辑窗口是激活状态，按 F5 键就不显示信息窗口，它的作用是扩大编辑窗口，以便能容纳和显示较长的源程序。若再按一次 F5 键，就会恢复信息窗口。如果当前信息窗口是激活的，按 F5 键就不显示编辑窗口，它的作用是扩大信息窗口，显示较多的信息，便于工作观看。若再按一次 F5 键，就会恢复原状。

(3) F6+Switch (转换): 按 F6 键就激活信息窗口 (可以看到信息窗口中的标题 Message 以高亮度显示), 此时编辑窗口不能工作。若再按一次 F6 键, 就又激活编辑窗口 (可以看到信息窗口中的标题 Edit 以高亮度显示), 此时可以在编辑窗口中编辑源程序。

(4) F7+Trace (跟踪): 用于跟踪程序的运行情况。

(5) F8+Step (按步执行): 按一次 F8 键执行一个语句。

(6) F9+Make (生成目标文件): 进行编译和链接, 生成.obj 文件和.exe 文件, 但不进行运行。

(7) F10+Menu (菜单): 回到主菜单, 激活“File”菜单 (“File”以反相显示, 如图 1.3 所示)。

1.4 C 语言程序上机步骤

1.4.1 单程序文件的编辑、编译、链接与调试

(1) 进入“TC”窗口 (如图 1.3 所示)。

(2) 选择“File”/“New”命令, 按回车键, 打开一新文件 (如需打开已存在的文件, 应选择“File”/“Open”命令, 根据系统提示打开文件), 如图 1.4 所示。

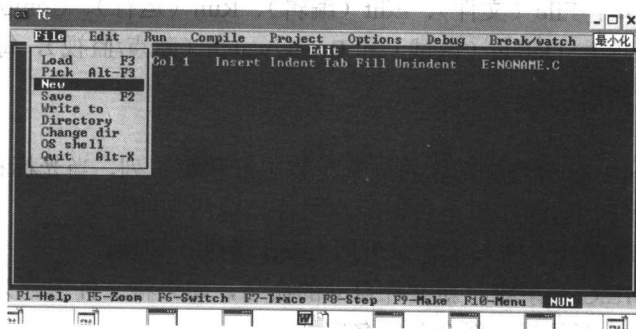


图 1.4 新建文件

(3) 输入源程序, 如图 1.5 所示。

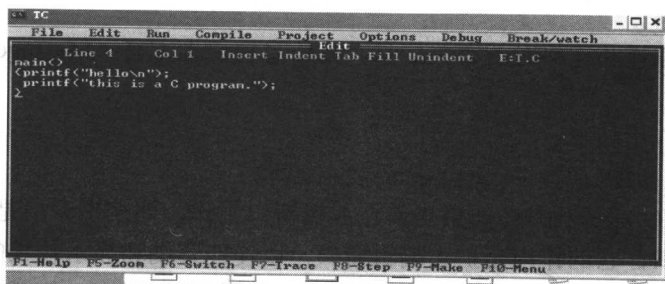


图 1.5 输入源程序

如在编辑窗口中输入如下源程序。

```
main( )
{ printf("hello\n");
  printf("this is a C program.");
}
```

(4) 保存源程序。选择“File”/“Save”命令，输入文件名，如图 1.6 所示。此处以“prgram.c”为名保存在 E:\下。

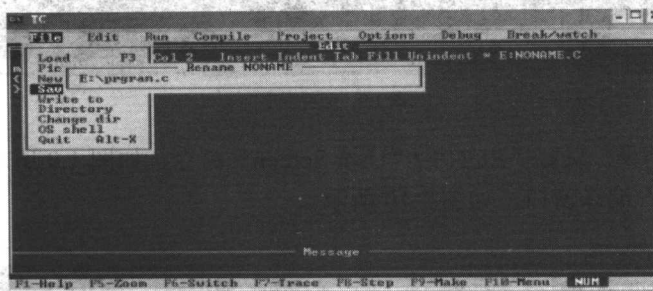


图 1.6 保存源程序

(5) 编译源程序。选择“Compile”/“Compile to obj”命令，编译生成该程序的目标文件。如果编译没有错误，则出现 Success（成功）提示，然后按任意键继续，如图 1.7 所示。

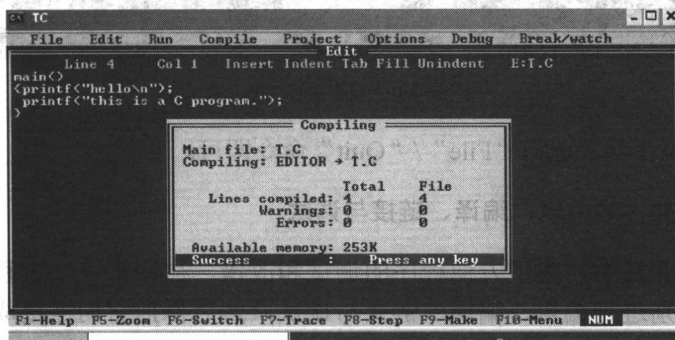


图 1.7 编译源程序

(6) 链接生成可执行文件。有了目标文件还不能直接运行，还要将目标文件与系统提供的库函数和包含文件等连接成一个可执行文件（后缀为.EXE），才能运行。选择“Compile”/“Link exe file”命令，如没有错误，则出现成功提示，如图 1.8 所示。

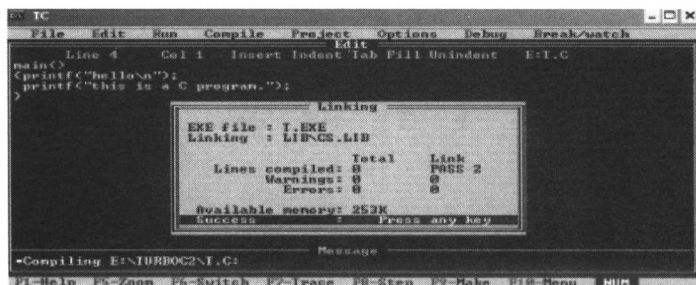


图 1.8 连接

(7) 运行程序。选择“Run”/“Run”命令，如图 1.9 所示。

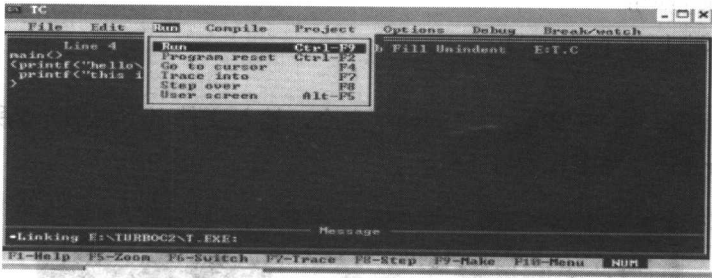


图 1.9 运行程序

(8) 查看运行结果。选择“RUN”/“User Screen”命令，如图 1.9 所示。检查完结果后，按任意键返回“TC”编译窗口，如图 1.10 所示。

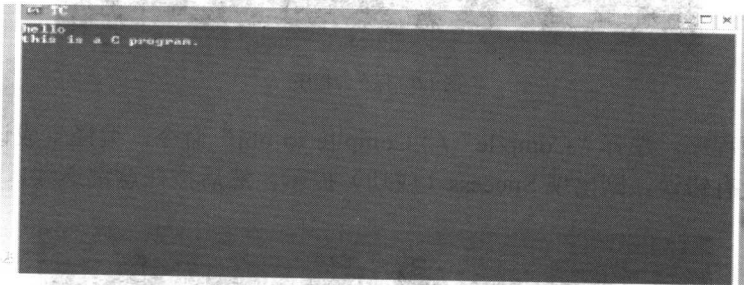


图 1.10 查看运行结果

(9) 退出 TC 编译器。选择“File”/“Quit”命令即可。

1.4.2 大程序文件的编辑、编译、链接与调试

在实用软件的开发中，为了方便程序的调试和改错，往往将一个大程序文件拆分成多个小程序文件，分别进行编辑、编译、链接与调试，然后再把它们连接成一个完整的可执行程序。

例如，设将一个 C 语言程序拆分成 3 个程序 file1.c，file2.c 和 file3.c，这 3 个程序分别如下。

```
file1.c
#include "stdio.h"
main ()
{ printf("I'm a teacher!\n");
  abc2();
  abc3();
}
```

file2.c

```

abc2()
{
    printf("*****\n");
}

file3.c
abc3() /*这是一个打印程序*/
{
    printf("Hello!Welcome to our school!\n");
    printf("_____");
}

```

若要将这 3 个文件编译装配成一个名为 abc.exe 的执行文件，可按下列步骤进行。

(1) 先分别编辑 3 个程序文件，并分别命名为 file1.c, file2.c 和 file3.c。

(2) 建立一个 abc.prj 的项目文件，在编辑窗口输入以下内容。

```

file1.c
file2.c
file3.c

```

以文件名 abc.prj 保存。

(3) 按 F10 键，在“TC”窗口中选择“Project”/“Project name”命令，在“Project name”对话框中输入 abc.prj。以后进行编译时将自动对项目文件中规定的 3 个源文件分别进行编译，然后连接成 abc.exe 文件。

(4) 把 file1.c, file2.c 和 file3.c 中的任意一个文件装入编辑窗口，进行编译、链接，最后生成 abc.exe 文件。

(5) 按 Ctrl+F9 组合键或退到 DOS 状态，运行 abc.exe，屏幕上显示如下运行结果。

```

I'm a teacher!
*****
Hello!Welcome to our school!
_____

```

注意：

(1) 如果其中有些文件已经编译成.OBJ 文件，而又没有修改过，可直接写上.obj 扩展名。此时将不再编译而只进行链接。

例如：

```

file1.obj
file2.c
file3.c

```

将不对 file1.c 进行编译，而直接链接 file1.obj。

(2) 当项目文件中的每个文件无扩展名时，均按源文件（即后缀为.c）对待。

(3) 项目文件中的文件也可以是库文件，但必须写上扩展名.lib。

所有文件在编译链接过程中如果发现错误时，只要修改出现错误的文件，其他文件不必

修改, 编译系统在重新编译时, 会根据文件最后的修改日期, 只对修改的文件重新编译, 这样大大缩短了改错和编译的时间。

1.5 Turbo C 各菜单功能

1.5.1 主菜单

1. “File” 菜单

按 Alt+F 组合键可进入 “File” 菜单, 该菜单包括以下命令 (如图 1.11 所示)。

(1) Load (加载): 装入一个文件, 可用类似 DOS 的通配符 (如 *.C) 来进行列表选择; 也可装入其他扩展名的文件, 只要给出文件名 (或只给路径) 即可。该项的热键为 F3, 即只要在主菜单中按 F3 键即可进入该项, 而不需要先进入 “File” 菜单再选择此项。

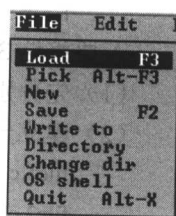


图 1.11 “File” 菜单

(2) Pick (选择): 将最近装入编辑窗口的 8 个文件列成一个表让用户选择, 选择后将该程序装入编辑区, 并将光标置在上次修改过的地方。其热键为 Alt+F3。

(3) New (新文件): 说明文件是新的, 缺省文件名为 NONAME.C, 存盘时可改名。

(4) Save (存盘): 将编辑区中的文件存盘, 若文件名是 NONAME.C 时, 将询问是否更改文件名。其热键为 F2。

(5) Write to (存盘): 可由用户给出文件名将编辑区中的文件存盘, 若该文件已存在, 则询问要不要覆盖。

(6) Directory (目录): 显示目录及目录中的文件, 并可由用户选择。

(7) Change dir (改变目录): 显示当前目录, 用户可以改变显示的目录。

(8) Os shell (暂时退出): 暂时退出 Turbo C 2.0 到 DOS 提示符下, 此时可以运行 DOS 命令, 若想回到 Turbo C 2.0 中, 只要在 DOS 状态下输入 EXIT 即可。

(9) Quit (退出): 退出 Turbo C 2.0, 返回到 DOS 环境中。其热键为 Alt+X。

说明: 以上各项可用光标键移动色棒进行选择, 按回车键则执行。也可用每一项的第一个大写字母直接选择。若要退到主菜单或从它的下一级菜单列表框退回均可按 Esc 键, Turbo C 2.0 所有菜单均采用这种方法进行操作, 以下不再说明。

2. “Edit” 菜单

按 Alt+E 组合键可进入 “Edit” 菜单, 若再按回车键, 则光标出现在编辑窗口, 此时用户可以进行文本编辑。

“Edit” 菜单中各命令功能如下。

PageUp: 向前翻页。

PageDn: 向后翻页。

Home: 将光标移到所在行的开始。

End: 将光标移到所在行的结尾。

Ctrl+Y: 删除光标所在的一行。

Ctrl+T: 删除光标所在处的一个词。

Ctrl+KB: 设置块开始。

Ctrl+KK: 设置块结尾。

Ctrl+KV: 块移动。

Ctrl+KC: 块复制。

Ctrl+KY: 块删除。

Ctrl+KR: 读文件。

Ctrl+KW: 存文件。

Ctrl+KP: 块文件打印。

Ctrl+F1: 如果光标所在处为 Turbo C 2.0 库函数, 则获得有关该函数。

Ctrl+Q[: 查找 Turbo C 2.0 双界符的后匹配符。

Ctrl+Q]: 查找 Turbo C 2.0 双界符的前匹配符。

说明:

(1) Turbo C 2.0 的双界符包括以下几种符号:

花括号 { }

尖括号 < >

圆括号 ()

方括号 []

注释符 /* */

双引号 ""

单引号 ''

(2) Turbo C 2.0 在编辑文件时还有一种功能, 就是能够自动缩进, 即光标定位和上一个非空字符对齐。在编辑窗口中, Ctrl+OL 组合键为自动缩进开关的控制键。

3. “Run” 菜单

按 Alt+R 组合键可进入 “Run” 菜单, 该菜单包括以下命令 (如图 1.12 所示)。

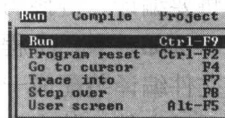


图 1.12 “Run” 菜单

(1) Run (运行程序): 运行由 “Project” / “Project name” 命令指定的文件名或当前编辑区的文件。如果对上次编译后的源代码未做过修改, 则直接运行到下一个断点 (没有断点则运行到结束); 否则先进行编译、连接后才运行。其热键为 Ctrl+F9。

(2) Program reset (程序重启): 中止当前的调试, 释放分给程序的空间。其热键为 Ctrl+F2。

(3) Go to cursor (运行到光标处): 调试程序时使用, 选择该命令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, 否则提示错误。其热键为 F4。

(4) Trace into (跟踪进入): 在执行一条调用其他用户定义的子函数时, 若选择该命令, 则执行长条将跟踪到该子函数内部去执行。其热键为 F7。

(5) Step over (单步执行): 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。其热键为 F8。

(6) User screen (用户屏幕): 显示程序运行时在屏幕上显示的结果。其热键为 Alt+F5。

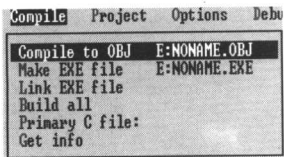


图 1.13 “Compile” 菜单

4. “Compile” 菜单

按 Alt+C 组合键可进入 Compile 菜单, 该菜单包括以下命令(如图 1.13 所示)。

- (1) Compile to OBJ (编译生成目标码): 将一个 C 源文件编译生成 OBJ 目标文件, 同时显示生成的文件名。其热键为 Alt-F9。
- (2) Make EXE file (生成执行文件): 此命令生成一个 .EXE 文件, 并显示生成的 EXE 文件名。其中 EXE 文件名是下面几项之一。
 - ① 由 “Project” / “Project name” 命令说明的项目文件名。
 - ② 若没有项目文件名, 则是由 “Primary C file” 命令说明的源文件。
 - ③ 若以上两项都没有文件名, 则为当前窗口的文件名。
- (3) Link EXE file (链接生成执行文件): 把当前 OBJ 文件及库文件链接在一起生成 EXE 文件。
- (4) Build all (建立所有文件): 重新编译项目里的所有文件, 并进行装配生成 EXE 文件。该命令不做过时检查(上面的几个命令要做过时检查, 即如果目前项目里源文件的日期和时间与目标文件相同或更早, 则拒绝对源文件进行编译)。
- (5) Primary C file (主 C 文件): 当在该项中指定了主文件后, 在以后的编译中, 如没有项目文件名则编译此项中规定的主 C 文件, 如果编译中有错误, 则将此文件调入编辑窗口, 不管目前窗口中是不是主 C 文件。
- (6) Get info: 获得有关当前路径、源文件名、源文件字节大小、编译中的错误数目、可用空间等信息。

5. “Project” 菜单

按 Alt+P 组合键可进入 “Project” 菜单, 该菜单包括以下命令(如图 1.14 所示)。

- (1) Project name (项目名): 项目名具有 .prj 的扩展名, 其中包括将要编译、链接的文件名。

例如, 有一个程序由 file1.c, file2.c 和 file3.c 组成, 要将这 3 个文件编译装配成一个 file.exe 的执行文件, 可以先建立一个 file.prj 的项目文件, 其内容如下。

file1.c file2.c file3.c

此时将 file.prj 放入 “Project name” 项中, 以后进行编译时将自动对项目文件中规定的 3 个源文件分别进行编译, 然后连接成 file.exe 文件。如果其中有些文件已经编译成 .obj 文件, 而又没有修改过, 可直接写上 .obj 扩展名。此时将不再编译而只进行连接。例如, file1.obj file2.c file3.c 将不对 file1.c 进行编译, 而直接连接。

说明: 当项目文件中的每个文件无扩展名时, 均按源文件对待, 另外, 其中的文件也可以是库文件, 但必须写上扩展名 .lib。

- (2) Break make on (中止编译): 由用户选择是否在有 Warning (警告)、Errors (错误)、Fatal Errors (致命错误) 时或 Link (连接) 之前退出 Make 编译。

- (3) Auto dependencies (自动依赖): 当开关置为 on, 编译时将检查源文件与对应的 OBJ 文件日期和时间, 否则不进行检查。

- (4) Clear project (清除项目文件): 清除 “Project” / “Project name” 命令指定的项目文

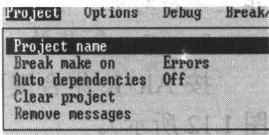


图 1.14 “Project” 菜单

件名。

(5) Remove messages (删除信息): 把编译或连接时产生的信息从信息窗口中清除掉。

6. “Options” 菜单

按 Alt+O 组合键可进入“Options”菜单。该菜单包括以下命令(如图 1.15 所示, 该菜单对初学者来说要谨慎使用)。

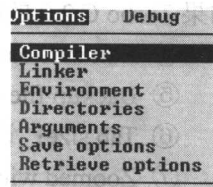


图 1.15 “Options” 菜单

(1) Compiler (编译器): 本菜单又有许多子菜单, 可以让用户选择硬件配置、存储模式、查错技术、代码优化、诊断信息控制和宏定义, 内容如下。

① Model: 共有 Tiny, small, medium, compact, large 和 huge 6 种不同模式可由同户选择。

② Define: 打开一个宏定义框, 同户可输入宏定义。多重定义可同分号, 赋值可用等号。

③ Code generation: 又有许多任选项, 这些任选项告诉编译器产生什么样的目标代码。

④ Optimization: 控制代码生成策略, 弹出一个子菜单, 供选择。

⑤ Source: 控制编码器如何处理源代码, 弹出一个子菜单, 提供以下选项。

- Identifier length: 说明标识符有效字符的个数, 默认为 32 个。

- Nested comments: 是否允许嵌套注释。

- ANSI keywords only: 是只允许 ANSI 关键字还是也允许 Turbo C 2.0 关键字。

On: 允许使用 ANSI 关键字, 把 Turbo C 2.0 的关键字看做是用户定义的标识符。

Off: 允许使用 Turbo C 2.0 的关键字。

⑥ Errors: 控制编译器处理和响应诊断信息, 弹出一个子菜单, 供选择。

⑦ Names: 改变代码、数据和 BSS 段的缺省段、组和类名, 一般不需改变。

说明: 该菜单内容很多, 具体细节请查阅有关手册。

(2) Linker (链接器): 本菜单设置有关连接的选择项, 它有以下内容。

① Map file menu 选择是否产生 MAP 映射文件。

② Initialize segments: 是否在连接时初始化没有初始化的段。

③ Devault libraries: 是否在连接其他编译程序产生的目标文件时去寻找其缺省库。

④ Graphics library: 是否连接 graphics 库中的函数。

⑤ Warn duplicate symbols: 检查 OBJ 文件和 LIB 文件中的重复符号。

⑥ Stack warinig: 是否让连接程序产生 No stack 的警告信息。

⑦ Case-sensitive link: 是否区分大小写字母。

(3) Environment (环境): 本菜单规定是否对某些文件自动存盘及制表键和屏幕大小的设置, 内容如下。

① Message tracking: 当滚动信息窗口中的错误信息时, 控制 Turbo C 是否跟踪编辑程序中的语法错误及跟踪方式。

Current file: 跟踪在编辑窗口中的文件错误。

All files: 跟踪所有文件错误。

Off: 不跟踪。

② **Keep message:** 编译前是否清除“Message”窗口中的信息。

③ **Config auto save:** 选 on 时, 在选择“Run”, “Shell”命令或退出集成开发环境之前, 如果 Turbo C 2.0 的配置被改过, 则所做的改动将存入配置文件中; 选 off 时不存。

④ **Edit auto save:** 是否在选择“Run”或“Shell”命令之前, 自动存储编辑的源文件。

⑤ **Backup file:** 是否在源文件存盘时产生备份文件 (BAK 文件)。

⑥ **Tab size:** 设置制表键大小, 默认为 8。可取值为 2~16 (Tab mode on) 时有效。

⑦ **Zoomed windows:** 将现行活动窗口放大到整个屏幕, 其热键为 F5。

⑧ **Screen size:** 弹出一个子菜单, 允许选择显示屏幕的行数, 标准为 25, EGA 为 43, VGA 为 50。

(4) **Directories (路径):** 规定编译、链接所需文件的路径, 内容如下。

① **Include directories:** 包含文件的路径, 多个子目录用“;”分开。

② **Library directories:** 库文件路径, 多个子目录用“;”分开。

③ **Output directoried:** 输出文件 (OBJ, EXE, MAP 文件) 的目录。

④ **Turbo C directoried:** Turbo C 所在的目录。

⑤ **Pick file name:** 定义加载的 pick 文件名, 如不定义则从 current pick file 中取。

⑥ **Arguments (命令行参数):** 允许用户使用命令行参数。

⑦ **Save options (存储配置):** 保存所有选择的编译、连接、调试和项目到配置文件中, 缺省的配置文件为 TCCONFIG.TC。

⑧ **Retrive options:** 装入一个配置文件到 Turbo C 中, Turbo C 将使用该文件的选择项。

7. “Debug”菜单

按 Alt+D 组合键可进入“Debug”菜单, 该菜单主要用于查错, 它包括以下命令。

(1) **Evaluate Expression:** 要计算结果的表达式。

① **Result:** 显示表达式的计算结果。

② **value:** 赋给新值。

(2) **Call stack:** 该项不可接触; 而在 Turbo C debugger 时用于检查堆栈情况。

(3) **Find function:** 在运行 Turbo C debugger 时用于显示规定的函数。

(4) **Refresh display:** 如果编辑窗口偶然被用户窗口重写了, 可用此命令恢复编辑窗口的内容。

(5) **Display swapping:** 控制编辑窗口与程序输出窗口的转换关系。选择此命令将弹出一个选择窗口, 其中有 3 种选择。

① **Smart (默认方式):** 执行代码产生输出时切换到程序输出窗口, 然后又返回到编辑窗口。

② **None:** 不进行屏幕切换。

③ **Always:** 每执行一条语句切换一次屏幕。

(6) **Source debugging:** 控制编译器是否在可执行文件中加入调试信息。选择此命令将弹出一个选择窗口, 其中有 3 种选择。

① **On:** 在可执行文件中加入调试信息, 为源代码级调试做准备。

② **Standalone:** 在可执行文件中加入调试信息, 但只能使用独立调试工具 Turbo

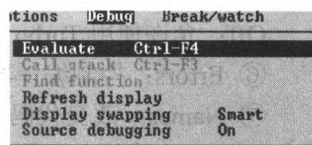


图 1.16 “Debug”菜单