

21世纪应用型本科院校规划教材

# C语言实验指导与习题解析

CYUYANSHIYANZHIDAOYUXITIJIEXI

主 编 陈舜青  
蔡晓丽  
饶 琛



南京大学出版社

21世纪应用型本科院校规划教材

# C语言实验指导与习题解析

主 编 陈舜青  
蔡晓丽  
饶 琛

 南京大学出版社

## 图书在版编目(CIP)数据

C语言实验指导与习题解析/陈舜青编著. —南京:  
南京大学出版社, 2006. 9

21世纪应用型本科院校规划教材

ISBN 7-305-04855-0

I. C... II. 陈... 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112149号

出版者 南京大学出版社  
社 址 南京市汉口路22号 邮 编 210093  
网 址 <http://press.nju.edu.cn>  
出版人 左 健  
丛 书 名 21世纪应用型本科院校规划教材  
书 名 C语言实验指导与习题解析  
主 编 陈舜青 蔡晓丽 饶 琛  
责任编辑 何永国 编辑热线 025-83686531  
照 排 南京南琳图文制作有限公司  
印 刷 丹阳兴华印刷厂  
开 本 787×1092 1/16 印张 11.25 字数 280千  
版 次 2006年9月第1版 2006年9月第1次印刷  
印 数 1~3000  
ISBN 7-305-04855-0/TP·317  
定 价 18.00元  
发行热线 025-83592169 025-83592317  
电子邮件 [sales@press.nju.edu.cn](mailto:sales@press.nju.edu.cn)(销售部)  
[nupress1@public1.ptt.js.cn](mailto:nupress1@public1.ptt.js.cn)

---

\* 版权所有, 侵权必究

\* 凡购买南大版图书, 如有印装质量问题, 请与所购  
图书销售部门联系调换

# 前 言

计算机 C 语言是一种基础性程序设计语言,它不仅具有表达能力强、代码质量高和可移植性好等特点,并且能极好地支持结构化、模块化等软件开发方法,更兼备高级语言和低级语言的许多优点,既可以编写应用程序,又可以编写系统程序,是一种应用广泛的计算机语言。

很多高校都开设 C 语言课程,普遍认为 C 语言是学习计算机语言的最好一门入门语种,能很好地锻炼一个人的编程能力。除了计算机专业的学生开设 C 语言课,很多非计算机专业也开设 C 语言课程,该课程对学生将专业知识与计算机编程相结合具有较大帮助作用。

本书通过习题练习和上机实验,能够帮助读者很好地掌握 C 语言的基础知识和基本概念,逐步建立用计算机处理问题的逻辑思维方式并掌握程序设计的基本方法,具有基本的编程能力。对计算机专业的学生具有较好的引导帮助作用;对非计算机专业的学生参加计算机等级考试具有较好的辅导作用。本书按照先易后难的规律,内容由浅入深,每单元习题包含单项选择题、填空题、编程题三个部分。由于“文件”部分在实际应用中使用广泛,在教材中内容安排较后,期末结束时往往学得有点匆忙,因此在实验 15 文件的使用上机前就先编写了简单的文件操作上机实验,并且在习题部分也作了加强,以便学生把这部分内容学得更好,弥补课堂教学的不足。

进行上机实验时,可以在“运行结果及体会”框中记录多次运行程序的输入量、输出结果和体会。习题中有些是为了帮助读者掌握相关的基本概念和基础知识,有些是为了帮助读者阅读程序、培养编程能力,有些题目有多种解法。建议读者在学习 C 语言时,尽量自己完成习题,独立思考、上机验证,并用不同的输入数据运行。做题过程中提倡开阔思路,求得更多的解法,这样才能真正提高编程能力。本书的全部程序均在 Turbo C 环境中测试通过。需要电子稿的读者可以与出版社联系。

本书 Turbo C 2.0 集成开发环境、实验 1、实验 15~16、习题 4、习题 6、第 4 单元和附录由陈舜青编写;习题 1~2 和习题 5 由蔡晓丽编写;实验 6~9 和习题 3 由饶琛编写;实验 10~14 由刘模群编写;实验 2~5 由蒋小莺编写;全文由陈舜青负责统稿。

感谢读者选择阅读本书。由于时间仓促,作者水平有限,书中难免有疏漏和不当之处,欢迎同行和读者对本书内容提出修改建议,我们将不胜感激。

作 者

2006 年 8 月

# 目 录

|                                        |           |
|----------------------------------------|-----------|
| <b>第 1 单元 Turbo C 2.0 集成开发环境 .....</b> | <b>1</b>  |
| 一、进入 Turbo C 的工作窗口 .....               | 1         |
| 二、Turbo C 的工作窗口介绍 .....                | 1         |
| 三、设置 TC 工作环境 .....                     | 2         |
| 四、改变用户工作目录 .....                       | 3         |
| 五、TC 常用菜单项的使用 .....                    | 3         |
| 六、对单文件的编辑、编译、连接和运行 .....               | 8         |
| 七、对多文件程序进行编译和连接 .....                  | 10        |
| 八、TC 程序的调试 .....                       | 10        |
| 九、常用功能键表 .....                         | 11        |
| <br><b>第 2 单元 C 语言上机实验 .....</b>       | <b>13</b> |
| 实验 1 C 程序的运行方法 .....                   | 13        |
| 实验 2 数据类型及表达式 .....                    | 17        |
| 实验 3 基本语句(一) .....                     | 20        |
| 实验 4 基本语句(二) .....                     | 23        |
| 实验 5 基本语句(三) .....                     | 27        |
| 实验 6 数组(一) .....                       | 33        |
| 实验 7 数组(二) .....                       | 37        |
| 实验 8 函数(一) .....                       | 40        |
| 实验 9 函数(二) .....                       | 47        |
| 实验 10 指针(一) .....                      | 54        |
| 实验 11 指针(二) .....                      | 58        |
| 实验 12 指针(三) .....                      | 63        |
| 实验 13 结构体和共用体(一) .....                 | 66        |
| 实验 14 结构体和共用体(二) .....                 | 70        |
| 实验 15 文件的使用 .....                      | 74        |

## 目 录

---

|                                  |            |
|----------------------------------|------------|
| 实验 16 综合编程 .....                 | 81         |
| <br>                             |            |
| <b>第 3 单元 习题与习题解析 .....</b>      | <b>90</b>  |
| 习题 1 基本语句 .....                  | 90         |
| 习题 1 基本语句解析 .....                | 94         |
| 习题 2 数组 .....                    | 99         |
| 习题 2 数组解析 .....                  | 104        |
| 习题 3 函数 .....                    | 108        |
| 习题 3 函数解析 .....                  | 115        |
| 习题 4 指针 .....                    | 122        |
| 习题 4 指针解析 .....                  | 126        |
| 习题 5 结构体和共用体 .....               | 133        |
| 习题 5 结构体和共用体解析 .....             | 140        |
| 习题 6 文件 .....                    | 147        |
| 习题 6 文件解析 .....                  | 153        |
| <br>                             |            |
| <b>第 4 单元 C 语言理论考试模拟试卷 .....</b> | <b>159</b> |
| <b>附录 Turbo C 编译常见出错信息 .....</b> | <b>171</b> |

# 第 1 单元 Turbo C 2.0 集成开发环境

## 一、进入 Turbo C 的工作窗口

Turbo C 提供了两种编译环境：集成编译开发环境 TC 和命令行编译 TCC。TC 是一个集编辑、编译、连接、调试及运行为一体的集成模块，为用户提供了一个方便的集成开发环境。厂家提供的编译系统都是以压缩文件的形式存放在软盘或光盘上的，安装时只要在硬盘建立 TC 文件夹，将存放在软盘上的压缩文件 TTC.EXE 复制到硬盘 TC 文件夹中，双击该文件解压，或者用 Turbo C(简称 TC)系统提供的 install.exe 安装程序进行安装，用户可以根据系统提示进行操作，指定存放系统文件的目录和存储模式，也可以按默认方案安装。

在 TC 主目录下已经存放了 tc.exe、tcc.exe 两个执行文件，其中 tc.exe 是将编辑、编译、连接、调试和运行集成为一体的基本模块；tcc.exe 则提供了某些补充功能，例如可以在程序中嵌入汇编代码等，一般只需用到 tc.exe。

进入 Turbo C 环境需要调用 tc.exe，可以在 DOS 平台进入 TC 目录，键入 TC 执行 tc.exe。或者在 Windows 平台通过“资源管理器”、“我的电脑”找到 TC 文件夹中的 tc.exe，双击该文件即可进入 Turbo C 环境。为更方便地进入 TC 环境，可以在桌面上建立相应的快捷方式。

## 二、Turbo C 的工作窗口介绍

进入 Turbo C 的工作窗口后，可以使用 Alt+Enter 进行全屏和窗口工作两种方式的切换，Turbo C 的界面如图 1 所示。屏幕上最上面一行是主菜单，有 8 个下拉菜单选项；主菜单下面为编辑框，程序文件的编辑工作就在该框中进行，在编辑框的最上面一行显示编辑状态，如当前光标所在行号(Line)、列号(Col)、当前正在编辑的文件的文件名和路径等；编辑框下面有一部分是信息框(Message)，表示程序编译和连接时产生的信息；Turbo C 工作窗口的最下面一行作为常用热键，用户可以使用热键快速完成所需要的操作。

主菜单的操作有两种方法：

- 1) 按 F10 键激活主菜单，按光标移动键“↓”或“↑”把光标移到希望的菜单项处，然后按回车键。
- 2) 直接按 Alt+菜单项第一个字母(例如按 Alt+F 就是选择了“File”项，进入文件菜单)。

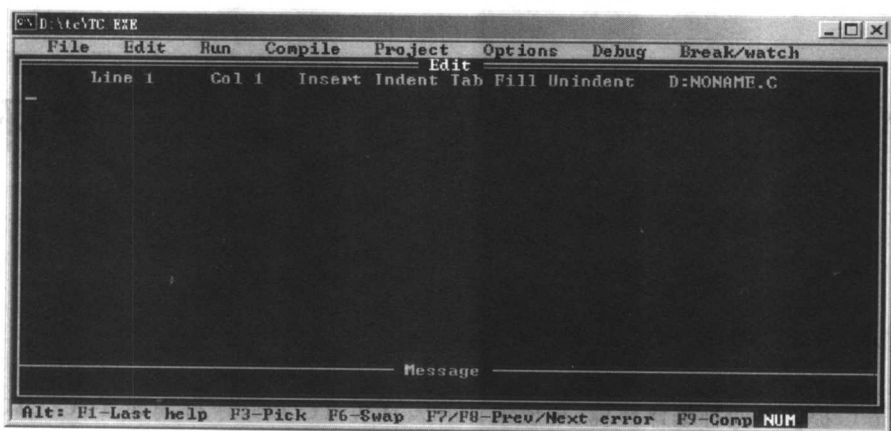


图 1 TC 环境界面

### 三、设置 TC 工作环境

除 Edit 菜单项仅为进入编辑状态外,其他菜单项被选择后会打开一个子菜单。通过如下操作可以配置文件:

选择“Options”菜单项,屏幕上弹出一个子菜单窗口,如图 2 所示。为了保存安装程序建立的环境,在子菜单中选择“Save Options”命令,被选中后,弹出一个对话框,要求输入配置文件名,为了方便,一般取缺省文件名 TCCONFIG.TC。

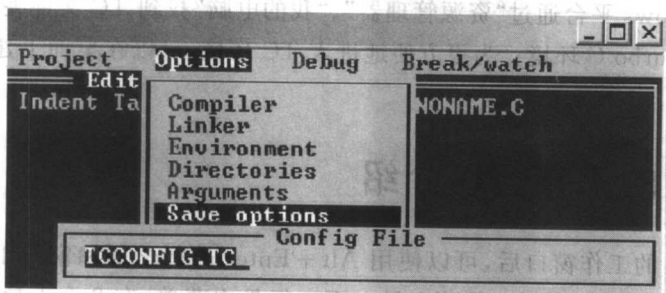


图 2 配置文件界面

如果想重建工作环境,可以改变“Options”子菜单中的命令。例如:可以选择“Directories”命令,并按下图 3 所示设置存放文件的目录(这里假定 TC 文件夹位置是安装在 C 盘,用户文件夹为 D:\ZXF)。设置完毕后按 Esc 键返回“Options”子菜单,然后用上面的方法(执行 Save Options 命令)保存设置好的环境。

|                                    |                               |
|------------------------------------|-------------------------------|
| Include directories: C:\TC\INCLUDE | Install 建立的头文件所在的缺省目录         |
| Library directories: C:\TC\LIB     | Install 建立的库文件所在的缺省目录         |
| Output directories: D:\ZXF         | 自己建立的存放 *.EXE 和 *.OBJ 输出文件的目录 |
| Turbo C directories: C:\TC         | Install 建立的 TC 所在的缺省目录        |
| Pick file name:                    | 弹出一个窗口,允许用户规定 pick 文件名        |
| Current Pick file:                 | 显示当前 pick 文件名和它所在的目录,不能修改     |

图 3 “Options”子菜单设置界面

## 四、改变用户工作目录

安装程序把 Turbo C 系统的可执行文件存放在 C:\TC 目录中,如果直接在这个目录中工作,源文件、\*.OBJ 及 \*.EXE 等文件都将被存储在该目录中,这样,系统文件和用户自己的文件混杂在一起,导致了管理不便。可以专设存放源文件和存放用户自己的可执行文件的目录,比如,用目录 D:\ZXF 存放源文件、用户生成的 \*.OBJ 和 \*.EXE 文件(ZXF 是某用户在 D 盘上自己建立的文件夹)。

为了能将用户生成的 \*.OBJ 和 \*.EXE 文件自动存放在 D:\ZXF 中,在图 3 中要明确指出输出文件的目录(Output directories:D:\ZXF)。

为了将源文件 \*.C 存放在 D:\ZXF 目录中,一般是把 D:\ZXF 目录作为工作目录。具体方法是:在“File”下拉菜单中选择“Change dir”项,出现新目录输入框,提示用户输入所选择的工作目录名(如:将原有系统默认工作目录删除,输入新的用户工作目录 D:\ZXF),如图 4 所示。此后,存盘和加载文件时,只要键入文件名即可,无需再输入盘符路径。

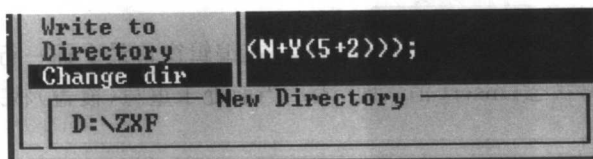


图 4 改变用户文件夹界面

## 五、TC 常用菜单项的使用

### 1. File 菜单

1) Load(加载):在编辑器中装入一个已有的文件,可以用通配符(\*)进行列表选择,也可以直接键入文件名及其路径。例如:想装入的文件是 E 盘的 GB 文件夹中的 P3.C 文件,则输入文件名如图 5 所示。按回车键后,该文件就被调入内存并显示。如果记不清所要装入的源文件名,想看一下当前目录中有哪些源文件,则可以在 Load 命令后出现 \*.C 时直接按回车键,TC 就会显示当前目录下所有后缀为 .C 的文件名,用光标移动键移到要装入的文件名处按回车键,就可以显示该文件内容。要调出 A 盘上的源程序,只要键入:A:\\*.C 即可。

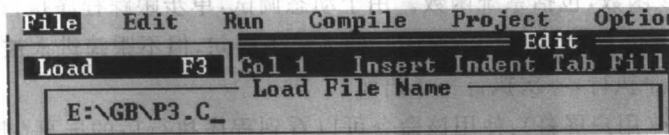


图 5 加载已有文件的界面

2) Pick(选择):将最近装入编辑器的八个文件组成一个表,让用户选择后装入编辑器,可以方便快速地打开最近操作过的文件。

3) New(新文件):说明当前要编辑的文件是最新文件,装入编辑器后的文件缺省名为 NONAME.C。

4) Save(存盘):将编辑器中的文件存盘。如果该文件已存盘,则按原有文件名存盘;如果该文件未存过盘,则出现输入文件名对话框,非当前盘非当前目录必须输入盘符路径。TC 中的文件名,不管输入的是大写还是小写,都会自动统一成大写。

5) Write to(另存为):将编辑器中的文件以另一个文件名存盘,如果该文件已存在,则覆盖该文件。

6) Directory(目录):按回车键显示 \*.\* ,再按回车键显示当前目录及其所需文件列表,可以选择要打开的文件。如果显示 \*.\* 时改为 \*.C,则只显示当前目录及其 C 源程序文件列表。

7) Change Dir(改变驱动器):显示当前目录,改变当前驱动器及其目录。

8) OS shell(暂时退出):暂时退出 Turbo C,转到 DOS 提示符下,可以用 EXIT 命令返回。

9) Quit(退出):退出 Turbo C。

## 2. Edit 菜单

执行 Edit 命令可以对源程序文件进行编辑。在编辑窗口最上面一行出现 Insert 提示时,表示当前处于插入编辑状态;Insert 消失时,表示当前处于重写(覆盖)状态。按 Insert 键可以改变“插入/重写”状态。

Ctrl+F1 键的作用是显示在线帮助。用法是:把光标移到任一关键字或函数名处按 Ctrl+F1 键,会在屏幕上显示出关键字或函数的使用信息。

## 3. Run 菜单

使用该菜单命令可以运行和调试程序。下面介绍各菜单命令:

1) Run(运行):编译、连接生成目标文件和可执行文件并运行程序。如果选择 Debug/Source debug 为 on,程序执行时每一个断点处都会暂停下来,再次按 Ctrl+F9 键,继续执行到下一个断点处或程序结束。

2) Program reset(程序重启):该命令终止当前调试,释放分配给程序的空间,关闭已打开的文件。但不改变断点设置。

3) Go to cursor(执行到光标所在行):用于动态调试,使程序执行到编辑窗口中光标所在行。注意光标所在行应该是可执行代码语句。

4) Trace into(跟踪进入被调用函数):执行当前函数中的下一条语句,如果该语句为函数调用语句,则进入该函数,包括系统函数。用于动态调试,单步跟踪程序执行。

5) Step over(单步执行):执行当前函数的下一条语句,但不跟踪进入被调入的子函数,用该命令单步跟踪程序执行,一次执行一条语句。

6) User screen(用户屏幕):使用该命令可以看到程序执行后的屏幕(把编辑窗口切换到程序输出窗口)。

## 4. Compile 菜单

使用该菜单命令可以编译和连接程序,生成可执行文件。各子菜单命令功能如下:

1) Compile to OBJ(把源文件编译成目标文件):把 C 语言源文件(\*.C)编译成目标文件(\*.OBJ),在编译过程中,弹出一窗口显示编译信息,如有编译错误,则显示在信息框中,并用高亮度条在编辑中显示出错语句的位置。

2) Make EXE file(生成可执行文件):此命令调用相应工程文件生成一个可执行文件,并显示生成的可执行文件的文件名。

如果处理的是.C文件,编译并连接生成相应的.OBJ和.EXE文件。如果源文件的生成时间比.OBJ文件早,则不再对源文件进行编译,只有更新了源文件后才重新编译。

如果处理的是.OBJ文件,只进行连接生成相应的.EXE文件,同样只有更新了.OBJ文件后才重新连接。

对文件的处理顺序是:① Project name 中的.PRJ;② Primary C file 中的.C文件;③ 最后装入编辑窗口的.C文件。

3) Link EXE file(连接生成可执行文件):将当前目标文件和相应库文件连接成一个新的可执行文件,但不执行。该命令不检查日期和时间。

4) Build all(编译连接相关文件):重新编译连接相应工程文件中的所有文件,不管其是否是过期文件,无条件重新编译与连接生成相应的可执行文件。

5) Primary C File(主 C 文件):该命令用于编译含有多个嵌入文件的 C 语言源程序,执行时弹出一个对话框,要求输入将要编译或 Make 的新文件名。当编译出现错误时,自动装入源文件以便用于修改错误。

6) Get info(获得编译信息):该命令可以显示编译信息,如当前文件名、源文件长度,可用内存空间等。

## 5. Project 菜单

该命令可以用来把多个源文件和目标文件连接成一个完整的程序,子程序功能如下:

1) Project name:弹出对话框,要求输入将要编译、连接的.PRJ文件名。用这种方法生成的.EXE文件的主文件名就是.PRJ文件的主文件名。

2) Break make on:规定终止 Make 的缺省条件,共有四个选项。

◆ Warnings:编译完一个文件后如发现“警告”以上的错误停止编译。

◆ Errors:编译完一个文件后如发现“错误”,停止编译。

◆ Fatal errors:编译完全部文件后,如发现错误,停止编译。

◆ Link:在连接前 Make,即只生成.OBJ文件。

3) Auto dependencies:设置自动依赖关系。

◆ On:自动检查.C文件与相应.OBJ文件的日期时间关系,TC在编译时把日期时间信息存放在.OBJ文件中,若.C文件比.OBJ文件新,则重新编译。

◆ Off:不进行日期时间检查。

4) Clear project:清除 Project name 和重置消息窗口。

5) Remove messages:清除消息窗口中的出错信息。

## 6. Options 菜单

该命令可以设置集成开发环境的各项设置,包括库和包含文件的目录、程序运行参数等。

常用设置如下:

1) Compiler(编译程序选项):其下子菜单提供如下功能:

- ◆ Model 子菜单:设置存储器模式,允许用户选择 Tiny、Small、Medium、Compact、Large、Huge 等 6 种模式中的一种。
- ◆ Defines 子菜单:输入宏定义。
- ◆ Code generation 子菜单:控制生成目标代码的形式,如 Instruction set 可以选择 8088、8086 或 80286 指令。
- ◆ Optimization:控制代码生成策略,弹出子菜单供选择。
- ◆ Source:控制编译器如何处理源代码,弹出一个子菜单供选择。如指定标识符有效字符个数(Identifier length 可选择 1~32)。
- ◆ Errors 子菜单:代码优化和诊断信息,设置停止编译的错误个数及警告个数,弹出子菜单供选择。
- ◆ Names:改变代码、数据和 BSS 段的缺省段、组和类名,一般不需改变。

2) Linker(连接子菜单):改变连接子菜单中的命令,可以改变连接器的设置。子菜单如下:

- ◆ Map file:控制是否产生映射文件。
- ◆ Initialize segments:控制连接器是否对未初始化的段进行初始化。
- ◆ Default libraries:当由其他编译器产生的目标文件时,那些编译器可能在目标文件中放入了一个缺省库。该命令控制连接器是否在这些库中寻找所需函数。
- ◆ Graphics library:打开或关闭自动查找图形库的开关,当选择了 Off 时可在 .PRJ 文件中写上 graphics.lib,也可输出图形。
- ◆ Warn duplicate symbols:控制是否检查 .OBJ 文件和 .LIB 文件中的重复符号。
- ◆ Stack warning:控制是否 No stack 警告信息。
- ◆ Case-sensitive link:控制是否区分大小写字母不通用。

3) Environment(环境子菜单):设置文件是否自动存盘、制表键的空格数和显示屏幕行数等,为编辑程序设置环境。子菜单功能如下:

- ◆ Message tracking:当滚动信息窗口中的错误信息时,控制 TC 是否跟踪编辑程序中的语法错误及跟踪方式。
- ◆ Keep messages:控制在编译、Make 之前是否保存信息窗口中的错误信息。
- ◆ Config auto save:控制是否保存选项设置。
- ◆ Edit auto save:控制是否将正在编辑的文件自动存盘。
- ◆ Backup files:控制是否产生备份文件。
- ◆ Tab size:选择制表(Tab)键的空格数。默认值为 8,可取值为 2~16(Tab mode on 时有效)。
- ◆ Zoomed windows:与热键 F5 的作用一样,把当前激活的窗口(编辑窗或信息窗)放大到整屏。
- ◆ 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(目录子菜单):选择头文件、库文件、编译器等所在的目录及可执行文件的

输出目录。子菜单功能如前面图 3 所示。

5) Arguments(参数设置命令):允许用户在集成环境下运行程序时使用参数,执行这一命令后,在弹出的窗口中输入运行所需参数,如图 6 所示,然后 Run/Run 执行程序。



图 6 集成环境运行程序输入运行所需参数的界面

6) Save Options(环境保存命令):如前面图 2 所示。该命令的作用是将设置的命令保存到磁盘文件中,缺省的文件是配置文件 tcconfig. tc。

7) Retrieve Options(恢复环境命令):装入以前用 Save options 命令保存的配置文件,为 TC 设置环境。

## 7. Debug 菜单

该菜单命令可以设置除断点和表达式之外的集成调试环境和各项功能。常用设置如下:

1) Evaluate(计算):用于计算机和显示变量或表达式的值,还允许用户修改此值。运行该命令后,弹出一个窗口,该窗口包含计算域(Evaluate)、结果域(Result)和新值域(New value)三部分。用户可以在其中计算任何合法的 C 语言表达式或变量,但不包括函数调用语句和宏定义的常量。如果调试程序可以计算表达式的值,它就可以在结果域中显示出来;如果表达式引用了一个变量或简单的数据单元,用户可以将光标移动到新值域并把表达式作为新值输入。在调试过程中可以采用给某些变量赋新值的办法寻找错误。

2) Call strack(调用栈):显示程序正在运行的函数调用、序列。

3) Find function:显示编辑窗口中某一函数的定义。弹出一个窗口,在其中输入要显示的函数名并按回车,光标就指向函数的定义处。

4) Refresh display(恢复原先屏幕):如果编辑屏幕被重写,该选择项可恢复当前屏幕内容。

5) Display swapping(屏幕切换):控制编辑窗口与程序输入窗口的转换,弹出一个窗口提供三种选择:

- ◆ None:不进行屏幕切换。
- ◆ Smart(缺省方式):执行代码产生输出时切换到程序输入窗口,然后返回到编辑窗口。
- ◆ Always:每执行一条语句切换一次屏幕。

6) Source debugging(源程序调试):控制编译器是否在可执行文件中加入调试信息,弹出选择窗口提供三种选择:

- ◆ On:在可执行文件中加入调试信息,为源代码调试做准备。
- ◆ 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 debugger。
- ◆ None:在可执行文件中不加入调试信息。

## 8. Break/watch 菜单

使用该菜单命令可以进行集成调试环境中断点和监视表达式的设置,子菜单功能如下:

1) Add watch(加入一个监视表达式):监视表达式是其运算值可在监视窗口中显示的表达式,每当程序暂停时它的值会被重新计算。用户可以通过监视表达式的值来判断程序执行的情况。当用户选择本命令时,调试程序打开一个弹出窗口,提示用户键入一个监视表达式,按回车键可以增加一个表达式及其当前值到观察窗口。

2) Delete watch(删除监视表达式):该命令用来从观察窗口中删除监视表达式,使用该命令时,监视表达式必须是可见的。也可以把光标移到 Watch 窗口中某一表达式用 Del 键或 Ctrl+Y 键删除。

3) Edit watch(编辑监视表达式):该命令允许用户编辑观察窗口中的当前监视表达式。

4) Remove all watches(删除所有监视表达式):该命令用来从观察窗口中删除所有监视表达式。

5) Toggle breakpoint(打开或关闭断点):断点是程序执行中暂停的地方,设置断点可以让用户有足够的时间检查关键变量和表达式的值。否则,用户无法确定程序是否按设计的那样执行。该命令用来设置和删除(再次执行时光标所在处的断点,程序运行过程中遇到断点就暂停。

6) Clear all breakpoints(清除所有断点):该命令用来从程序中清除所有断点。

7) View next breakpoint(显示下一个断点):该命令将光标移动到程序的下一个断点,它不执行程序,只是在编辑窗口中定位活动的断点。

## 六、对单文件的编辑、编译、连接和运行

### 1. 编辑

要输入和编辑一个新的 C 程序,应先按 F10 激活主菜单,用光标移动键选定 File/New 命令,按回车键。出现空白的编辑窗口,用户就可以输入和编辑源程序了。输入一个完整程序或部分程序后,应及时将源程序保存起来,按 F2 功能键或执行 File/Save 命令。第一次存盘将弹出一个对话框,要求用户输入文件名,以后再次存盘不再弹出对话框。文件将保存在用户工作目录内,如果 Turbo C 安装在 D 盘 TC 文件夹中,且用户未更改过工作目录时,默认的工作目录即为 D:\TC,默认文件名为 NONAME.C(一般应改成自己命名的文件名)。建议较长的程序在录入时多存几次。不指定路径时,文件就存在 D:\TC 中,如果用户不想保存在 D:\TC,也可以自己设定盘符路径,如:D:\ZXF\P126\_2.C,或在存盘前先用 File/Change dir 命令改变用户的工作目录。

在用指定的文件名 P126\_2.C 存盘后,编辑窗口中右上角的文件名就自动改为 P126\_2.C。如果以后还想改名,可以使用 File/Write to 命令。

编辑一个已存在的文件,可以使用 File/Load 命令,调出已存在的文件,重新修改、处理。为了验证 TC 是否能正常工作,TC 自带一个测试程序 Hello.C,用菜单命令 File/Load 可将该文件从 TC 文件夹中调出,观察其是否能正常编译、连接和运行。

## 2. 编译

执行 Compile/Compile to OBJ 命令,此时系统显示默认的目标文件名。如果当前被编译的源程序文件名为 P126\_2.C,则系统自动将目标文件名定为 P126\_2.OBJ,按回车键执行该命令即可编译程序。

## 3. 连接

有了目标文件后还不能直接运行,必须把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),才能运行这个 .EXE 文件。

执行 Compile/Link EXE file 命令就可进行连接,在当前工作目录或配置文件 TCCONFIG.TC 指定的输出目录中生成一个可执行文件 P126\_2.EXE。

用 Compile/Make EXE file 命令或者直接按 F9 键可以一次完成编译和连接,生成目标文件和可执行文件。

## 4. 运行

如果在编译、连接过程中未出现错误,系统就会生成目标文件和可执行文件,这时就可以运行了,运行程序可以采用以下几种方法:

### 1) 在 DOS 环境下运行

如果当前目录就是输出文件 .EXE 所在目录,直接输入可执行文件名,即可运行程序。假设可执行文件名为 P126\_2.EXE,输入命令 P126\_2,按回车就能运行该程序。如果该文件不在当前目录中,则应指出其路径,如输入:D:\ZXF\P126\_2 命令。

### 2) 在 TC 环境中通过菜单选择运行

按 Ctrl+F9 键,或选择执行 Run/Run 命令,即可运行程序。实际上 Ctrl+F9 的作用不仅是运行,而且是包括编译、连接和运行。因此,在编辑窗口完成一个源程序的编辑后,如果经过检查认为没有问题,就可直接按 Ctrl+F9 键,Turbo C 将一次完成从编译、连接到运行的全过程。如果在编译或连接中有错,就停下,显示出错信息,用户改正后再按 Ctrl+F9 键重新编译、连接,若没有错误就自动开始运行。

程序运行时,如有键盘输入语句(scanf 语句),屏幕显示会切换到用户屏,此时用户可以从键盘输入数据。

如果想看运行结果,可选择 Run/User screen 命令,或者直接按 Alt+F5 键,此时屏幕的显示切换到用户屏,程序运行的结果显示在用户屏上。用户看完结果后,按任一键就回到编辑窗口。

下面是最常用的功能键的有关操作:

- ◆ F9:进行编译和连接。
- ◆ Ctrl+F9:编译、连接、运行。
- ◆ Alt+F5:观看运行结果。

操作顺序一般为:输入程序→F9→Ctrl+F9→Alt+F5。

## 七、对多文件程序进行编译和连接

如果一个源程序包含多个文件(源程序. C),则应当对各文件分别进行编译,得到多个文件(目标文件. OBJ),然后将这些目标文件以及库函数、包含文件等连接成一个可执行文件。

TC 提供了对多文件程序进行编译和连接的简便方法。要将这些文件组成一个“项目”为此要建立一个“项目文件”,在该文件中包含各文件的名字,然后将该项目连接,就可以得到可执行文件. EXE。具体步骤如下:

在 TC 编辑环境中,输入各源文件的名字。如果有一个程序,包含两个文件,分别为 P1. C 和 P2. C(后缀名. C 可以省略不写,源文件名可以带有路径),则在 TC 编辑窗口中输入这两个文件名,如图 7 所示。



图 7 建立项目文件

执行 File/Write to 命令将文件保存,文件名为 P1\_P2. PRJ, P1\_P2 是用户自己指定的名字,后缀必须用. PRJ (Project 的缩写),以表示为项目。

执行 Project/Project name 命令,在对话框中输入需要进行编译和连接的项目文件名(系统显示为 \*. PRJ,要求用户修改为实际的文件名)。由于刚才是用 P1\_P2. PRJ 存盘的,因此现在应输入 P1\_P2. PRJ,按回车键确认。

选择 Compile/Make EXE file 命令,系统就会对此项目文件进行编译和连接,并生成两个目标文件 P1. OBJ 和 P2. OBJ,以及可执行文件 P1\_P2. EXE。

在选择“Make EXE file”或按 F9 键进行编译连接时,系统首先查找有无项目文件(. PRJ)。如果在 Project name 中指定了当前有效的的项目文件,则系统优先编译该项目中的文件,而不是编译编辑窗口中的文件。因此,在处理完一个多文件程序的编译和连接后,应及时将 Project name 项清空(选择 Project 下拉菜单中的 Clear Project 项),否则就会在编译连接时仍然把项目文件. PRJ 当做编译的对象,而不是编译编辑窗口中的源文件。

## 八、TC 程序的调试

### 1. TC 集成环境的调试程序

我们把发现、改正程序中错误的过程称为程序调试。有时仅仅靠观察不正确的程序是很难找到错误的关键所在的。因此大多数程序员都借助程序调试,再对被调试程序进行定位。调试程序能够控制程序的运行,可以在任一点上停止程序的执行,每次执行一句,观察程序输出的数据等。

要想使用集成调试程序,首先应该设置启用集成调试程序的开关,即选择 Debug/Source debugging 为打开状态。这时,用 RUN 选项或 Ctrl+F9 进行调试程序时,Turbo C 2.0 集成调试程序自动接管对程序的调试工作。

在程序较大时,一般是将一个源程序设置几个断点,当运行程序到断点处时,它就在执行断点所在行的第一个语句之前将程序挂起,同时把调试程序的控制权交给程序员。

在程序暂停时,你可以用许多方法来研究和处理程序。如:可以显示一个变量或表达式的值;在一个特定的窗口设置一些表达式并观察它们值的变化;修改变量的值;清除所有的断点或者设置断点;一次一行地单步执行程序;编辑文件、重新编译和连接程序等。调试程序的方法互不相同,要靠自己不断积累,使之日益丰富。一般使用 TC 的集成环境编译用户程序时,采用 RUN 选择项或 Ctrl+F9 命令,系统也能给出出错的地点供程序员修改调试之用,我们称这种方法为静态调试;而将使用调试程序进行的调试称其为动态跟踪调试。

## 2. 动态调试

### 1) 一般调试方法

为了使用集成调试工具,在编译和连接时要把 Options/Compiler/Code generation/OBJ debug information 选择开关和 Debug/Source debugging 选择开关都置为 On。这样,在生成的运行文件中才能包含与调试有关的符号信息。

为了调试方便,在编译和连接时应当把 Options/Compiler/Optimization/Jump optimization 选择开关置为 Off。因为当这个开关为 On 时,编译程序在优化代码时会引起执行代码与源代码行的不对应。

### 2) 步进执行法

为了找出逻辑错误,用 Run/Trace into(热键 F7)或 Run/Step over(热键 F8)可以一步一步地逐行执行程序,在怀疑可能出错的地方停下来使用 Break/watch/Add watch(热键 Ctrl+F7)命令或 Debug/Evaluate(热键 Ctrl+F4)命令,在弹出的监视窗口中输入监视表达式(一个变量或一个 C 表达式等)。通过查看、分析表达式的值,逐步找出出错原因。

### 3) 设置断点法

当调试一个大的程序时,为了加快调试速度,可以使用 Break/watch/Toggle breakpoint(热键 Ctrl+F8)命令在怀疑可能出错的代码行设置断点。设置了断点后,当按下 Ctrl+F9 键快速执行程序时,程序执行到断点处就会暂停下来等待输入命令。这时,可以用上面的方法查看表达式的值,也可以接着步进跟踪程序,再次按 Ctrl+F9 键,程序继续执行到下一个断点处或程序结束。

如果在调试的过程中想终止调试操作,可以按下 Ctrl+F2 键。要删除某一断点,可把光标移到断点行按 Ctrl+F8 键。

## 九、常用功能键表

表 1 Turbo C 常用编辑命令及功能键

| 命令     | 功能      | 命令 | 功能                                      |
|--------|---------|----|-----------------------------------------|
| Home   | 光标移到行首  | F1 | 打开帮助文件                                  |
| End    | 光标移到行尾  | F2 | 把当前编辑的文件存储到磁盘上,见 File/Save              |
| Ctrl+N | 插入一行    | F3 | 装载一个文件,见 File/Load                      |
| Ctrl+Y | 删除光标所在行 | F4 | 让程序从当前执行处起一直执行到光标所在行,见 Run/Go to cursor |