



工业和信息化普通高等教育“十二五”规划教材立项项目

21 世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

C语言程序设计 实验指导

The Practice of The C Programming Language

王洪海 陈向阳 盛魁 邵立 主编

- 注重理论联系实际
- 注重可读性与实用性
- 注重读者能力的培养



高校系列



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化普通高等教育“十二五”规划教材立项项目

21 世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

C语言程序设计 实验指导

The Practice of The C Programming Language

王洪海 陈向阳 盛魁 邵立 主编

杜秀全 薛峰 李凌峰 文萍芳 徐丽萍 余会娟 副主编

赵守忠 郑诚 主审



高校系列

人民邮电出版社

· 北 京 ·

图书在版编目 (CIP) 数据

C语言程序设计实验指导 / 王洪海等主编. — 北京:
人民邮电出版社, 2011.7
21世纪高等学校计算机规划教材
ISBN 978-7-115-25306-4

I. ①C… II. ①王… III. ①
C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(2011)第106965号

内 容 提 要

本书是《C语言程序设计》的配套教材,用于上机实训。全书共分4部分,第1部分为C语言的上机环境及其基本操作,第2部分为主要实验内容,第3部分为上机实验参考答案,第4部分为全国计算机等级考试二级C考试大纲及模拟试卷。

本书适合作为高等学校本科、高职高专、成人高校和其他初学者学习C程序设计的实训教材,也可供参加全国计算机等级考试(二级C)的读者参考。

工业和信息化部普通高等教育“十二五”规划教材立项项目

21世纪高等学校计算机规划教材

C语言程序设计实验指导

-
- ◆ 主 编 王洪海 陈向阳 盛 魁 邵 立
副 主 编 杜秀全 薛 峰 李凌峰 文萍芳 徐丽萍 余会娟
主 审 赵守忠 郑 诚
责任编辑 邹文波
- ◆ 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 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京昌平百善印刷厂印刷
- ◆ 开本: 787×1092 1/16
印张: 9.5 2011年7月第1版
字数: 242千字 2011年7月北京第1次印刷

ISBN 978-7-115-25306-4

定价: 23.00 元

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

反盗版热线: (010)67171154

前 言

C 语言是目前世界上最流行、使用最广泛的高级程序设计语言之一。在对操作系统、系统应用及需要对硬件进行操作的场合中,用 C 语言明显优于其他高级语言,因此,许多大型应用软件都是用 C 语言编写的。由于 C 语言功能强、使用灵活、可移植性好、目标程序质量好,从而受到编程者广泛的欢迎。

一般来讲,学习 C 语言程序设计时,很大程度是通过上机实训和大量练习来掌握其基本概念的,本书是《C 语言程序设计》的配套教材,用于上机实训。全书共分 4 部分,第 1 部分为 C 语言的上机环境及其基本操作,第 2 部分为主要实验内容,第 3 部分为上机实验参考答案,第 4 部分为全国计算机等级考试二级 C 考试大纲及模拟试卷。

本书由王洪海、陈向阳、盛魁、邵立任主编,杜秀全、薛峰、李凌峰、文萍芳、徐丽萍、余会娟任副主编,参与编写任务的有王洪海、陈向阳、盛魁、徐丽萍。全书由王洪海统稿并定稿。本书由中国科学技术大学教授、现安徽三联学院信息与通信技术系主任赵守忠教授和安徽大学计算机科学与技术学院副院长郑诚教授主审,安徽三联学院院长领导对于本套教材的出版也给予了大力的支持,在此一并表示衷心的感谢。

本书适合作为大学本科、高职高专、成人高校和其他初学者学习 C 程序设计的教材,也可供参加全国计算机等级考试(二级 C)的读者参考。

由于编者水平有限,书中难免存在错误和不妥之处,望广大读者提出宝贵意见,邮件请发至 sanlian_whh@163.com。

编者

2011 年 5 月

参编及合作单位

(排名不分先后)

安徽三联学院

中国科学技术大学

安徽大学

解放军电子工程学院

安庆医药高等专科学校

亳州职业技术学院

池州职业技术学院

安徽医学高等专科学校

目 录

第 1 部分 C 语言的上机环境及其基本操作

第 1 章 C 语言的上机环境	2
1.1 Turbo C 2.0 的安装和启动	2
1.2 Turbo C 2.0 集成开发环境的组成	2
第 2 章 C 语言上机的基本操作	11
2.1 实验基本步骤	11
2.2 实验报告基本内容	11
2.3 程序调试的方法	12
2.3.1 上机前要先熟悉程序运行的环境	12
2.3.2 程序设计过程中要为程序调试做好准备	12
2.3.3 调试程序的方法与技巧	13
2.4 C 语言调试运行操作中的常见错误	15

第 2 部分 主要实验内容

实验一 程序的运行环境操作和简单程序运行	28
实验二 数据类型、运算符及表达式	33
实验三 顺序结构程序设计	36
实验四 选择结构程序设计	39
实验五 循环结构程序设计	43
实验六 数组（一）	47
实验七 数组（二）	51
实验八 函数（一）	55
实验九 函数（二）	59

实验十 指针（一）	63
实验十一 指针（二）	68
实验十二 结构体与共用体	73
实验十三 文件	77

第 3 部分 上机实验参考答案

第 4 部分 全国计算机等级考试二级 C 考试大纲及模拟试卷

全国计算机等级考试二级 C 语言考试大纲	104
全国计算机等级考试二级 C 笔试模拟试卷 1	107
全国计算机等级考试二级 C 笔试模拟试卷 2	115
全国计算机等级考试二级 C 笔试模拟试卷 3	124
全国计算机等级考试二级 C 笔试模拟试卷 4	134
参考文献	143

第 1 部分

C 语言的上机环境及其 基本操作

第 1 章

C 语言的上机环境

1.1 Turbo C 2.0 的安装和启动

1. Turbo C 2.0 基本配置要求

Turbo C 2.0 可运行于 IBM-PC 系列微机, 包括 XT、AT 及 IBM 兼容机。此时操作系统要求为 DOS 2.0 或更高版本, 并至少需要 448KB 的 RAM, 可在任何彩色、单色 80 列监视器上运行。Turbo C2.0 支持数学协处理器芯片, 也可进行浮点仿真, 这将加快程序的执行。

2. Turbo C 2.0 的安装和启动

Turbo C 2.0 的安装非常简单, 只要运行安装程序 install 即可, 此时屏幕上显示以下 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。

这里选择按第 1 种方法进行安装, 在安装过程中按照提示顺序插入各个软盘, 就可以顺利地进行安装。安装完毕系统将在 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, 即 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。

1.2 Turbo C 2.0 集成开发环境的组成

在 Windows98/2000/XP 操作环境中, 运行 Turbo C 2.0 软件有两种方式: 在 Windows 中进入 DOS 环境, 运行 TC 软件; 在 Windows 环境中, 直接单击 TC。进入 Turbo C 2.0 集成开发环境后, 屏幕显示如图 1-1 所示。

其中, 主菜单、编辑区、信息窗口和参考区构成了 Turbo C 2.0 的主屏幕, 编程、编译、调试以及运行都将在这个主屏幕中进行。下面详细介绍主菜单的内容。

主菜单显示下列菜单项:

File Edit Run Compile Project Options Debug Break/watch

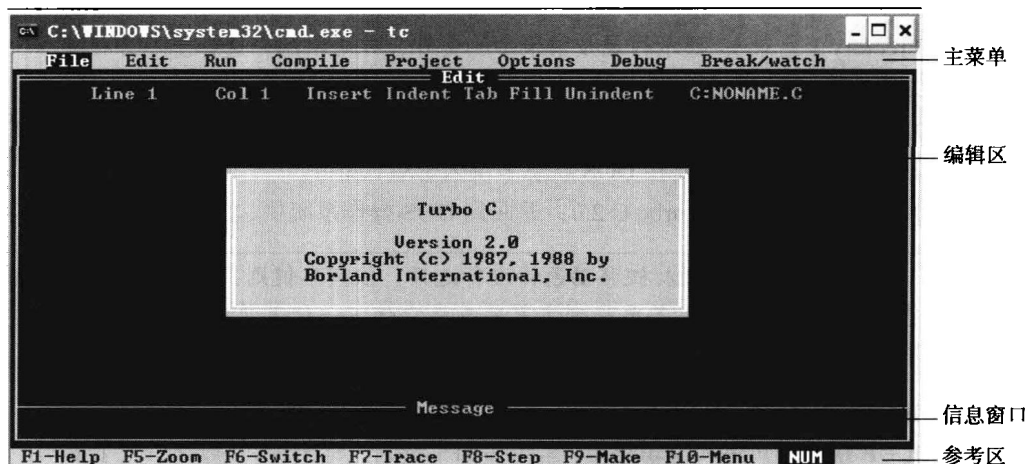


图 1-1 Turbo C 2.0 集成开发环境

除 Edit 外, 其他各项均有子菜单, 只要用 Alt 加上某菜单项中第 1 个字母 (即大写字母), 就可进入该项的子菜单中。

(1) File (文件) 菜单

按 Alt+F 快捷键可进入 File 菜单, 如图 1-2 所示。该菜单包括以下各项。

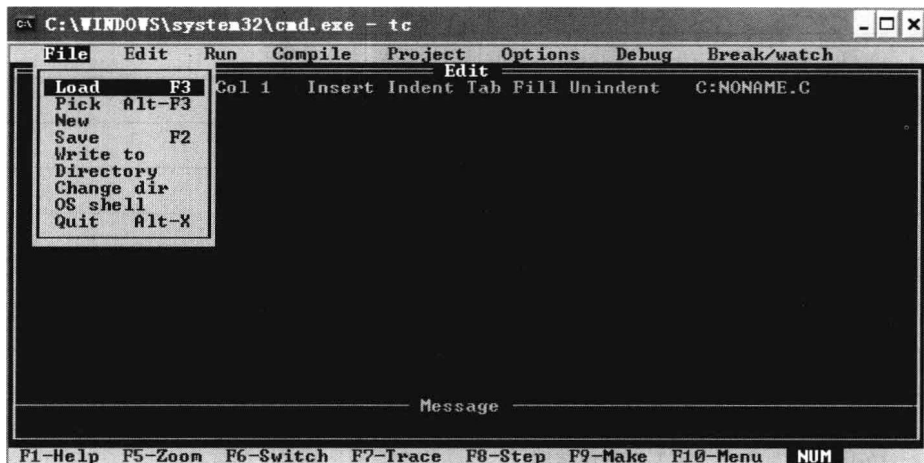


图 1-2 File 菜单

- 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 菜单再选此项。

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

- New (新文件): 建立一个新文件, 默认文件名为 NONAME.C, 存盘时可更改文件名。

- 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。

- 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。

- Directory（目录）：显示目录及目录中的文件，并可由用户选择。
- Change dir（改变目录）：显示当前目录，用户可以改变显示的目录。
- 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 状态下键入 EXIT 即可。
- Quit（退出）：退出 Turbo C 2.0，返回到 DOS 操作系统中，其热键为 Alt + X。



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 1 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（编辑）菜单

按 Alt + E 快捷键可进入编辑菜单，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。编辑方法基本与 Word 相同，按 F1 键获得有关编辑方法的帮助信息。与编辑有关的功能键如表 1-1 所示。

表 1-1 功能键及其说明

功 能 键	说 明
F1	获得 Turbo C 2.0 编辑命令的帮助信息
F5	扩大编辑窗口到整个屏幕
F6	在编辑窗口与信息窗口之间进行切换
F10	从编辑窗口转到主菜单
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	块拷贝
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，则获得有关该函数的帮助信息
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: {和}。
- 尖括号: <和>。
- 圆括号: (和)。
- 方括号: [和]。
- 注释符: /*和*/。
- 双引号: "。
- 单引号: '。

② Turbo C 2.0 具有文档自动缩进功能, 即光标定位和上一个非空字符对齐。在编辑窗口中, Ctrl+OL 为自动缩进开关的控制键。

(3) Run (运行) 菜单

按 Alt+R 快捷键可进入 Run 菜单, 如图 1-3 所示。该菜单包括以下各项。

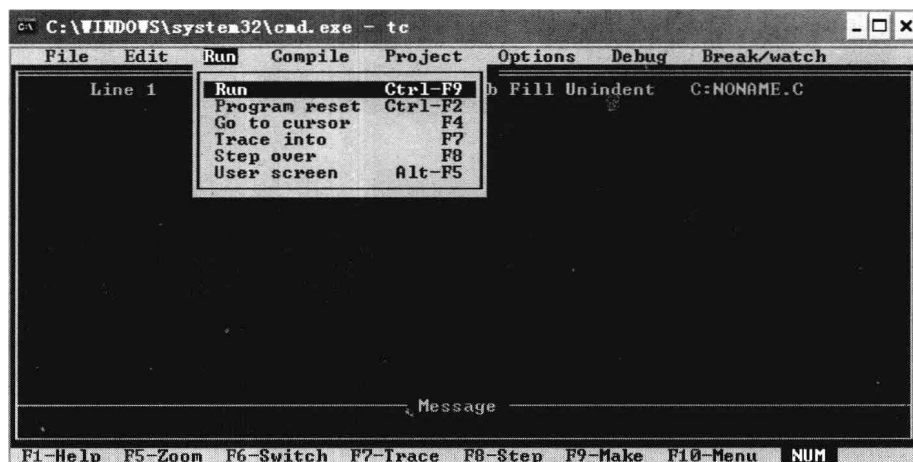


图 1-3 Run 菜单

● Run (运行程序): 运行由 Project/Project name 项指定的文件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。

● 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。

● Go to cursor (运行到光标处): 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, 否则提示错误。其热键为 F4。

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

● 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。

● 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。

(4) Compile (编译) 菜单

按 Alt+C 快捷键可进入 Compile 菜单, 如图 1-4 所示。该菜单包括以下各项。

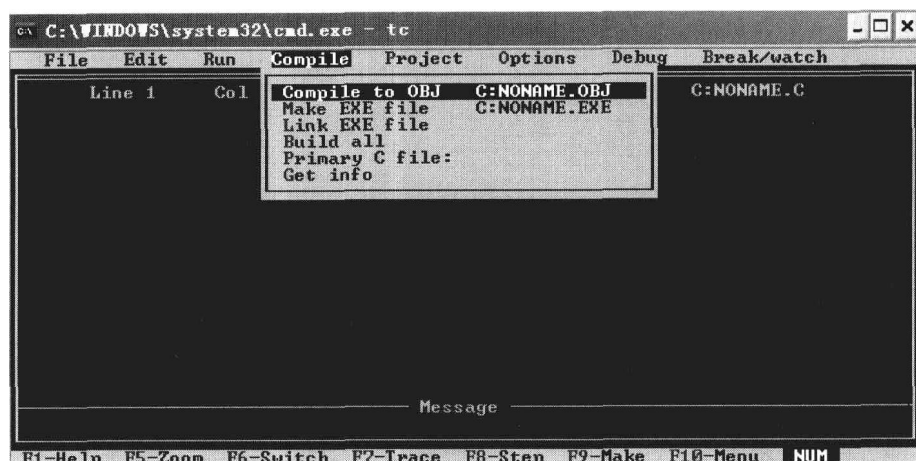


图 1-4 Compile 菜单

- **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。

- **Make EXE file** (生成执行文件): 此命令生成一个.EXE 的文件, 并显示生成的.EXE 文件名。

其中.EXE 文件名是下面几项之一:

由 Project/Project name 说明的项目文件名;

若没有项目文件名, 则由 Primary C file 说明的源文件;

若以上两项都没有文件名, 则为当前窗口的文件名。

- **Link EXE file** (连接生成执行文件): 把当前.OBJ 文件及库文件连接在一起生成.EXE 文件。

- **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。该命令不作过时检查 (上面的几条命令要作过时检查, 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)。

- **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。

- **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 (项目) 菜单

按 Alt+P 快捷键可进入 Project 菜单, 如图 1-5 所示。该菜单包括以下各项。

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

例如, 有一个程序由 file1.c, file2.c, 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
```

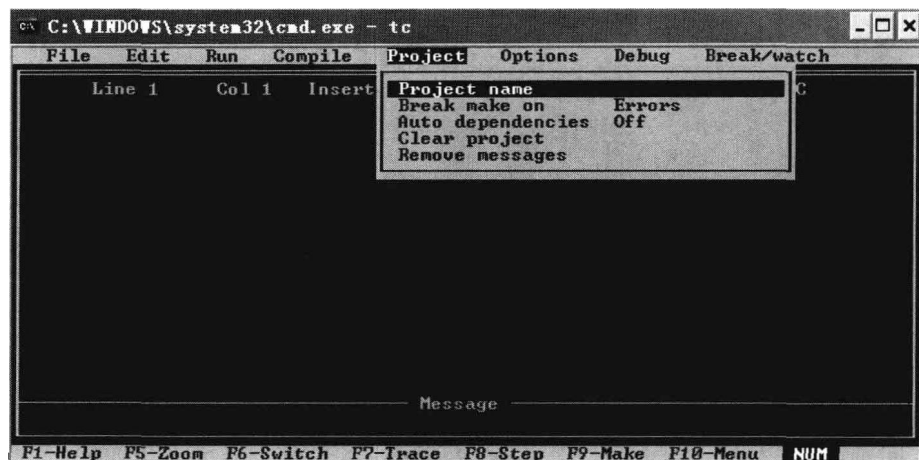


图 1-5 Project 菜单

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。



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

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

- 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 文件的日期和时间，否则不进行检查。

- Clear project (清除项目文件)：清除 Project/Project name 中的项目文件名。

- Remove messages (删除信息)：把错误信息从信息窗口中清除掉。

(6) Options (选择) 菜单

按 Alt+O 快捷键可进入 Options 菜单，如图 1-6 所示。该菜单对初学者来说要谨慎使用。

- Compiler (编译器)：本项选择又有许多子菜单，可以让用户选择硬件配置、存储模型、调试技术、代码优化、对话信息控制和宏定义。这些子菜单的功能如下。

Model：有 tiny, small, medium, compact, large, huge6 种不同模式可供用户选择。

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

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

Calling convention：可选择 C 或 Pascal 方式传递参数。

Instruction set：选择 8088/8086 或 80186/80286 指令系列。

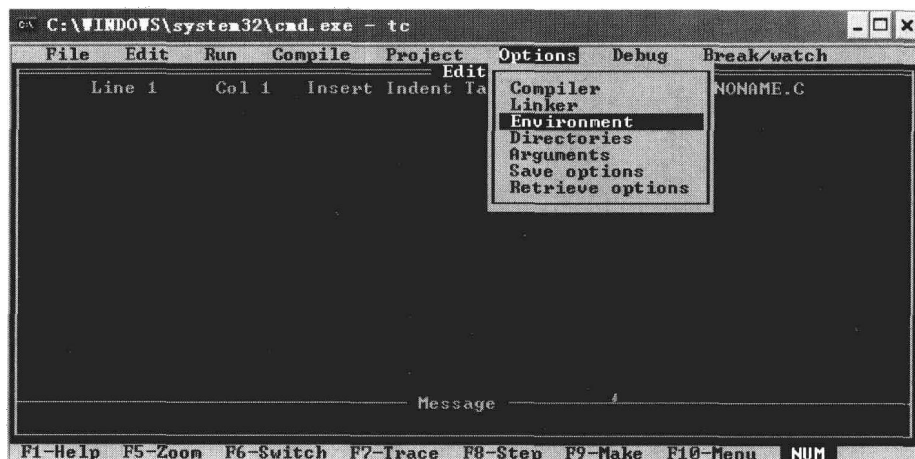


图 1-6 Options 菜单

Floating point: 可选择仿真浮点、数学协处理器浮点或无浮点运算。

Default char type: 规定 char 的类型。

Alignment: 规定地址对准原则。

Merge duplicate strings: 作优化用, 将重复的字符串合并在一起。

Standard stack frame: 产生一个标准的栈结构。

Test stack overflow: 产生一段程序运行时检测堆栈溢出的代码。

Line number: 在.OBJ 文件中放进行号以供调试时用。

OBJ debug information: 在.OBJ 文件中产生调试信息。

Optimization 包括以下内容。

Optimize for: 选择是对程序小型化还是对程序速度进行优化处理。

Use register variable: 用来选择是否允许使用寄存器变量。

Register optimization: 尽可能使用寄存器变量以减少过多的取数操作。

Jump optimization: 通过去除多余的跳转和调整循环与开关语句的办法压缩代码。

Source 包括以下内容。

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: 设置只允许 ANSI 关键字还是也允许 Turbo C 2.0 关键字。

Error 包括以下内容。

Error stop after: 多少个错误时停止编译, 默认为 25 个。

Warning stop after: 多少个警告错误时停止编译, 默认为 100 个。

Display warning 包括以下内容。

Portability warning: 移植性警告错误。

ANSI Violations: 侵犯了 ANSI 关键字的警告错误。

Common error: 常见的警告错误。

Less common error: 少见的警告错误。

Names: 用于改变段 (segment)、组 (group) 和类 (class) 的名字, 默认值为 CODE, DATA, BSS。

- **Linker (连接器)**: 本菜单设置有关连接的选择项, 包括以下内容。

Map file menu: 选择是否产生 .MAP 文件。

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

Devault libraries: 选择是否在连接其他编译程序产生的目标文件时去寻找其默认库。

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

Warn duplicate symbols: 当有重复符号时产生警告信息。

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

Case-sensitive link: 选择是否区分英文的大、小写。

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。

Message tracking 包括以下内容。

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。

Keep message: 编译前是否清除 Message 窗口中的信息。

Config auto save: 选 on 时, 在 Run、Shell 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 时不保存。

Edit auto save: 是否在 Run 或 Shell 之前, 自动存储编辑的源文件。

Backup file: 是否在源文件存盘时产生后备文件 (.BAK 文件)。

Tab size: 设置制表键大小, 默认为 8。

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

Screen size: 设置屏幕文本大小。

- **Directories (路径)**: 规定编译、连接所需文件的路径, 包括以下内容。

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

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

Output directoried: 输出文件 (.OBJ、EXE 和 .MAP 文件) 的目录。

Turbo C directoried: Turbo C 所在的目录。

Pick file name: 定义加载的 pick 文件名, 如不定义则从 current pick file 中选取。

- **Arguments (命令行参数)**: 允许用户使用命令行参数。

● **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, 默认的配置文件为 TCCONFIG.TC。

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

(7) Debug (调试) 菜单

按 Alt+D 快捷键可进入 Debug 菜单, 该菜单主要用于查错, 如图 1-7 所示。该菜单包括以下各项。

- **Evaluate**: 包括以下内容。

Expression: 要计算结果的表达式。

Result: 显示表达式的计算结果。

New value: 赋予新值。

- **Call stack**: 在 Turbo C debugger 时用于检查堆栈情况。

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



图 1-7 Debug 菜单

- Refresh display: 如果编辑窗口偶然被用户窗口重写了可用此恢复编辑窗口的内容。

(8) Break/watch (断点及监视表达式) 菜单

按 Alt+B 快捷键可进入 Break/watch 菜单, 如图 1-8 所示。该菜单包括以下各项。



图 1-8 Break/watch 菜单

- Add watch: 向监视窗口插入一监视表达式。
- Delete watch: 从监视窗口中删除当前的监视表达式。
- Edit watch: 在监视窗口中编辑一个监视表达式。
- Remove all watches: 从监视窗口中删除所有的监视表达式。
- Toggle breakpoint: 对光标所在的行设置或清除断点。
- Clear all breakpoints: 清除所有断点。
- View next breakpoint: 将光标移动到下一个断点处。