

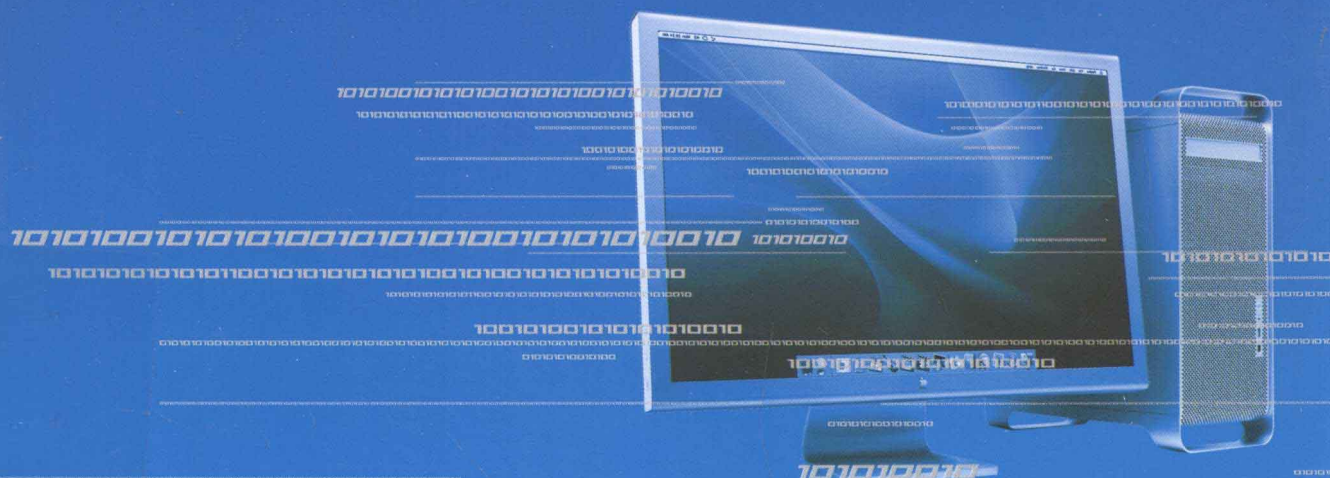


· 普通高等教育“十二五”规划教材

程序设计教程 习题解析与上机指导

刘前 孟宪伟 主编

C CHENGXU SHEJI
JIAOCHENG XITI JIEXI
YU SHANGJI ZHIDAO



国防工业出版社

National Defense Industry Press

普通高等教育“十二五”规划教材

C 程序设计教程 习题解析与上机指导

刘前 孟宪伟 主编

国防工业出版社

· 北京 ·

内 容 简 介

本书是《C 程序设计教程》的配套实验教材,主要是为教材各章的课后习题提供参考答案,并为学生上机实验提供指导。其目的是帮助学生进一步消化吸收 C 语言的基础知识和基本技能,提高运用 C 语言解决实际问题的能力,并为学生提供一个复习与检测的工具。

本书内容共分 4 章。第 1 章为 C 语言上机环境及操作指导,详细介绍了在 Turbo C 2.0 和 Visual C++ 6.0 集成环境下编辑、编译、调试和运行程序的方法,同时作者结合多年的教学经验,给出了常见的错误分析;第 2 章为 C 语言上机实训设计,一共包括 10 个实训内容,根据常见的程序算法,由浅入深地安排了 9 个基础的上机实验,在此基础上再进行综合的上机练习;第 3 章为综合模拟试题,提供了三套综合模拟试题;第 4 章为各章习题与参考答案,其中的所有程序都是在 Turbo C 环境下调试通过的。

本书内容丰富、概念清晰、实用性强,不仅可以作为《C 程序设计教程》的参考书,而且可以作为任何 C 语言教材的参考书;既适于高等学校师生或计算机培训班使用,也可供报考计算机等级考试者和其他自学者参考。

图书在版编目(CIP)数据

C 程序设计教程习题解析与上机指导/刘前,孟宪伟主编. —北京:
国防工业出版社,2013.1
普通高等教育“十二五”规划教材
ISBN 978-7-118-08343-9

I. ①C... II. ①刘...②孟... III. ①C 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(2012)第 244103 号

※

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

北京奥鑫印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 8 字数 178 千字

2013 年 1 月第 1 版第 1 次印刷 印数 1—4000 册 定价 20.00 元

(本书如有印装错误,我社负责调换)

国防书店: (010)88540777

发行邮购: (010)88540776

发行传真: (010)88540755

发行业务: (010)88540717

前 言

本书是《C 程序设计教程》一书的配套实验指导教材,旨在使学生更好地理解和掌握程序设计的思想、方法和技巧。本书内容从教学实际需要出发,兼顾不同学生的实际计算机水平,共设置了 4 章内容:C 语言上机环境及操作指导、C 语言上机实训设计、综合模拟试题和习题与参考答案。

第 1 章是 C 语言上机环境及操作指导,详细介绍了在 Turbo C 2.0 和 Visual C++ 6.0 集成环境下编辑、编译、调试和运行程序的方法;同时作者结合多年的教学经验,给出了常见的错误分析。

第 2 章是 C 语言上机实训设计,一共包括 10 个实训内容,上机训练循序渐进。便于学生由浅入深地学习,逐步熟悉编程环境、程序调试方法。根据常见的程序算法,由浅入深地安排了 9 个基础的上机实验,在此基础上再进行综合的上机练习。由于篇幅和课时的限制,在教材和讲授中不可能介绍很多例子,只能介绍一些典型的实训题目。建议读者除了完成教师指定的实训题目,尽可能阅读本书介绍的全部程序,并上机运行本书提供的全部实验内容以及自己感兴趣的程序,以开阔思路,提高编程能力。

第 3 章是综合模拟试题,提供了三套综合模拟试题,通过完成综合模拟试题,可使学习者更方便地测试自己理解概念、阅读程序及简单编程的能力。

第 4 章为习题与参考答案,其中的所有程序都是在 TC 环境下调试通过的。应该说明,本书给出的程序并非是唯一正确的解答,对同一个题目可以编出多种程序,本书给出的只是其中的一种,甚至不一定是最佳的一种。对有些题目,给出了两种参考答案,供读者参考和比较,以启发思路。读者在使用本书时,千万不要照抄照搬,本书只是提供了一种参考方案,读者完全可以编写出更好的程序。

本实验指导教材力求语言通俗,算法表达精练,易读易懂,注重应用实践,注重培养学生的综合应用能力和上机动手能力。本书不仅可以作为《C 程序设计教程》的参考书,而且可以作为任何 C 语言教材的参考书;既适用于高等学校或计算机学习班,也可以作为 C 语言程序设计爱好者、程序开发人员和教师的参考用书及其他自学者参考。

本书的出版得到了国防科技出版社的大力支持,在此表示衷心感谢。欢迎使用本书的教师和同学提供宝贵的意见和建议。由于时间仓促和水平有限,书中难免有错误和疏漏之处,敬请专家、读者不吝赐教。

编 者

目 录

第1章 C语言上机环境及操作指导	1
1.1 Turbo C的上机操作	1
1.2 Visual C++的上机操作	12
1.3 常见错误分析	21
第2章 C语言上机实训设计	24
2.1 实训一 C语言运行环境和运行过程	24
2.2 实训二 C语言的基础知识	25
2.3 实训三 简单的C程序设计	28
2.4 实训四 选择结构程序设计	29
2.5 实训五 循环结构程序设计	33
2.6 实训六 数组及其应用	36
2.7 实训七 函数及其应用	40
2.8 实训八 指针	45
2.9 实训九 文件	47
2.10 实训十 综合练习	49
第3章 C语言模拟试题	54
3.1 模拟试题一	54
3.2 模拟试题二	60
3.3 模拟试题三	64
第4章 C语言习题与参考答案	69
4.1 C语言概述	69
4.2 数据类型、运算符与表达式	70
4.3 顺序结构	73
4.4 选择结构	75
4.5 循环结构	82

4.6	数组	87
4.7	函数	90
4.8	编译预处理	97
4.9	位运算	99
4.10	指针	102
4.11	结构体与共用体	108
4.12	文件	111
4.13	模拟题答案	114

第1章 C语言上机环境及操作指导

1.1 Turbo C 的上机操作

通过课堂上学习,我们对 C 语言已有了初步了解,对 C 语言源程序结构有了总体的认识,那么如何在机器上运行 C 语言源程序呢?任何高级语言源程序都要“翻译”成机器语言,才能在机器上运行。“翻译”的方式有两种,一种是解释方式,即对源程序解释一句执行一句;另一种是编译方式,即先把源程序“翻译”成目标程序(用机器代码组成的程序),再经过连接装配后生成可执行文件,最后执行可执行文件而得到结果。

C 语言是一种编译型的程序设计语言,它采用编译的方式将源程序翻译成目标程序(机器代码)。运行一个 C 程序,从输入源程序开始,要经过编辑源程序文件(.C)、编译生成目标文件(.obj)、连接生成可执行文件(.exe)和执行 4 个步骤。

下面主要介绍 Turbo C 2.0 下运行 C 语言源程序。

Turbo C 是美国 Borland 公司推出的 IBM PC 系列机的 C 语言编译程序。它具有方便、直观、易用的界面和丰富的库函数。它向用户提供了集成环境,把程序的编辑、编译、连接和运行等操作全部集中在一个界面上进行,使用十分方便。

1.1.1 上机环境介绍

一、进入 C 语言环境的方法

一个 C 语言程序的实施是从进入 Turbo C 的集成环境开始的,进入 C 语言的环境,一般有两种途径:从 DOS 环境进入和从 Windows 环境进入。

1. 从 DOS 环境进入

在 DOS 命令行上键入:

```
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 集成环境的主菜单窗口,屏幕显示如图 1-1 所示。

2. 从 Windows 环境进入

在 Windows 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;或者从开始菜单中找到“运行”,在运行对话框中键入“C:\TC\TC”,单击“确定”按钮即可。

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,如果你的计算机中 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。

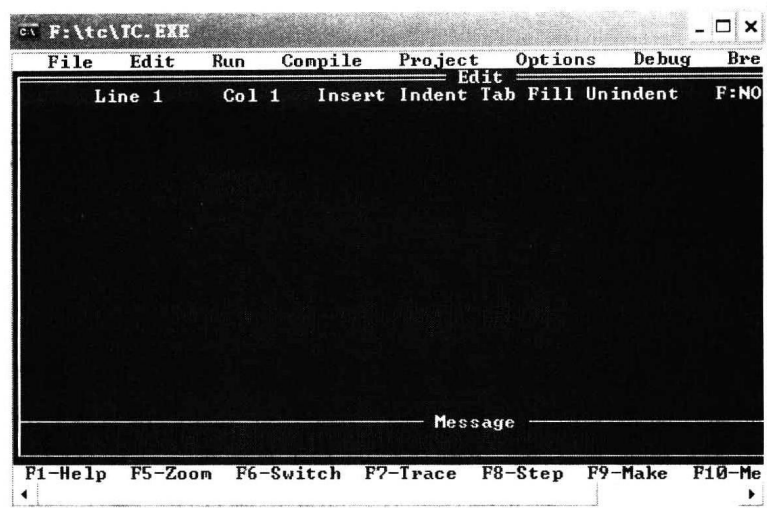


图 1-1

二、TC 窗口的组成

如图 1-1 所示，其中顶上一行为 Turbo C 2.0 主菜单，中间窗口为编辑区，接下来是信息窗口，最底下一行为参考行。这四个窗口构成了 Turbo C 2.0 的主屏幕，以后的编程、编译、调试以及运行都将在这个主屏幕中进行。

主菜单在 Turbo C 2.0 主屏幕顶上一行，显示下列内容，它提供了 8 个选择项。

File: 处理文件(装入、存盘、选择、建立、换名存盘、写盘)，目录操作(列表、改变工作目录)，退出 Turbo C，返回 DOS 状态。

Edit: 建立、编辑源文件。

Run: 自动编辑、连接并运行程序。

Compile: 编辑、生成目标文件组合成工作文件。

Project: 将多个源文件和目标文件组合成工作文件。

Option: 提供集成环境下的多种选择和设置(如设置存储模式、选择编参数、诊断及连接任选项)以及定义宏；也可记录 Include、Output 及 Library 文件目录，保存编译任选项和从配置文件加载任选项。

Debug: 检查、改变变量的值、查找函数，程序运行时查看调用栈。选择程序编译时是否在执行代码中插入调试信息。

Break/Watch: 增加、删除、编辑监视表达式，及设置、清除、执行至断点。

在主菜单中，Edit 选项仅仅是一条进入编辑器的命令。其他选项均为下拉式菜单，包含许多命令选项，使用方向键移动光带来选择某个选项时，按回车键，表示执行该命令，若屏幕上弹出一个下拉菜单，以提供进一步选择。

除 Edit 外，其他各项均有子菜单，只要用 Alt 键加上某项中第一个字母，就可进入该项的子菜单中。下面分别介绍各个菜单的功能。

1. File 菜单

按 Alt+F 键可进入 File 菜单，如图 1-2 所示。

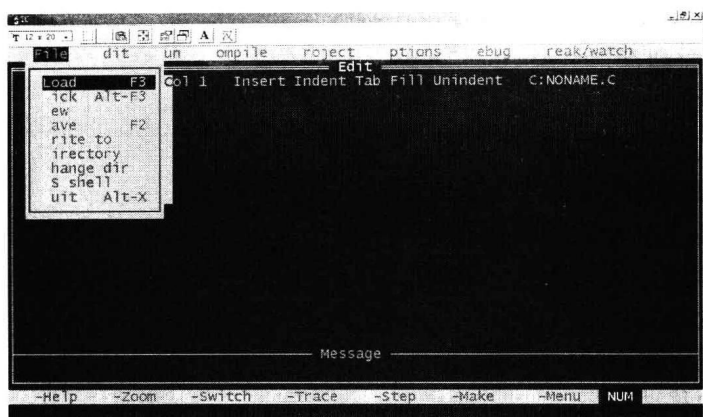


图 1-2

File 菜单的子菜单共有 9 项，分别叙述如下。

(1) Load: 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，即只要按 F3 即可进入该项，而不需要先进入 File 菜单再选此项。

(2) Pick: 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: 新建文件，缺省文件名为 NONAME.C，存盘时可改名。

(4) Save: 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: 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: 显示目录及目录中的文件，并可由用户选择。

(7) Change dir: 显示当前默认目录，用户可以改变默认目录。

(8) Os shell: 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。

(9) Quit: 退出 Turbo C 2.0，返回到 DOS 操作系统中，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，回车则执行。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用 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 键可进入编辑菜单，若再回车，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，可用 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下。

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。

编辑命令简介:

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 {#}

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 为自动缩进开关的控制键。

3. Run 菜单

按 Alt+R 键可进入 Run 菜单, 该菜单内容如图 1-3 所示。

(1) Run: 运行由 Project/Project name 项指定的文件名或当前编辑区的文件。如果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: 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: 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。

(4) Trace into: 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。

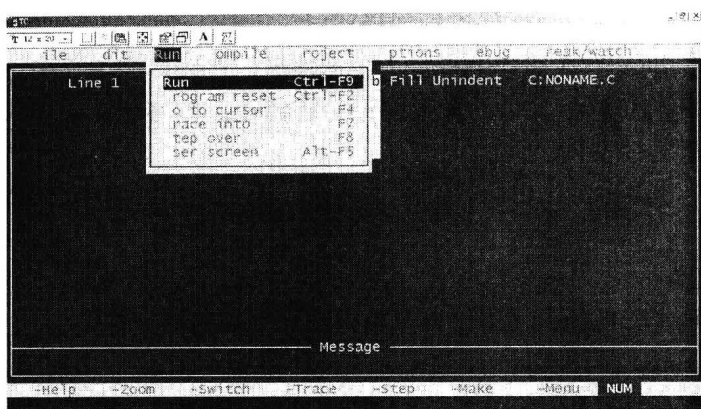


图 1-3

(5) **Step over**: 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**: 显示程序运行时在屏幕上显示的结果。其热键为 Alt+F5。

4. Compile 菜单

按 Alt+C 键可进入 Compile 菜单, 该菜单内容如图 1-4 所示。

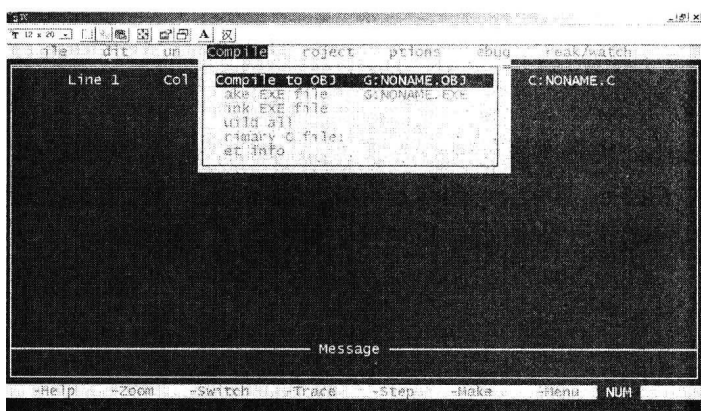


图 1-4

(1) **Compile to OBJ**: 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**: 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:

- ① 由 **Project/Project name** 说明的项目文件名。
- ② 若没有项目文件名, 则由 **Primary C file** 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**: 把当前.OBJ 文件及库文件连接在一起生成.EXE 文件。

(4) **Build all**: 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)。

(5) **Primary C file:** 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 键可进入 Project 菜单，该菜单内容如图 1-5 所示。

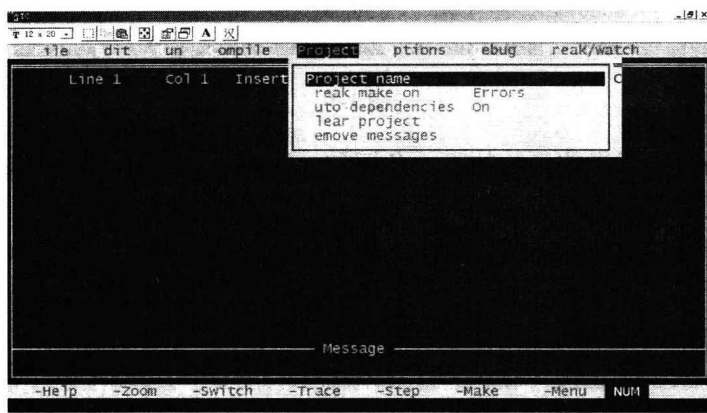


图 1-5

(1) **Project name:** 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
```

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三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:** 由用户选择是否有 Warning、Errors、Fatal Errors 时或 Link 之前退出 Make 编译。

(3) **Auto dependencies:** 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 Project/Project name 中的项目文件名。

(5) **Remove messages:** 把错误信息从信息窗口中清除掉。

6. Options 菜单

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

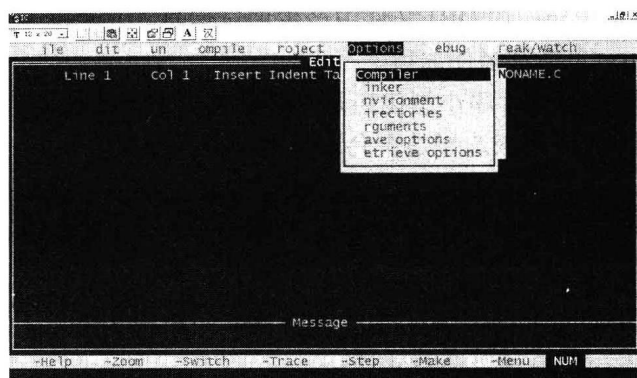


图 1-6

(1) **Compiler:** 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。这些子菜单如图 1-7 所示。

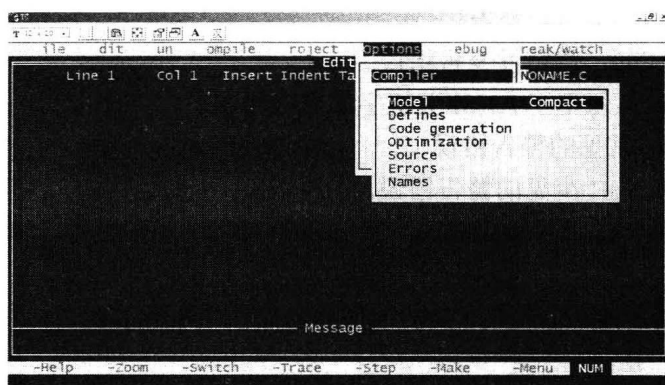


图 1-7

① **Model:** 共有 Tiny、small、medium、compact、large、huge 不同模式可由用户选择。

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

③ **Code generation:** 包括许多任选项, 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: 可选择 8088/8086 或 80186/80286 指令系列。

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。

④ **Optimization**, 包括许多任选项。

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。

⑤ **Source**, 包括许多任选项。

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 C2.0 关键字。

⑥ **Error**, 包括以下选项。

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: 不显示警告。

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。

⑦ **Names:** 用于改变段(segment)、组(group) 和类(class)的名字,默认值为 CODE、DATA、BSS。

(2) **Linker:** 本菜单设置有关连接的选择项, 如图 1-8 所示。

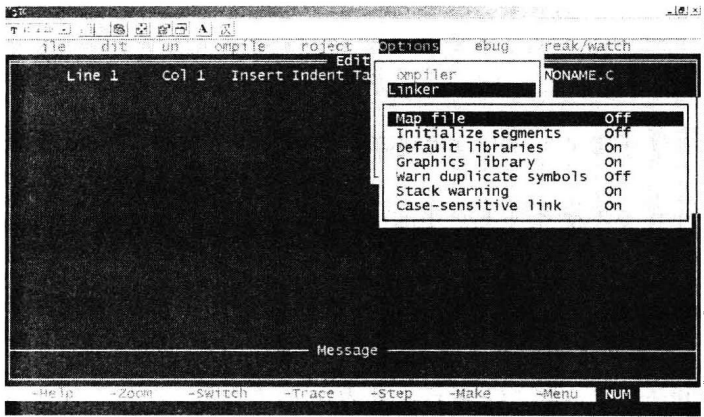


图 1-8

- ① Map file menu: 选择是否产生.MAP 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: 当有重复符号时产生警告信息。
- 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。

(3) Environment: 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, 如图 1-9 所示。

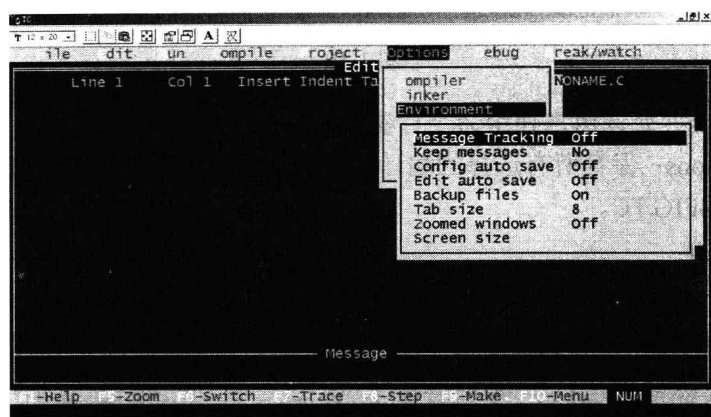


图 1-9

- ① Message tracking: 包括以下选项。

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 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 时不存。

- ④ Edit auto save: 是否在 Run 或 Shell 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后备文件(.BAK 文件)。

- ⑥ Tab size: 设置制表键大小, 默认为 8。

- 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 设置屏幕文本大小。

(4) Directories: 规定编译、连接所需文件的路径, 如图 1-10 所示。

- 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 currentpick file 中取。

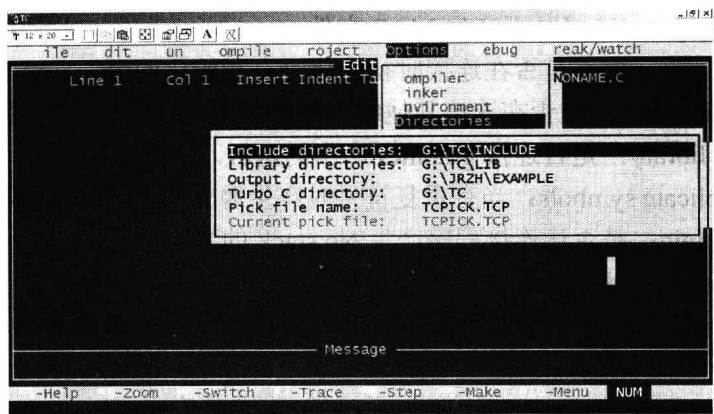


图 1-10

(5) Arguments: 允许用户使用命令行参数。

(6) Save options: 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。

7. Debug 菜单

按 Alt+D 键可选择 Debug 菜单, 该菜单主要用于查错, 如图 1-11 所示。

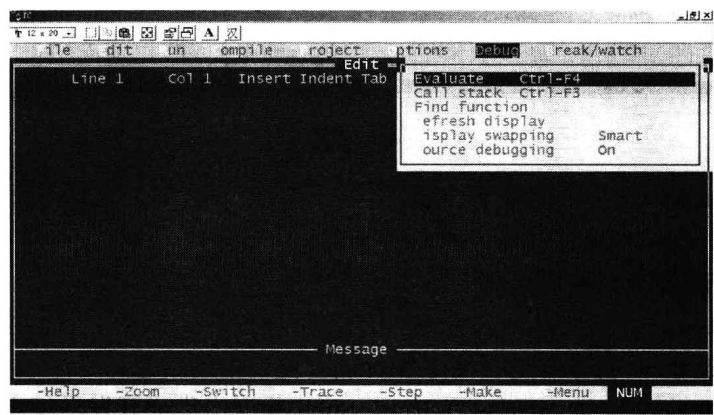


图 1-11

(1) Evaluate 包括以下选项。

① Expression: 要计算结果的表达式。

② Result: 显示表达式的计算结果。

③ New 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恢复编辑窗口的内容。

8. Break/watch 菜单

按 Alt+B 键可进入 Break/watch 菜单, 如图 1-12 所示。

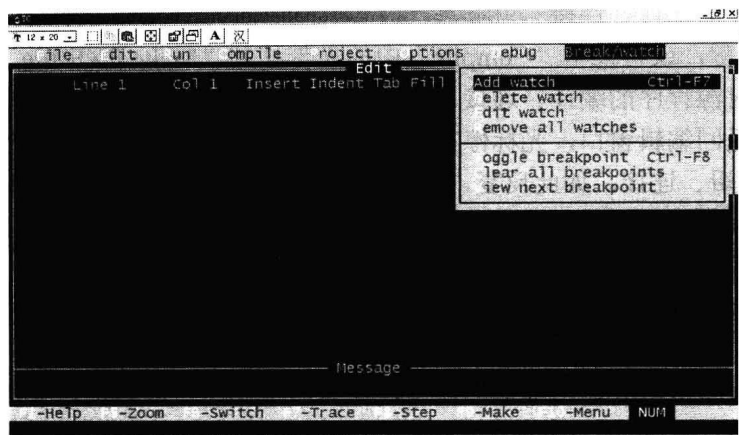


图 1-12

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

1.1.2 上机操作步骤

一、编辑源文件

在主菜单下，直接按 Alt+F 键，或按 F10 键后将光带移到“File”选项上，按回车键，在“File”下面出现一个下拉菜单，菜单中有以下选项：

Load(F3): 表示加载或装入一个文件

Pick(Alt+F3): 从指定的文件列表中选择文件装入编辑器。

New: 表示新文件，缺省文件名为 NONAME.C。

Save(F2): 将正在编辑的文件存盘。

Write to: 将正在编辑的文件写入指定的文件中，若文件名已存在，则重写。

Directory: 表示文件目录。

Change Dir: 改变驱动器及目录

OS shell: 进入 Turbo C 命令行模式，命令 EXIT 可返回集成环境

Quit(Alt+X): 退出 Turbo C，返回 DOS 状态。

建立一个新文件，可用光标移动键将“File”菜单中的光带移到“New”处，按回车键，即可打开编辑窗口。此时，编辑窗口是空白的，光标位于编辑窗口的左上角，屏幕自动处于插入模式，可以输入源程序。屏幕右上角显示缺省文件名为 NONAME.C，编辑完成之后，可按 F2 键或选择“Save”或“Write to”进行存盘操作，此时系统将提示用户将文件名修改成为所需要的文件名。