

Turbo C

科技计算与绘图

陈中亮 编著

● 最新个人微机实用手册丛书 ●

基础知识

基本要素

操作指南

编程技巧

自学提高

华东理工大学出版社

703/2

499

Turbo C 科技计算与绘图

陈中亮 编著

华东理工大学出版社

内 容 提 要

本书共分五章。第一章介绍 Turbo C 基础知识,简单的例程序及必要的操作指南,使读者一开始即可上机操作;第二章详细介绍语言要素,给出大量例题;第三章专门介绍数学计算语言;第四章介绍绘图专用函数;第五章是综合实例。

本书既是 Turbo C 的入门教材,适合自学,又是一本很有参考价值的 C 语言编程手册。大量的例程序可供读者直接模仿、引用及参考,有一定实用价值。

(沪)新登字 208 号

Turbo C 科技计算与绘图

陈中亮 编著

华东理工大学出版社出版发行

上海市梅陇路 130 号

邮政编码 200237 电话 64104306

新华书店上海发行所发行经销

上海展望印刷厂印刷

开本 787×1092 1/16 印张 12.5 字数 307 千字

1996 年 11 月第 1 版 1996 年 11 月第 1 次印刷

印数 1—4 000 册

ISBN 7-5628-0689-6/TP·87 定价 14.50 元

《最新实用个人微机丛书》编委会

万思维	王家贞	王春燕	许东明
陈茜	吴冰	周兰军	罗耀明
胡枋	胡猛	赵伟明	

前 言

编写本书的目的,主要是想为年轻的计算机爱好者们提供一本简明、实用、适于自学的 C 语言教材。

在长期的工作实践中,我经常接触年轻的计算机爱好者。他们精力充沛、思维敏捷,很容易接受最新的事物。他们大多基本掌握一种以上高级语言,如 FORTRAN、BASIC、PASCAL 等。他们中的很多人也已经相当熟练地掌握了 C 语言,他们并不是计算机相关专业的学生,也不是专业的编程工作者,很多人完全靠自学,而且上机的机会也很少,他们为掌握 C 语言所走过的道路给了我很大的启发和鼓励。C 语言作为第二计算机语言学习,并非是大困难的事,问题只是从何处开始。应当为他们提供一本简明、实用、适于自学的教材,帮助他们更好地学习计算机语言,基于此点,开始了本书的编写。

本书分为五章。第一章试图给读者一个总概念,初学者不必完全理解其中的细节。第二章介绍 Turbo C 的基本要素,读者可以结合自己已经掌握的语言参照学习。计算机语言虽然要求准确无误,但对初学者,错误是不可避免的。应当在实际应用的过程中逐步加深理解,结合应用掌握 C 语言的细枝末节。第三章则进一步介绍 Turbo C 的科学计算基本函数,并提供了大量程序实例,供读者选择练习。第四章介绍 Turbo C 的屏幕输出控制技术,使程序运行的结果,包括图形,能按照我们的要求显示在屏幕上。第五章介绍几个实用的程序,帮助读者进一步掌握 Turbo C 的编程技巧。随着学习的进展,读者一定会不断地把本书前后翻阅,并会发现本书具有保存及参考的价值。

虽然本书编写以非计算机相关专业的计算机爱好者为主要对象,但相信本书对已基本掌握一种以上高级语言,而希望进一步学习 C 语言的读者,也是一本适用的自学教材,特别是如果你能有较多上机练习机会的话,那将更好。

本书中提供的全部程序例子都由作者在 286 微机上用 Turbo C 2.0 调试通过后,用计算机将源程序打出付印,并附有详细的中文注释。读者如能自己动手输入计算机中调试并按照自己的想法修改运行,那么这些简单的程序将会给你的学习带来帮助和愉悦。

为方便读者,本书中全部源程序已制成一张软盘,需要的读者可与出版社联系。

作者

1996.9

本书磁盘订购单

本书作者为方便读者学习 Turbo C 语言,特将本书所有例程序集中于一磁盘(3 寸)中,免去读者学习时烦琐的录入工作。本磁盘定价 30 元,订购联系如下:

联系地址: 上海市梅陇路 130 号
联系单位: 华东理工大学出版社
联系人: 本书责任编辑(徐惠娟)
联系电话: (021) — 64772069
邮政编码: 200237

本磁盘以邮购方式发行,请需要者务必通过邮局寄款,并将回寄的地址、姓名、单位、邮政编码等内容写清楚且准确无误。我们在收到订购款后一周内必将磁盘寄出。

目 录

1 C 语言及 Turbo C 简介	(1)
1.1 C 语言概述	(1)
1.1.1 C 语言的产生与发展	(1)
1.1.2 C 语言的特点	(2)
1.1.3 几个简单的 C 语言程序	(3)
1.2 Turbo C 2.0 概述	(4)
1.2.1 Turbo C 的产生与发展	(4)
1.2.2 Turbo C 2.0 的基本配置要求	(4)
1.3 Turbo C 2.0 的使用	(5)
1.3.1 Turbo C 2.0 的安装及启动	(5)
1.3.2 Turbo C 2.0 的集成开发环境简介	(5)
2 Turbo C 语言基本要素	(13)
2.1 数据类型、变量、运算符和表达式	(13)
2.1.1 数据类型	(13)
2.1.2 关键字和标识符	(21)
2.1.3 变量	(21)
2.1.4 运算(操作)符	(31)
2.1.5 表达式	(36)
2.2 输入、输出函数	(38)
2.2.1 标准(控制台)输入、输出函数	(38)
2.2.2 文件的输入、输出函数	(46)
2.3 程序流程的控制语句	(58)
2.3.1 条件语句	(58)
2.3.2 循环语句	(59)
2.3.3 开关语句	(63)
2.3.4 break, continue, goto 和 exit 语句	(64)
2.4 结构、联合和枚举	(69)
2.4.1 结构(struct)	(69)
2.4.2 联合(union)	(77)
2.4.3 枚举	(79)
2.4.4 用 typedef 定义类型	(83)
2.5 函数	(84)
2.5.1 函数的说明和定义	(85)
2.5.2 函数的调用	(87)
2.6 预处理指令	(96)

2.6.1	宏定义 (#define、#undef) 指令	(97)
2.6.2	文件包含 (#include) 指令	(98)
2.6.3	条件编译指令	(98)
2.6.4	其他预处理指令	(100)
3	科学计算常用的库函数	(102)
3.1	舍入函数	(103)
3.2	绝对值	(103)
3.3	乘方和开方	(104)
3.4	三角函数和双曲函数	(106)
3.5	指数函数和对数函数	(108)
3.6	随机数产生函数	(111)
4	字符屏幕和图形函数	(113)
4.1	显示器和图形适配器	(113)
4.2	字符屏幕函数	(114)
4.2.1	屏幕与窗口	(114)
4.2.2	文本窗口的字符属性函数	(115)
4.2.3	文本窗口内的输入输出函数	(118)
4.2.4	有关屏幕操作的函数	(119)
4.3	图形函数	(119)
4.3.1	图形模式的初始化	(120)
4.3.2	屏幕颜色的设置	(122)
4.3.3	图形窗口和图形屏幕操作函数	(122)
4.3.4	基本图形函数	(124)
4.3.5	封闭图形的填充函数	(127)
4.3.6	图形模式下的文本输出	(129)
5	科学计算与绘图程序开发举例	(132)
5.1	线性方程组的求解	(132)
5.2	抛物体运动轨迹的计算和图示	(137)
5.3	三次样条曲线	(141)
5.3.1	三次样条曲线的概念	(141)
5.3.2	约束边界条件下的三次样条曲线	(143)
5.3.3	自由边界条件下的三次样条曲线	(149)
5.3.4	周期边界条件下的三次样条曲线	(155)
5.4	中文下拉式菜单的设计	(163)
附录		
	Turbo C 编译错误信息	(174)
	本书中例程序目录检索	(187)
	本书中表的目录检索	(190)

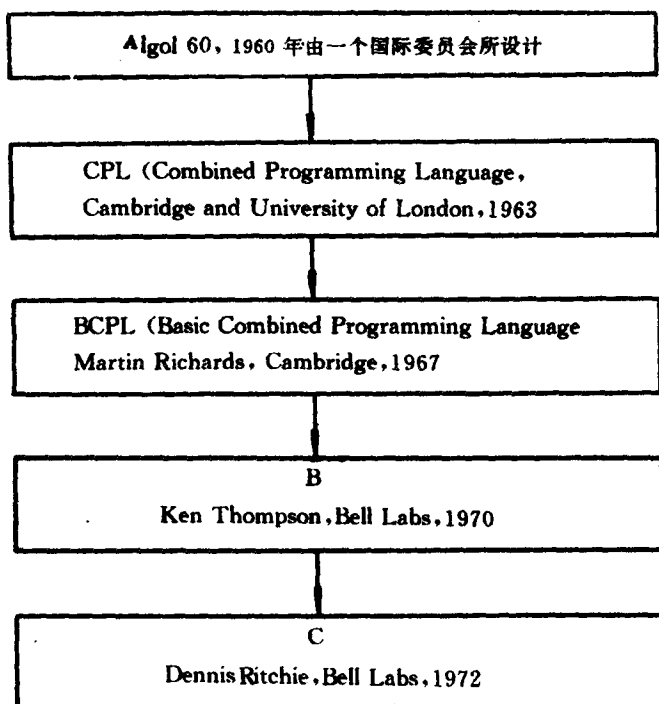
1 C 语言及 Turbo C 简介

1.1 C 语言概述

1.1.1 C 语言的产生与发展

C 语言是贝尔实验室(Bell Laboratories)的 Dennis M. Ritchie 在 1969 ~1972 年间开发而成的。那时候,Dennis 与 Ken Thompson 一起在贝尔实验室为小型机 PDP-7 开发 UNIX 操作系统。开始时的 UNIX 操作系统由 Ken Thompson 用 B 语言写成,而 Dennis 和 Ken Thompson 又用 C 语言重新改写 UNIX 操作系统,使之适用于新的小型机。1979 年,Brian W. Kernigan 和 Dennis M. Ritchie (合称 K&R)写成了一本名为《The C Programming Language》的书。在这本书中介绍的 C 语言后来被称为标准 C。1983 年,美国国家标准化协会(ANSI)根据 C 语言问世以来的各种版本对 C 作了发展和扩充,并制订了标准,一般称之为 ANSI C。1988 年,Brian W. Kernigan 和 Dennis M. Ritchie 把《The C Programming Language》作了改写,C 语言的应用得到进一步的扩展。

有人给 C 画了一个寻根图:



虽然 UNIX 和 C 开始时是为功能强大的小型机写的,但由于微型机的功能不断发展,它们很快被引入微型机中使用。目前,微型机中使用的 C,除了在 UNIX, XENIX 等操作系统中使用之外,还有在 MS-DOS, CP/M-80, CP/M-86 等操作系统使用的 Microsoft C, Turbo C, Quick C 等。与别的计算机语言一样,不同版本的 C 语言之间,虽然基本部分相同,但相互之间又有差别,使用时可查阅有关手册。本书为使读者能更直观地了解 and 掌握书中的内容和例题,只介绍 Turbo C。在不作特别声明时,本书中以后所说的 C 同时也是指 Turbo C。

1.1.2 C 语言的特点

C 语言发展如此迅速,成为最受欢迎的语言之一,显示了强大的生命力。究其原因,当然在于其具有较为独特的优点。归纳起来, C 语言有如下特点:

(1) C 是中级语言,它以高级语言为主体,又对低级语言作了很好的综合。很多高级语言在 C 语言之前早已问世,它们各有特点。FORTRAN 用于科学计算, COBOL 用于工商管理, BASIC 因其简单易学,很受初学者和非专业编程人员欢迎, PASCAL 结构严密,用作教学很是理想,等等。低级语言如汇编语言,编出的程序具有紧凑、灵活、运行速度快的特点;但它的缺点也很突出,如编程乏味、不同的 CPU 有不同的汇编语言等。C 语言则既具有高级语言的功能,又具有低级语言的许多特点,例如能对位、字节和地址进行操作等,功能丰富,表达能力强,使用灵活方便,应用面广。著名的 dBASE III 等软件就是用 C 语言写成的。

(2) C 是结构式语言。结构式语言的显著特点是代码和数据分隔化。程序的各个部分除了必要的信息交流外彼此完全独立,这样可使程序层次清晰,使用、维护和调试方便。从技术上说,模块结构语言允许子程序或函数中再定义子程序或函数,由于 C 语言不允许在函数中建立函数,因此它不是真正的模块结构语言。但 C 语言的主要结构成分是函数,函数是其基本的结构模块,所有的程序活动内容都包含在其中。函数可以在程序中被定义完成独立的任务,独立地编译成目标代码,这样可以实现程序的模块化。函数建立以后,可以在各种情况下正确运行,而不必担心它对程序的其他部分产生不良影响。在大型的软件工程中,可以建立独立的函数这一点至关重要,因为只有这样才能避免不同编程者所编的程序由于偶然的因素而相互干扰。

(3) C 语言是编程者的语言。请看一下 COBOL 语言和 BASIC 语言,前者是为工商管理者使用而写的,语言的设计尽可能地考虑到非编程者也能看懂,后者则基本上是为非专业编程者(初学者)用来编制计算机程序以解决一些相对简单的问题。相比之下, C 语言由一些富有实际经验的专业编程者设计,并在实际使用中不断完善和接受考核,因此最终获得的结果恰恰是专业编程者所需要的。

C 语言很少限制,很少缺陷,模块结构、函数彼此独立,关键字数量很少而又很紧凑。

C 语言生成的目标代码质量高,程序执行效率接近于用汇编语言生成的目标代码的执行效率,却又兼有 Algol 等语言的结构形式。

C 语言功能齐全,数据类型和运算符丰富,特别是引入了指针等概念,使程序效率更高。丰富的图形功能能支持多种显示器和驱动器,可适应各种不同的需要。

用 C 语言编写的程序可移植性好。C 语言能适合 DOS、UNIX 等多种操作系统和多种机型。

每一个专业的 C 编程工作者都可以根据需要建立和维护自己的函数库,这一函数库可以

用于各种不同的程序中。由于 C 语言允许和提倡使用分块编译,因此编程者易于管理大型软件工程。加上众多的专业编程工作者不断地向 C 函数库作出贡献,将大大地减少重复工作。

1.1.3 几个简单的 C 语言程序

下面的 KTG1-1.C 是一个简单的 C 语言程序,尽管它很简单,但它是一个完整的 C 语言程序,它的功能是向屏幕输出一行文字。

/* KTG1-1.C 向屏幕输出一行文字 */

```
#include<stdio.h>                /* 把标准输入输出函数文件包含进来 */
main()                            /* 主函数 */
{
    printf("This is a complete C program. \n");    /* 屏幕输出 */
    getch();
}
```

程序第一行是注解,以“/*”开头,以“*/”结束。程序中的“/*……*/”也是注释部分,可以灵活地出现在程序的任何位置,但“/*”和“*/”必须成对出现。第二行是一个预处理指令,把文件 stdio.h 中的有关内容包括了进来。第三行中 main() 表示主函数,每个 C 程序都必须有一个 main() 函数。而用“{”和“}”括起来的是函数体,其中调用了 printf() 这一标准输出函数,以使“This is a complete C program.”这一行字符按一定的格式输出到屏幕上。“\n”是换行符,语句的结束用“;”号表示。语句“getch();”使程序执行完成后屏幕停留一下,直至键入任一键后退回到集成环境,以便能看清程序执行的结果。

下面的 KTG1-2.C 程序是求两个整数 a 和 b 的和 sum。

/* KTG1-2.C 求两个整数之和 */

```
#include<stdio.h>                /* 把标准输入输出函数文件包含进来 */
main()
{
    int a,b,sum;                /* 变量 a,b 和 sum 定义为整型(int) */
    a=123;b=234;                /* 对 a,b 分别赋值 */
    sum=a+b;                    /* 作运算 a+b,把结果赋给 sum */
    printf("Sum is %d\n",sum);    /* 按一定格式输出结果 */
    getch();
}
```

程序的第五行是两个执行语句,所以有两个分号,但写在同一个程序行中。而第七行除了输出字符串“Sum is”外,还按“%d”格式(十进制整数类型)输出 a+b 的和 sum。因此程序运行的结果在屏幕上输出“Sum is 357”。

程序 KTG1-3.C 把输入的摄氏温度转换成华氏温度输出。与上面的程序相比,增加了键

盘输入部分,请用回车键表示输入结束。

```
/* KTG1-3. C 输入摄氏温度输出华氏温度 */

#include <stdio.h> /* 把标准输入输出函数文件包含进来 */
main()
{
    float c,f; /* 把 c,f 定义为实型变量 */
    printf("Enter your temperature in Centigrade:"); /* 屏幕提示 */
    scanf("%f",&c); /* 键盘输入 */
    f=c*9/5+32; /* 计算 */
    printf("That equals %f Fahrenheit degree",f); /* 输出结果 */
    getch(); /* 等待按任意键 */
}
```

通过以上几个例子,可以对 C 语言程序的组成和形式有一些初步的了解。C 语言程序书写格式比较自由,但为了增加程序的可读性,应当从一开始就养成良好的习惯。学完本章后,即可在 Turbo C 的集成环境中上机试试这些程序了。建议故意写错一些内容,看看集成环境是怎样指出错误的。集成环境在对源程序编译过程中,会在信息窗中尽可能多地指出可能的错误。常见的错误信息提示见本书后的附录。

1.2 Turbo C 2.0 概述

1.2.1 Turbo C 的产生与发展

Turbo C 是美国 Borland 公司的产品。Borland 公司是一家著名的从事软件开发、研制的大公司。该公司相继推出了 Turbo 系列软件,如 Turbo Basic、Turbo Pascal、Turbo Prolog、Turbo Assembler、Turbo Debugger、Turbo C++ 等,都很受用户的欢迎。1987 年, Borland 公司推出 Turbo C 1.0 版本,采用了面目一新的集成开发环境,将文本编辑、程序编译、连接以及程序的调试运行等功能置于一体化的下拉式菜单下,给程序的开发带来了极大的方便。1988 年,该公司又推出了 Turbo C 1.5 版本,增加了图形库和文本窗口函数等。1989 年推出的 Turbo C 2.0 在原来集成开发环境中又增加了查错功能,并可在 Tiny 模式下直接生成 .COM 文件,以及对数学协处理器(8087/ 80187 / 80287)进行仿真等。Turbo C 支持 C 标准版本,支持 ANSI C,备有与硬件有关的特殊函数。Turbo C 还完全兼容于 Turbo C++ 版本。

1.2.2 Turbo C 2.0 的基本配置要求

Turbo C 2.0 对硬件系统的配置要求如下:

- * 适用于 IBM PC 系列机,包括 XT、AT、PS/2 及其他兼容机
- * 需要 2.0 或更高版本的 DOS 支持
- * 至少需要 448K RAM

- * 80 列彩色/单色监视器
- * 至少一个软盘驱动器,建议使用两个软盘或一个硬盘一个软盘的系统
- * 能仿真 8087 协处理器,若系统中有该芯片,将大大加快浮点运算速度

1.3 Turbo C 2.0 的使用

1.3.1 Turbo C 2.0 的安装及启动

Turbo C 2.0 版有两种编译程序,即集成环境编译程序 TC 和命令行编译程序 TCC。系统发行盘有 6 张,安装前应先用 DOS 的 DISKCOPY 命令对发行盘作好备份。安装有两种方式,一是把所有文件用 COPY 命令拷贝到适当的目录下,但一般以使用安装程序安装为好。把含有 INSTALL.EXE 文件的 1# 盘插入 A 驱动器,在 DOS 提示符 A> 下键入:

```
A> INSTALL
```

并按回车,即开始安装。安装有三种选择:

(1) 安装到硬盘中适当的子目录中(例如 C:\TC 中),并根据子目录创建 TURBOC.CFG 文件,规定 Turbo C 系统的结构。

(2) 把原有的 Turbo C 1.5 版本升级为 2.0 版本。

(3) 安装到软盘中,事先须格式化好相应数量的软盘。

本书在以后的介绍中均假定安装到 C 盘的 TC 目录中。安装完成后,在其目录下键入 TC 或 TCC 即可启动集成开发环境或命令行编译程序。

1.3.2 Turbo C 2.0 的集成开发环境简介

集成开发环境(The Integrated Development Environment),简称 IDE,可以用来编辑、编译、连接和运行一个程序而不必脱离 Turbo C 环境。这一环境使软件的生成、检验和调试的速度大大提高。

1. 系统主菜单及其使用

在 TC 目录下键入 TC 并回车,即可启动 Turbo C 并将在屏幕上出现主菜单和版本提示。按任意键将使版本信息消失,屏幕从上到下有四部分组成:

- * 主菜单
- * 编辑程序的状态行和编辑窗
- * 编译程序的信息窗
- * “热键”的快速参考行

屏幕顶部的主菜单行有八项,如下:

```
File  Edit  Run  Compile  Project  Options  Debug  Break/watch
```

其中每项又有若干子菜单项。主菜单各项所包括的功能大致如下:

File:对文件进行装入、保存、选取、建立、写入等操作,退出集成环境操作;

Edit: 建立并编辑源文件;

Run: 自动编译、连接并运行程序;

Compile: 将源程序编译并装配为目标文件或执行文件;

Project: 标识组成程序的若干个文件, 处理工程;

Options: 选择编译操作模式, 如存储模式, 编译选择, 诊断和连接等功能;

Debug: 跟踪排错;

Break/watch: 控制断点及监视表达式。

在执行各功能后用 F10 键可以返回主菜单, 然后用左右光标键选择主菜单项, 可用光标移动键移动光条, 用回车键选择执行项。也可用 Alt 键加主菜单的首字母(F, E, R, C, P, O, D, B)直接进入子菜单。在子菜单状态下, 可用上下光标移动键移动光条, 用回车键选择执行项。或用子菜单的首字母进入执行项。用 Esc 键退出一层菜单。

主屏幕下部有快速功能键参考行, 显示常用的快速功能键, 如下:

F1—Help F5—Zoom F6—Switch F7—Trace F8—Step F9—Make F10—Menu

每个主菜单项下所属各子菜单项简介如下。

(1) File (文件操作)项下有:

Load (F3) 装入一个文件, 可以在提示窗中键入文件名或用类似 DOS 中通配符的方式得到一类文件名, 再用光标键移动光条选择。括号中的 F3 表示快速功能键, 也即热键, 下同。

Pick (Alt-F3) 从以前装入的最多 8 个文件中选择一个装入, 光条在上次的最后一个文件上。

New 进入编辑窗并编辑一个默认文件名为 noname.c 的新文件, 在存盘时可以修改文件名。

Save (F2) 将编辑窗中的文件存盘。

Write to 将编辑窗中的文件以新文件名存盘。

Directory 显示目录及所需文件列表。

Change dir 显示当前目录, 改变驱动器及目录。

Os shell 暂时退出集成环境, 转入 DOS 提示符下, 以执行 DOS 命令。键入 EXIT 命令则重新返回集成环境。

Quit (Alt-x) 退出 Turbo C, 返回 DOS 环境。

(2) Edit 文件编辑, 执行类似 Wordstar 或 WPS 式的编辑操作, 稍后再详细介绍。

(3) Run (程序运行)项下有:

Run (F9) 运行当前编辑的文件或按 Project/Project name 项指定的文件。

Program reset (Ctrl-F2) 中止当前的调试, 程序重启。

Go to cursor (F4) 程序运行到光标所在行, 该行必须是一条可执行语句, 否则将提示出错。

Trace into (F7) 在执行调用其他用户函数的函数时, 本项将跟踪至用户函数的内部。

Step over (F8) 执行当前函数的下一条语句。但不跟踪至被调用的用户函数的内部。

User screen (Alt-F5) 显示程序运行后屏幕上显示的结果。

(4) Compile (编译)项下有:

Compile to .OBJ 将编辑窗中的文件编译为同名的 .OBJ 文件。

Make .EXE file 将生成一个 .EXE 文件。其名称为下面几种可能之一：

- a. 由 Project/Project name 指定的项目文件名。
- b. 如非 a 项,则为由 Primary C file 指明的源文件名。
- c. 如非 a、b 项,则为编辑窗中的文件名。

Link .EXE file 连接当前的 .OBJ 文件及库文件,生成新的 .EXE 文件。

Build all 重新编译 Project 的所有文件,并进行装配生成 .EXE 文件。

Primary C file 主 C 文件,如一经指定,且没有 Project 文件名,则编译时即编译主 C 文件,如编译有错,不管目前编辑窗是何文件,均把主 C 文件调入编辑窗中。

Get info 取得当前路径、源文件名及其大小、编译错误情况等信息。

(5) Project (工程)项下有：

Project name 工程文件名(.OBJ),指定将要编译、连接的文件名。

Break make on Errors 终止文件制作的条件,可以选择在有 Warning, Errors, Fatal errors 时或 Link 连接之前退出 make 编译。

Auto dependencies (on off) 编译时是否自动检查原文件与目标文件的日期和时间关系。

Clear project 清除 Project/Project name 中的 project 文件名。

Remove messages 清除错误信息窗中的信息。

(6) Options (选择)项下有：

Compiler 本项下又有许多子菜单,有些项下又有子菜单,供用户选择硬件配置、存储模式、调试技术、代码优化、对话信息控制等。简介如下：

- a. Model 有 tiny, small, medium, compact, large, huge 六种不同的内存模式可供选用。
- b. Define 宏定义。用户可输入宏定义,多重定义可用分号,赋值可用等号。
- c. Code generation 产生目标代码的方式,可选如下各项：

Calling convention 参数传递方式,可选 C 或 PASCAL。

Instruction set 指令系列选择,可选 8088/8086 或 80186/80286。

Floating point 浮点方式,可选仿真、协处理器或无浮点方式。

Default char type 默认字符类型。

Alignment 规定地址对准原则。

Generate underbars 产生下划线。

Merge Duplicate strings 重复字符串合并。

Standard stack frame 标准栈结构。

Test stack overflow 堆栈溢出测试。

Line number 在 .OBJ 文件中是否产生行号以供调试时使用。

OBJ debug information 在 .OBJ 文件中是否产生调试信息。

d. Optimization 优化

Optimizer for 优化目标:程序大小或运行速度。

User register variable 寄存器变量的使用。

Register optimization 寄存器变量使用的优化。

Jump optimization 通过去除多余的跳转和调整循环等,压缩代码。

e. Source 源文件

Identifier length 标识符长度。

Nested comments 嵌套注释。

ANSI keywords only 只允许 ANSI 关键字还是也允许 Turbo C 关键字。

f. Error 错误

Error;stop after: 多少个错误时停止编译。

Warnings;stop after: 多少个警告时停止编译。

Display warning 显示警告。

Portability warning 显示移植性错误警告,又有七个项目的开关选择。

ANSI violations 显示侵犯 ANSI 关键字错误警告,又有八个项目的开关选择。

common error 常见错误,又有七个项目的开关选择。

less common error 少见错误,又有六个项目的开关选择。

g. Names 改变 CODE, DATA, BSS 的 segment (段), group (组), class (类) 的名字, 一般取默认值。

Linker 本项下又有许多供选择有关连接的开关选项:

Map file 产生 MAP 文件。

Initialize segments 在连接时初始化没有初始化的段。

Default library 缺省库的寻找。

Graphics library 连接 Graphics 库中的函数。

Warn duplicate symbols 警告重复符号。

Stack warning 产生 No stack 的警告。

Case-sensitive link 区分大小写。

Environment 本项下又有许多选择有关工作环境的选项:

Message tracking 文件错误跟踪, 可选 Current file (当前文件), All files (所有文件), Off (不跟踪)。

Keep message 保留错误信息。

Config auto save 在离开集成环境前 (包括 Run, OS shell) 自动保存设置文件。

Edit auto save 在 Run, OS shell 时自动存盘编辑中的文件。

Backup files 是否产生后备文件 (.BAK)。

Tab size 制表键大小。

Zoomed windows 当前窗口放大到整个屏幕 (热键是 F5)。

Screen size 屏幕文本大小。

Directories 默认目录的设置

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

Library directorie 库文件的路径, 多个子目录用“;”分开。

Output directories 输出文件的路径。

Turbo C directories Turbo C 所在的目录。

Pick file name 定义加载 (Pick) 文件名, 如不定义则按 Current pick file 中的执行。

Arguments 允许用户使用命令行参数。

Save options 保存配置文件(默认为 TCCONFIG.TC)。

Retrieve options 装入一个配置文件,并按该配置运行。

(7) Debug (调试)项下有:

Evaluate (Ctrl-F4)

Evaluate 需要计算结果的表达式。

Result 显示表达式的计算结果。

New value 赋给新值。

Call stack (Ctrl-F3) 该项不能进入,在 Turbo C Debugger 时用于检查堆栈。

Find function 该项不能进入,在运行 Turbo C Debugger 时用于显示规定的函数。

Refresh display 如果编辑窗口中的内容偶然被重写了,可用此恢复。

Display swapping 选项为:None Smart Always

Source Debugging 选项为:On Standalone None

(8) Break/watch (断点及监视)项下为:

Add watch (Ctrl-F7) 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监视表达式。

Edit watch 在监视窗口中编辑一监视表达式。

Remove all watches 删除全部监视表达式。

Toggle breakpoint (Ctrl-F8) 对光标所在行设置或删除断点。

Clear all breakpoints 删除全部断点。

View next breakpoint 将光标移到下一个断点处。

2. 编辑窗

按 Alt-E 键即进入编辑窗,在编辑窗内建立一个新源文件的方法有几种:

(1) 在主菜单下选择 File/New,按回车键即打开编辑窗,并开始编辑一个名为 NONAME.C 的新文件。

(2) 在主菜单下选择 File/Load,按回车键,键入源文件名,即打开编辑窗,并开始编辑该文件。如该文件已存在,则装入并接受编辑,如不存在,则创建新文件。

(3) 在主菜单下选择 File/Load,按回车键,修改(或不修改)提示框内的文件名一般表达形式,再次按回车键,屏幕将显示相应的文件目录,用光标移动键选择文件名,并再次按回车键,即打开编辑窗,并开始编辑该文件。

编辑操作所用的命令与 Wordstar 和 WPS 所用的很相似,可参考表 1-1:

对使用过 Turbo PASCAL 或 Turbo BASIC 的读者,Turbo C 的环境是熟悉的。对初学者,这么多命令不用马上记牢。在编辑时只需按下 F1 键,即可获得帮助。同样,对程序的语法有疑问时,只须将光标置于关键词处,然后按下 Ctrl-F1 键,可获得相应的帮助。

至此,读者已能启动 Turbo C,并运行前面的三个例子了。在 C:\TC 目录下键入 TC 并按回车,出现 Turbo C 的主菜单,按前面讲到的方法输入程序,即可按 Alt-C 编译程序,然后根据提示修改可能的错误,直至编译成功。再按 Alt-R 执行程序。