

C 语言实验与习题

主 编 张 洪 礼

电 3. 王 晓 红.



中国农业科技出版社

全国高等职业技术师范教育类专业教材

全国高等农业技术师范教育教材指导委员会审定

C 语言 实 验 与 习 题

主 编 张洪礼 陈 琦 臧怀沛

副主编 高 翔 蔡淑兰

主 审 余梅生

015205/04

TP 312/841

K02

中国农业科技出版社

3

图书在版编目(CIP)数据

C 语言实验与习题/张洪礼,陈琦,臧怀沛编著. —北京:中国农业科技出版社,2000

ISBN 7-80119-962-6

I .C… II .张… III .C 语言-实验与习题 IV .TP312

中国版本图书馆 CIP 数据核字(2000)第 26012 号

责任编辑	左月秋
出版发行	中国农业科技出版社 邮编:100081 电话:(010)68919711; 62173607; 传真:62189014
经 销	新华书店北京发行所
印 刷	河北职业技术师范学院印刷厂
开 本	787mm × 1092mm 1/16 印张:7.500
印 数	1 ~ 2 000 册 字数:178 千字
版 次	2000 年 7 月第 1 版 2000 年 7 月第 1 次印刷
定 价	10.50 元

前 言

本书是针对计算机专业及非计算机专业的本科生、专科生及函授生的需要提供的 C 语言上机指导教材。前面十个实验涉及的是 C 语言的基础内容, 要求学生必须熟练掌握。后面七个综合性实验能提高学生的分析及程序设计能力。另外每个实验后面都有习题, 主要目的是进一步巩固所学的知识。在本教材的第三、四部分给出了 turbo C 的常用函数表和错误信息表, 目的是方便学生编程并尽快找出错误的根源。

实验内容中有的给出了参考程序 (主要是前面几个实验), 更多的需要学生自己编写程序, 所以要求学生必须在每次实验前认真预习本次实验内容, 编写好程序方能参加实验。

编 者
2000. 5.

目 录

第一部分	Turbo C 集成编译环境	(1)
第二部分	C 语言实验	(11)
实验一	熟悉 C 语言运行环境(预备实验)	(12)
实验二	数据描述与基本操作	(14)
实验三	程序的分支	(18)
实验四	循环控制	(23)
实验五	数组的使用	(27)
实验六	函数的使用	(34)
实验七	指针(一)	(40)
实验七	指针(二)	(45)
实验八	结构体与共同体	(48)
实验九	位运算	(53)
实验十	文件	(54)
综合实验一	统计学生成绩	(57)
综合实验二	求矩阵鞍点	(59)
综合实验三	文件应用	(60)
综合实验四	分类与索引	(61)
综合实验五	对分法求非线性方程的根	(63)
综合实验六	求解雅瑟夫问题	(64)
综合实验七	求皇后问题	(66)
第三部分	常用函数表	(67)
第四部分	错误信息	(100)
一、警告信息		(100)
二、Turbo C 编译错误信息		(102)

第一部分 Turbo C 集成编译环境

Turbo C 集成编译环境是一个集程序编辑、编译、连接、调试为一体的 C 程序开发软件，它具有速度快、效率高、功能强、使用方便等优点。用户在这个集成环境下，可以利用内部的编辑器进行全屏幕编辑，利用窗口功能进行编译、连接、调试、运行、环境设置等工作。

如果你的计算机系统已经安装了 Turbo C 编译系统，则在 DOS 命令状态下键入命令
TC

或

TC filename

其中 filename 是用户需要进行编辑、编译、连接、运行的 C 程序的文件名。在前者情况下，该文件名可以在进入集成环境后再指定。

如果 Turbo C 编译系统不是安装在当前目录下，而是安装在别的目录下，并且该目录路径没有打通，则应在 TC 前面加上“路径”，以指出 Turbo C 编译系统所在的位置，但这种情况一般是很少出现的，这是因为，DOS 系统启动时要执行一个自动批处理文件 AUTOEXEC.BAT，在该文件中一般都包含有常用外部命令文件（TC 也属于外部命令）所在的目录路径打通的命令。因此，DOS 系统启动后，在任何目录下都可以很方便地使用外部命令，即在外部命令前不必再加上该外部命令文件所在的目录路径。

进入 Turbo C 集成环境后，首先在屏幕上显示 Turbo C 主菜单窗口，如图 1 所示。

File	Edit	Run	Compile	Project	Options	Debug	Break/Watch

图 1

由图 1 可以看出，在该菜单下，有八个菜单条目，即提供了八种选择。每一个条目的意义如下：

File 处理文件（包括装入、存盘、选择、建立、换名写盘），目录操作（包括列表、改变工作目录），退出系统及调用 DOS。

Edit 建立、编辑源文件

Run 控制运行程序。如果程序已经编辑连接好,且 Debug/Source Debugging 以及 Option/Compiler/Code generation/OBJ Debug Information 开关置为 ON,则可以用此菜单初始化调试阶段。

Compile 编译并生成目标程序与可执行文件。

Project 允许说明程序中包含哪些文件的管理条目 (Project)。

Options 可以选择集成环境任选项 (如存储模式、编译时的任选项、诊断及连接任选项) 及定义宏;也可以记录 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。

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。

特别要指出的是,除了 EDIT 项外,每一个菜单项以对应一个子菜单。而选择 EDIT 项目后,只是进入编辑器。

为了从主菜单中选择所需要的功能,可以用以下两种方式之一:

(1) 按 F10 键后,可以看到屏幕上部主菜单中的某个条目处出现亮块,此时,利用左、右光标移动键 (←与→) 将此亮块移到所要选择的条目位置处,然后按回车 (ENTER) 键,即出现相应的子菜单。

(2) 直接按 ALT+主菜单条目中的首字母 (分别为 F, E, R, C, P, O, D, B) 此时就会出现相应的子菜单。例如,按 ALT+F 表示选择文件子菜单 (FILE)。

当出现子菜单时,其中某个条目是高亮度的,此时可以利用上、下光标移动键 (↑与↓) 来移动该高亮度线,从而选择所需要的功能。

在主菜单或通过主菜单调用的任意一个子菜单中,按 ESC 键后将直接返回到活动窗口。

下面简要介绍各子菜单的功能。

1. 文件子菜单 (FILE)

当选中 FILE 子菜单后,在“FILE”下方将出现一个子窗口,如图2所示。在边个子窗口中,有的条目右边还标出了实现该功能的热键。所谓“热键”,是指为执行菜单中某一固定功能而设置的键。通过热键来实现某种功能,一般要比通过菜单选择更简单直接,但要求用户熟记这些热键。例如,为了选择“文件子菜单 (FILE)”,除了通过主菜单选择以外,还可以直接用热键 ALT+F 来选择。

下面简要说明各项的功能:

(1) Load (加载)

装入一个文件。当给定的文件名中有文件名通配符 (*或?) 时,将进行列表选择。

(2) Pick (选择)

将最近装入进编辑窗口的8个文件列成表,供用户选择,选择后又装入编辑器,光标置在上次修改过的地方。若选了“...LOAD FILE...”,屏幕上将出现“LOAD FILE NAME”提示框。

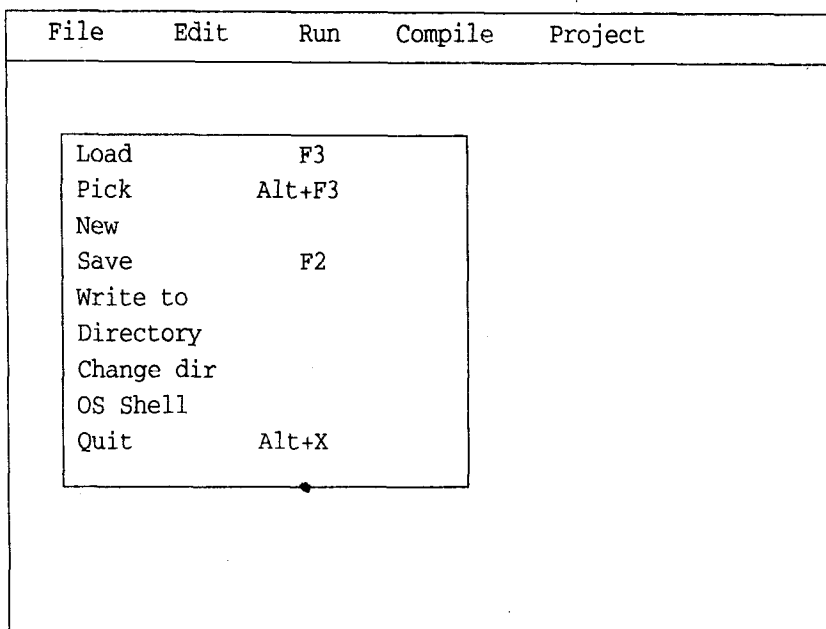


图 2

(3) New (新文件)

说明文件是新的，装入编辑器的缺省文件名为NONAME.C。

(4) Save (存盘)

将编辑器中的内容存盘。若文件名为NONAME.C，而又要存盘，编辑器会询问是否要改名。

(5) Write to (存盘)

把编辑器中的内容写入指定的文件中。若该文件已经存在，则导致重写。

(6) Directory显示目录与所需文件列表（右按回车键则选择当前上当）。热键F4改变匹配符，选择文件名后，将该文件装入编辑器。

(7) Change dir (改变驱动器)

显示当前目录，改变驱动器与目录。

(8) OS Shell (暂时退出)

暂时退出Turbo C，转到DOS状态，在DOS状态下用EXIT命令又可返回Turbo C。此功能对于在想运行DOS命令但又不想退出Turbo C时非常有用。

(9) Quit (退出)

退出Turbo C，返回到DOS状态。

2. 编辑命令 (EDIT)

调用内部编辑器。在编辑器中按F10可返回主菜单（或用ALT加所需主菜单命令的首字母），但此时编辑器中的内容仍保持在屏幕上。在主菜单中按ESC或E键即可回到编辑器（按ALT+E也可，且在任何时候都起作用）。

3. 运行子菜单 (RUN)

当选中RUN子菜单后, 在“RUN”下方将出现一个子窗口, 如图3所示, 其中也列出了对应的热键。

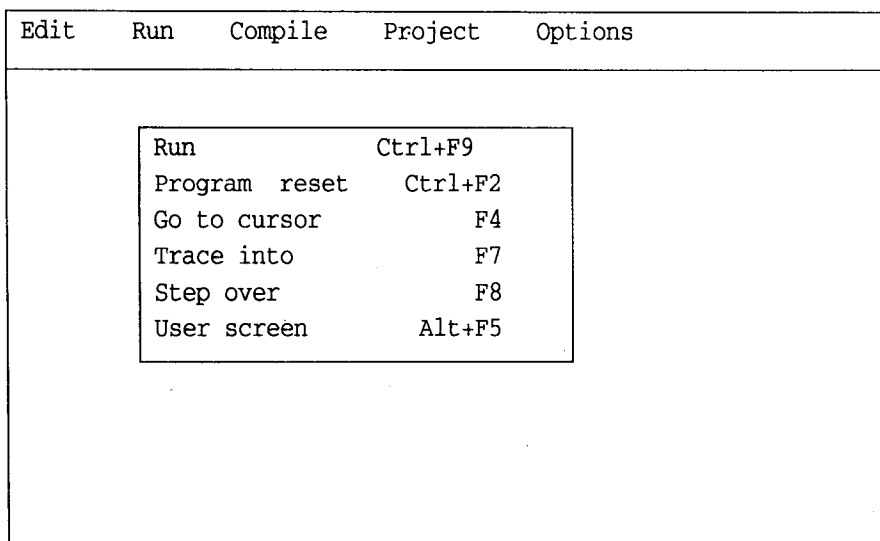


图 3

下面简要说明各项的功能:

(1) Run (运行)

运行当前程序。

(2) Program reset (程序重启动)

中止当前调试, 释放分配给程序的空间, 关闭已打开的文件。

(3) Go to cursor (执行到)

使程序从执行长条运行到编辑窗口中光标所在行。若光标所在行不含可执行代码语句, 则显示一个ESC框作警告。

(4) Trace into (跟踪进入)

运行当前函数中的下一个语句。若此语句不含调试器可访问的函数调用, 则停在下一条可执行语句上; 但若此语句含有调试器可访问的函数调用, 则停在函数定义的开始。

(5) Step over (单步执行)

执行当前函数的下一语句, 即使遇到调试语句可访问的函数调用也不会跟踪进入下一级函数中。

(6) User screen (用户屏幕)

切换到用户屏幕。

4. 编译子菜单 (Compile)

当选中Compile子菜单后, 在“Compile”下方将出现一个子窗口, 如图4所示。

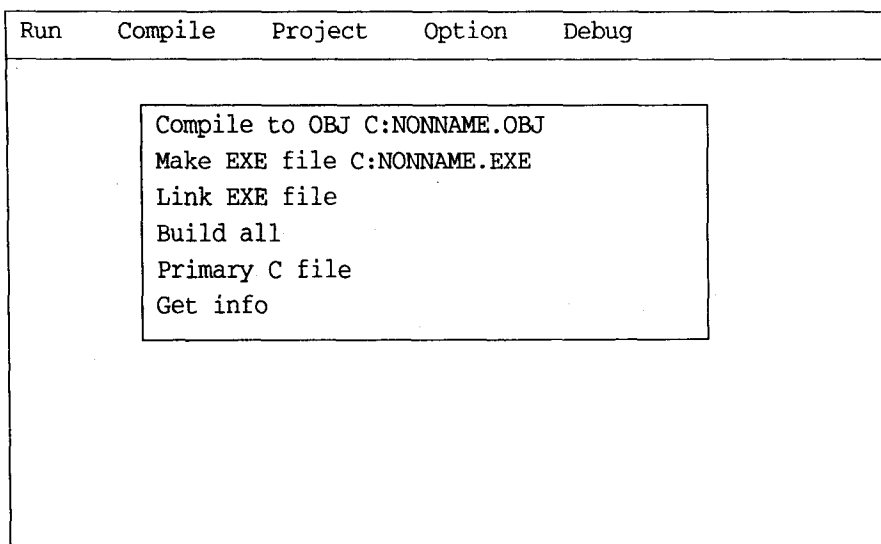


图 4

下面简要说明各项的功能:

(1) compile to obj (编译生成目标码)

本命令将一个.C源文件编译成.OBJ文件, 同时显示生成的文件名。.OBJ文件由源.C文件名产生; 或在没有指定文件名时, 由上次装入编辑器的文件名产生。Turbo C在编译时弹出一个窗口, 用于显示编译结果。在编译/组装(MAKE)完后, 按任一键将清除编译窗口。此时若发现有错误, 则转到消息窗口的第一个错误处(有亮度标志)。

本命令的热键为ALT+F9。

(2) Make EXE file (生成执行文件)

本命令调用用来生成.EXE文件, 并显示所生成的.EXE文件名。.EXE文件名是依次由下列文件名产生的:

Project/Project Name说明的文件名;

或Project C File说明的文件名;

或上次装入窗口的文件名。

本命令的热键为F9。

(3) link EXE file (连接执行文件)

把当前文件与库文件(既可以是缺省的, 也可以是定义在当前项目文件中的)连接在一起, 生成EXE文件

(4) Built All (建立所有文件)

重建项目中的所有文件。本命令类似Compile/make EXE File, 只是它是无条件执行的, 而Compile/make EXE File只重建那些非过时的文件。本命令首先将所有的Project文件中的.obj的日期与时间置为0, 然后再组装(make)。这样, 若用户因Ctrl+Break键中断了Build

All命令，只要用Compile/make EXE File即可恢复。

(5) Primary C file (主C文件)

当编译多个.H头文件单个.C文件时，Primary C File命令是很有用的（但并非必要的）。若在编译过程中发现错误，包含错误的文件（.C或.H）将被自动装入编辑器，可对其修改。但必须注意，.H文件只有在已将Option/Environment/Message Tracking缺省设置改为All File时才能自动装入，而原缺省设置不会自动加载.H文件。即使.C文件不在编辑器，但只要一按Alt+F9，.C主文件即被重新编译。

(6) Get Info (获得信息)

Compile/Get Info开辟一窗口，给出如下信息：

源文件；

与当前文件相联系的目标文件名；

当前源文件名；

当前源文件字节数；

程序退出码；

可用空间。

5. Project子菜单

当选中Project子菜单后，在“Project”下方将出现一个子窗口，如图5所示。

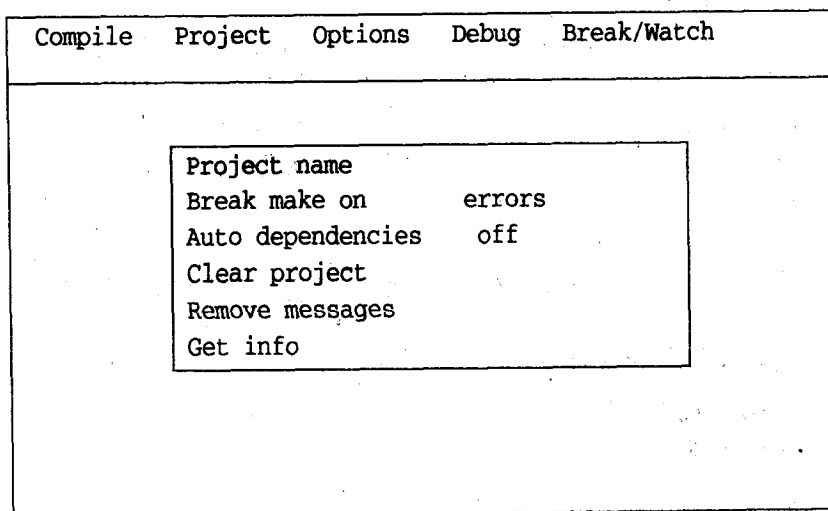


图 5

下面简要说明各项功能：

(1) Project name

选择一个包含将要编译连接的文件名的Project文件，项目名也将是以后要建立的。EXE或MAP文件名。典型的项目文件具有.PRJ扩展名。

(2) Break make on

提供用户说明中止make的缺省条件，如警告 (Warnings)、错误 (Errors)、致命错误 (Fatal

Error)。

(3) Auto dependencies(自动依赖)

这是一个开关。当置On为时,项目组装(Project-Make)自动检查每个项目表中在磁盘上有相应.C文件的那些.OBJ文件的源文件的日期/时间信息与.OBJ文件的依赖关系。所谓自动依赖关系检查是指:项目组装打开.OBJ文件,寻找包含在源代码的那些文件的有关信息。这种信息总是在编译源模块时即被TC或TCC放进.OBJ文件了。此时,把每个组成.OBJ文件的日期/时间信息与.OBJ中的进行比较,若不同,则重新编译.C源文件。

若Auto dependencies开关置为off,则不进行这种检查。

(4) Clear project(清除project)

改命令清除项目文件名,重置消息窗口(Message Window)。

(5) Remove message(删除信息)

该命令把错误信息从消息窗口中清除掉。

6.Option子菜单

当选中Option子菜单后,在“Option”下方将出现一个子窗口,如图6所示。

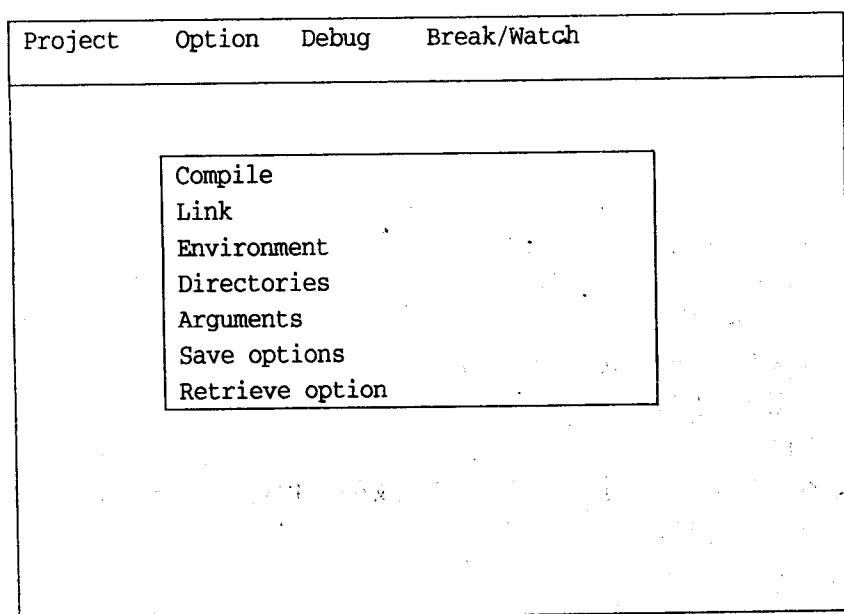


图 6

下面简要说明各项的功能:

(1) Compile(编译器)

本命令将产生一个子菜单,为用户提供说明硬件配置、存储模式、调试技术、代码优化、诊断消息控制以及宏定义等。各菜单条目如下:

Model(选择存储模型)

Define(打开一个宏定义框)

Code generation(代码生成)
Optimization(优化用户代码)
Source(处理源代码)
Errors(处理和响应诊断信息)
Names(改变代码、数据等)

其中每一个条目又对应一个子菜单,供用户选择各种功能。详细介绍请参看 Turbo C 的用户手册。

(2) Linker(连接器)

本命令将产生有关连接器的设置。它包括以下内容:

Map file(选择映射文件的类型On/Off, 缺省值为Off)
Initialize segments(段初始化On/Off, 缺省值为Off)
Default libraries(缺省库On/Off, 缺省值为On)
Graphics library(图形库On/Off, 缺省值为On)
Warn duplicate symbols(警告重复字符On/Off, 缺省值为On)
Stack warning(堆栈警告On/Off, 缺省值为On)
Case-sensitive link(大小写区别连接On/Off, 缺省值为On)

(3) Environment(环境设置)

本命令将产生编译环境的设置。它包括以下内容:

Message tracking(消息跟踪Current File/All Files/Off,缺省值为Current File)
Keeping message(保存消息Yes/No, 缺省值为No)
Config auto save(配置自动保存On/Off, 缺省值为On)
Edit auto save(编辑自动保存On/Off, 缺省值为Off)
Backup files(备份文件On/Off, 缺省值为On)
Tab size(制表键大小, 缺省值为8)
Zoomed window(放大窗口On/Off, 缺省值为Off)
Screen size(选择屏幕显示行数)

(4) Directories(目录)

本命令告诉Turbo C到那里去寻找编译连接所需的文件,生成的可执行文件放到何处,在哪里查找配置文件。具体内容如下:

Include directories(包含目录)
:C:\TURBOC\INCLUDE;C:\TURBOC\IN
Library directories(库目录):C:\TURBOC\LIB
Output directory(输出目录):
Turbo C directory(Turbo C目录)
Pick file name(pick文件名)
Current pick file(当前pick文件)

(5) Argument(参数)

本设置允许用户给出运行程序命令行。

(6) Save option (保存任意项)

将选择的编辑器、连接器环境、调试和project任选项保存到一个配置文件中(缺省文件名为TCCONFIG.TC)。启动时,Turbo C再到TURBOC目录中去寻找同样的文件。

(7) Retrieve option(恢复任选项)

加载以前用Option/Save options命令保存的配置文件。

7. Debug子菜单

当选中Debug子菜单后,在“Debug”下方将出现一个子窗口,如图7所示

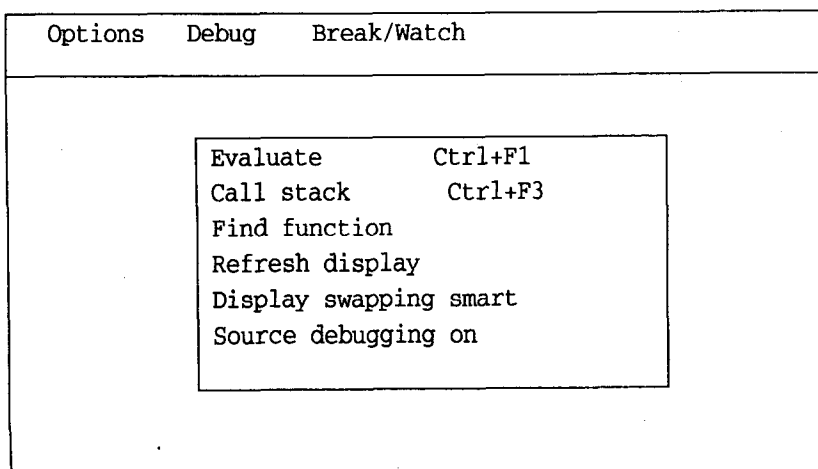


图 7

下面简要说明各项功能:

(1) Evaluate(计算)

计算变量或表达式值,并显示其结果。

(2) Call stack(调用栈)

本命令显示一个调用栈的弹出窗口。调用栈显示程序运行到正在运行的函数时调用的函数序列。其中主函数main在栈底,正在运行的函数在栈顶。调用函数的每一项显示了函数名以及传递给它的参数值。

(3) Find Function(查找函数定义)

显示编辑窗口每一函数的定义。只有在调试阶段才能使用本命令。

(4) Refresh display(刷新显示器)

万一编辑屏幕被重写,使用本命令可以恢复当前屏幕的内容。

(5) Display swapping (显示转换)

本命令提供三种选择: On(缺省)、Always 和 None。

(6) Source debugging(源代码调试)

本命令提供三种选择On(缺省)、Standalone和None。

8. Break/Watch子菜单

当选中Break/Watch子菜单后，在“Break/Watch”下方将出现一个子窗口，如图8所示。使用本命令可以控制断点和监视表达式。

下面简要说明各项的功能：

(1) Add watch(增加监视表达式)

向监视窗口插入一个监视表达式。

(2) Delete watch(删除监视表达式)

从监视窗口中删除当前监视表达式。

(3) Edit watch(编辑监视表达式)

选择本命令后，调试器弹出一个含有当前监视表达式拷贝的窗口。

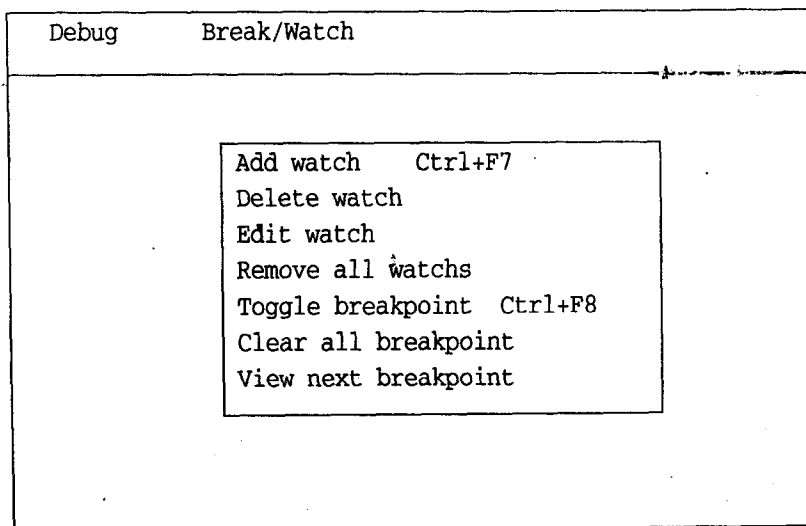


图 8

(4) Remove all watch(删除所有监视表达式)

将所有监视表达式从监视窗口中删除。

(5) Toggle breakpoint(打开或关闭断点)

设置或除去去光标所在断点。

(6) Clear all breakpoint(清除所有断点)

从程序中删除所有断点。

(7) View next breakpoint(显示下一个断点)

将光标移到程序中的下一个断点。

第二部分 C 语言实验

在学习 C 语言程序设计的过程中，上机实验是十分重要的环节，通过实验，可以加深对 C 语言功能特征、语法规则、程序编译与运行等基本概念和基本方法的理解和运用。通过上机调试程序，使学生能及时发现程序编制中出现的错误并找到修改正方法，提高学生的独立编程能力和编程技巧，为 C 语言在后续课程中的应用打下坚实的基础。

实验报告要求

1. 每次实验前，认真预习本次实验内容，按实验指导书的要求，需编写的程序，应书写整齐，经检查无误后方能上机。
2. 上机输入和调试程序，调试通过后，打印出程序清单并把运行结果记录下来（在条件允许下）。
3. 上机结束后，按照实验指导书的具体要求，整理出实验报告（字迹工整），下次上机交给指导教师。
4. 实验报告应包括以下内容：
 - (1) 实验题目。
 - (2) 算法说明（复杂的可用流程图表示）。
 - (3) 程序清单（有条件用打印机打印出来）。
 - (4) 运行结果。
 - (5) 对运行情况作分析，以及本次实验所取得的经验。如程序未能通过，应分析错误原因。

在实验内容里有“*”的部分为选做题，有条件的学生可做这部分。

实验一 熟悉 C 语言运行环境

(预备实验)

一 实验目的要求

1. 学会 Turbo C 2.0 的安装方法 (参看本书第一部分), 熟悉 C 语言程序的运行环境, 了解所用计算机系统软、硬配置。
2. 初步了解在该集成环境下如何编辑、编译、连接和运行一个 C 程序, 即运行一个 C 程序的全过程。
3. 通过运行简单的 C 程序, 初步了解 C 程序的基本结构及特性。

二 实验内容和步骤

1. 从开机开始进行操作, 熟悉一些常用的 DOS 命令, 包括如何建立子目录, 文件拷贝, 删除文件等。
2. 建立自己的子目录, 以备存放文件。
3. 进入 Turbo C 集成环境, 熟悉 Turbo C 主菜单下各选择项的功能及功能键的使用。
4. 输入一简单 C 程序 (可用教科书上的例题), 了解 C 程序运行的全过程。
5. 编写用 printf 语句将 3 个字符串: good morning, floppy disk, hard disk 在同一行显示的程序。

程序例

```
main()
{ printf ("good morning"); /*显示不换行*/
  printf ("floppy disk");
  printf ("hard disk \n");
}
```

运行结果: good morning floppy disk hard disk

6. 把上面的程序改为每行显示一个字串, 应如何修改程序, 并运行之。
7. 编写一程序, 用键盘输入语句输入三个数, 然后分别求它们的和、积及求余。

三 习题

- 1 以下各标识符中, 哪些是合法的用户标识符?

- (1) main (2) MAIN (3) a205 (4) _exp (5) a_b
(6) 3.5 (7) A[10] (8) A.name (9) %d (10) float
(11) \$100 (12) I am (13) max(5) (14) begin (15) #define
(16) NONAMEOO (17) proc (18) and (19) not (20) While

注: 字母或下划开始的后跟若干字母或下划线或数字组成的序列; 不能和保留字相同; 避