



C/C++ 程序设计 实验教程

(第三版)

夏宝岚 夏耘 编著



华东理工大学出版社

EAST CHINA UNIVERSITY OF SCIENCE AND TECHNOLOGY PRESS

C/C++程序设计实验教程

(第三版)

夏宝岚 夏 耘 编著

 华东理工大学出版社

图书在版编目(CIP)数据

C/C++程序设计实验教程/夏宝岚,夏耘编著.—3版.上海:华东理工大学出版社,2006.1

ISBN 7-5628-1824-X

I. C... II. ①夏... ②夏... III. C语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 152271 号

C/C++程序设计实验教程(第三版)

.....

编 著 / 夏宝岚 夏 耘

责任编辑 / 张 波

封面设计 / 王晓迪

责任校对 / 许 春

出版发行 / 华东理工大学出版社

地 址:上海市梅陇路 130 号,200237

电 话:(021)64250306(营销部)

传 真:(021)64252707

网 址:www.hdlgpress.com.cn

印 刷 / 上海崇明裕安印刷厂

开 本 / 787×1092 1/16

印 张 / 11

字 数 / 277 千字

版 次 / 2002 年 2 月第 1 版 2006 年 1 月第 3 版

印 次 / 2006 年 1 月第 1 次

印 数 / 34181—42230 册

书 号 / ISBN 7-5628-1824-X/TP·142

定 价 / 17.00 元

(本书如有印装质量问题,请到出版社储运部调换)

内 容 简 介

本书全篇由四个部分组成。第一部分为基础知识,对 C 程序的开发环境 Turbo C 和 Visual C++作了全面、详细的介绍,作为学习和使用 C 和 C++语言的基础。第二部分为实验操作,共安排了 15 个实验,其内容、节奏与“C/C++程序设计”课程的理论教学完全同步。第三部分为应用技巧,介绍了初学者在编程中常犯的 20 多种错误及其排除方法,调试程序的技巧以及提高程序运行效率的几点建议,这部分内容对提高学习者的排错、纠错能力和程序设计水平有极大的帮助。最后一部分为附录,其中附录 A 列出了详尽的编译出错信息,同时给出了造成出错的可能原因以及排除的方法;附录 C 给出全书实验的参考答案,为学生的独立自学提供帮助。

本书既适合用作高等院校“C/C++程序设计”课程配套的实验指导书,又可用作《C/C++程序设计》的参考书,同时也可用作初学者学习相关知识和技能的启蒙教材。

前 言

为适应计算机技术的飞速发展以及计算机应用的不断深入,提高学生程序设计能力,扩展程序设计的软件平台,使学生能够快速适应并掌握当今流行的面向对象的程序设计理念与方法,毕业后能够顺利择业并较快地胜任自己从事的工作,本实验教程与《C/C++程序设计》教材同步改编,将传统的面向过程的程序设计 Turbo C 与现代的面向对象的程序设计 C++ 以及可视化编程环境 Visual C++ 进行有机的结合与重建。

为了让初学者有一个循序渐进的适应过程,本实验教程对原有的内容进行了优化,在章节顺序上与《C/C++程序设计》教材保持同步,在实验内容上增加了四个 C++ 实验,这些实验都必须在 Visual C++ 视窗环境下完成编辑、调试和运行。本教程不仅给出了较全面的 C++ 实验内容,而且对面向对象程序设计的方法和实现,以及视窗平台 Visual C++ 作了细致、通俗、全面的介绍,包括如何调试 C++ 程序,如何在 Visual C++ 环境下编辑、调试、运行 C 程序等内容。

本书结构编排合理、语言描述精练、选题新颖。除此之外,与其他同类型教材相比较还具有以下与众不同的特色。首先,在给出每个实验具体的操作内容之前,作者都全面、透彻地阐述了与本次实验有关的基本概念和知识要点,学习者阅读以后,不仅为顺利完成本次实验操作可获得帮助,而且可以起到对本章节所学的理论知识进行全面、系统复习与回顾的作用。其次,本书突破了传统的实验题型的框架,并非就事论事地给出几道枯燥的、毫无启发性的题目,让学习者自己去编制程序,而是采用了丰富多样的形式,其中有:给出现成的程序让学习者模仿,给出一个简单的程序让学习者去扩充功能,给出一个有错的程序让学习者纠错、改错,给出一个功能不全的程序让学习者填空补缺,给出一道题目和程序说明让学习者独立编程,以及考查学习者综合运用能力的不同档次的习题。其三,在实验题的选择与编排上遵循深入浅出、由简至繁的原则,使读者在阅读和练习时循序渐进、轻松自如。经过近几年的教学实践已充分证明,一些原先并无程序设计基础、特别害怕编程的初学者,借助于本实验教程的学习后,大多都能较快掌握 C 语言的学习方法,程序设计能力也有不小的提高。因此本教程既是一本理想的实验指导书,又是一本有价值的《C/C++程序设计》的参考书。

本书在编写过程中得到了华东理工大学信息学院邵志清教授以及龚正良、黄建华、刘江、滕虹等诸位老师的热情帮助与指导,龚骏博士审阅本书后,还提出了不少宝贵意见,并为部分题目提供了具有优良算法的题解,在此一并表示衷心的感谢。

编 者

2005 年 12 月

目 录

1 基础知识

1.1 C 程序开发环境简介.....	1
1.1.1 Turbo C	1
1.1.2 Visual C++	1
1.2 Turbo C 编译系统.....	2
1.2.1 Turbo C 主屏幕的组成	2
1.2.2 TC 主菜单	3
1.2.3 系统状态切换	8
1.2.4 在 Turbo C 2.0 系统上编程	8
1.3 Visual C++	13
1.3.1 Visual C++ 6.0 集成开发环境简介	13
1.3.2 编辑、编译、运行C++程序	14
1.3.3 C++程序的调试	18
1.3.4 如何在 Visual C++环境下编辑、编译、运行 Turbo C 程序	18

2 实验操作

2.1 Turbo C 实验操作	21
实验一 Turbo C 操作初步	21
实验二 设计并运行简单的 C 程序	25
实验三 输入与输出	29
实验四 选择结构的程序设计	33
实验五 循环结构的程序设计	40
实验六 数组	47
实验七 函数	52
实验八 指针	62
实验九 结构	69
实验十 文件	78
实验十一 图形处理	84



2.2 C++实验操作	87
实验十二 Visual C++操作初步	87
实验十三 对象和类	90
实验十四 继承与派生	97
实验十五 多态性与虚函数	105
3 应用技巧	
3.1 初学者编程中常犯的错误	112
3.2 调试程序的技巧	119
3.2.1 在 Turbo C 集成环境下调试	119
3.2.2 在 Visual C++视窗环境下调试	121
3.3 有关程序效率的几个问题	122
附录	
附录 A 编译出错信息	124
附录 B 常用字符与 ASCII 码对照表	137
附录 C 实验参考答案	138



1 基础知识

本章主要介绍目前最流行的两个 C 编译系统:Turbo C 2.0 和 Visual C++ 6.0,介绍在这些系统上编辑、编译、运行 C 程序的方法和步骤,并初次尝试编程实践。

1.1 C 程序开发环境简介

C 语言的标准已经被大多数 C 和 C++ 开发环境所兼容,因此,今天我们可以使用相当多的工具来开发自己的 C 程序。例如,下面将要介绍的 Turbo C 和 Visual C++ 就是当今最流行的 C 程序开发工具。

1.1.1 Turbo C

Turbo C 编译系统是近几十年来最为流行的 C 程序开发环境。它以自身的小巧、易用和高效超越了绝大多数的 C 程序开发工具,经历了时间与实践的考验,被证实确实是一款优秀的 C 编译器。

Turbo C 是基于 DOS 操作系统的,欲在磁盘上建立 Turbo C 系统,应首先在硬盘上建立一个名字为 TC 的子目录,把 Turbo C 系统的全部文件都装入到该子目录下才可运行该集成环境;但是为了便于管理和数据存取,在 TC 子目录下再建两个名字分别为 Include 和 Lib 的二级子目录,用于存储嵌入文件和库文件,这样用户就可方便地从集成环境版本转换到命令行版本。

1.1.2 Visual C++

近些年来,随着 C++ 语言的崛起,Visual C++ 开发环境也逐渐流行起来。与 Turbo C 相比,它在开发程序的易用性方面又上了一个台阶,又为我们提供了一个基于 Windows 操作系统的 C 程序开发平台。

Visual C++ 是基于 Windows 和 Windows NT 32 位应用程序的可视化开发工具,对 Internet 有很强的支持,因此用它能够开发出规模更大、功能更复杂的 32 位 Windows 应用程序,包括数据库应用程序、Internet 应用程序和 Active X 应用程序等。

但是,必须指出的是:Visual C++ 集成开发环境内容相当多,功能十分繁杂,使用者必须具有一定的程序设计基础。与此相比,Turbo C 具有小巧、简单、易学、编译速度快、目标程序效率高的优点,因此对于初学者来说,明智的做法应该是先从 Turbo C 入手,掌握 C 语言的语义、语法和程序设计的基本方法,然后再循序渐进学习并掌握 C++ 语言以及 Visual C++ 程序



设计的方法。下面我们对Turbo C 2.0 和 Visual C++ 6.0 两种 C 程序开发环境作较详细的介绍。

1.2 Turbo C 编译系统

1.2.1 Turbo C 主屏幕的组成

本书介绍 Turbo C 2.0 系统。若事先对 tc.exe 文件所在的路径已经作了设置或已经创建了指向该文件的快捷方式,就可以直接启动 Turbo C 了。Turbo C 一旦启动,屏幕立即显示 TC 主屏幕和版本信息(见图 1.1),按任意键即可关闭所显示的版本信息。

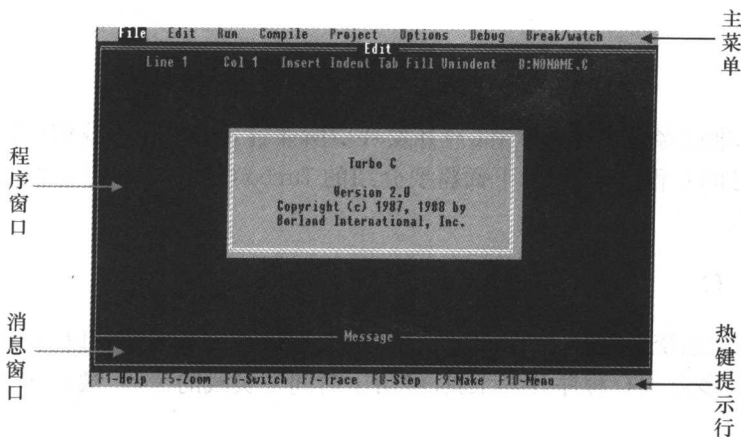


图 1.1 TC 主屏幕

Turbo C(以下简称 TC)主屏自上而下分成主菜单、程序窗口、消息窗口和热键提示行四部分。

1. 主菜单

为用户提供了用于文件操作的一切命令。

2. 程序窗口

用于输入和编辑源程序,按 F10 功能键或按 ALT-E 热键或单击 EDIT 菜单并按回车键均可激活编辑光标,进入源程序的输入和编辑状态,此时窗口顶部将显示与当前编辑状态有关的信息行,形式如下所示:

```
line×× col×× Insert Indent Tab Fill Unindent ××.c
```

line ×× 指示当前光标所在的行;

col ×× 指示当前光标所在的列;

Insert 表示当前为插入状态,单击此键可以在插入状态与替换状态之间切换;

Indent 首行自动缩进。即输入的新行自动与上一行的行首对齐,使用 Ctrl-O-I 热键可控制 on 和 off 的切换;



Tab	制表模式。on 时,按一次向右跳 8 格,使用 Ctrl-O-T 热键可控制 on 和 off 的切换;
Fill	填充模式。当 Tab 为 on 时,每行的开始处自动填入适当的制表符和空格,使用 Ctrl-O-F 热键可控制 on 和 off 的切换;
Unindent	对齐模式,按一次退格键(backspace 键),使当前行对齐上一行的第一个非空字符,使用 Ctrl-O-U 热键可控制 on 和 off 的切换;
××.c	当前编辑文件所在的路径和名字。

3. 消息窗口

用于显示系统提示信息。例如:编译时显示编译出错信息,调试程序时显示监视变量和表达式的值等。

4. 热键提示行

为用户提供了一组快速执行菜单命令的功能键。热键提示行的内容随活动光标所处的位置不同而不同,当光标位于菜单或程序窗口时,提示行内容如下:

F1(help)	打开帮助窗口,获取有关编辑命令的有关信息;
F5(zoom)	放大或缩小当前活动窗口;
F6(switch)	光标在编辑窗口与活动窗口之间切换;
F7(trace)	单步执行程序,跟踪到函数;
F8(step)	单步执行程序,不跟踪到函数;
F9(make)	编译、链接生成.exe 文件;
F10(menu)	光标在主菜单与编辑窗口之间交替切换。

当光标位于监视表达式的消息窗口时,热键功能如下:

F1	打开 help 窗口,获取如何使用错误跟踪的帮助信息;
F5	将消息窗口扩大至全屏幕或还原;
F6	激活程序窗口,即使光标跳入程序窗口;
INS	在 watch 窗口内增加一个监视表达式;
DEL	在 watch 窗口内删除一个监视表达式;
ENTER	编辑当前的监视表达式。

1.2.2 TC 主菜单

Turbo C 2.0 集成环境的主菜单有 8 个菜单栏目,它们的名字和功能见表 1.1。

表 1.1 Turbo C 2.0 菜单组成

菜单名称	功 能
File	建立、打开、保存文件,管理目录,暂转 DOS 和退出 Turbo C
Edit	建立、编辑源文件
Run	编译、链接和运行程序
Compile	对源程序文件进行编译,生成目标文件和可执行文件并查错报错
Project	生成并维护由多个源程序文件、目标文件组合而成的可执行文件



续 表

菜单名称	功 能
Options	设置集成环境的配置和工作方式
Debug	设置调试选项。例如:检查变量和表达式的值、查找函数、在编译代码中插入调试信息等
Break/Watch	设置和删除断点,增加、删除和编辑监视表达式等

1. File 菜单

File 菜单所包含的下拉式子菜单如图 1.2 所示,表 1.2 列出了各菜单选项的功能。

表 1.2 File 菜单

菜单选项	功 能
Load	将文件从磁盘调入内存,若指定的文件名磁盘上不存在,就表示建立一个新文件
Pick	显示最近编辑过的 8 个文件清单,直接单击其中的文件名,便将该文件装入内存
New	编辑一个新文件,默认的文件名为 NONAME. C
Save	将当前正在编辑的文件存盘
Write to	将当前正在编辑的文件以新的文件名存盘
Directory	显示指定的目录及文件清单,若直接按 Enter 键,将显示当前盘、当前路径的目录和文件清单
Change dir	显示当前目录,若键入新目录路径,则改变当前目录路径
OS shell	暂时脱离 Turbo C,返回 DOS 系统;若在 DOS 下键入 Exit 命令又返回 Turbo C
Quit	退出 Turbo C,返回 DOS 系统

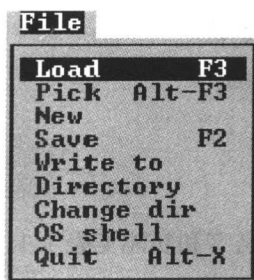


图 1.2 File 菜单

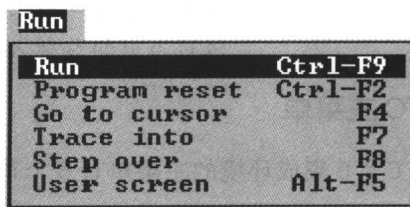


图 1.3 Run 菜单

2. Edit 菜单

单击该菜单名,直接激活程序编辑光标,对当前已调入内存的文件进行全屏幕编辑。

3. Run 菜单

Run 菜单所包含的下拉式子菜单如图 1.3 所示,表 1.3 列出了各菜单选项的功能。



表 1.3 Run 菜单

菜单选项	功 能
Run	运行指定的程序。若执行本命令时还未生成可执行文件,就自动对源程序先编译、链接,再运行
Program reset	终止程序的调试,释放用户程序占用的内存空间,关闭所有已打开的文件
Go to cursor	使程序执行到光标所在行;若光标所在行之前有断点,就执行到断点处;若所给的执行范围内没有可执行代码,将显示警告信息,按 ESC 键返回
Trace into	单步执行程序,遇函数调用自动进入函数跟踪(但函数必须是调试程序可访问的)
Step over	单步执行程序,若遇函数调用跳过
User screen	从 TC 主屏切换到用户屏幕,查看程序的运行结果,按任意键返回 TC 主屏

4. Compile 菜单

Compile 菜单所包含的下拉式子菜单如图 1.4 所示,表 1.4 列出了各菜单选项的功能。

表 1.4 Compile 菜单

菜单选项	功 能
Compile to OBJ	将源文件编译成.obj 文件,并显示目标文件全名
Make EXE file	将源程序生成.exe 文件,并显示可执行文件全名
Link EXE file	将当前的.obj 文件和.LIB 库文件链接成可供运行的文件
Build all	将项目文件中的所有源程序文件重新编译和链接
Primary C file	用于编译含多个.h 头文件的单个源文件,一旦有编译错,则含错文件(.c 文件或.h 文件)将自动装入编辑程序供修改
Get info	显示上次编译或运行程序的所有信息,包括:子目录名、源文件名、文件大小、已编译行数、错误个数、是否使用 EMS 等

Compile

```

Compile to OBJ  C:\NONAME.OBJ
Make EXE file   C:\NONAME.EXE
Link EXE file
Build all
Primary C file:
Get info
  
```

图 1.4 Compile 菜单

Project

```

Project name
Break make on      Errors
Auto dependencies  Off
Clear project
Remove messages
  
```

图 1.5 Project 菜单

5. Project 菜单

Project 菜单主要用于由多个 C 源文件组成的程序。因为一个相当大的 C 程序放在一个文件中编辑是很不方便的,对程序哪怕作再小的一点改动,整个程序都要重新编译。解决的办法是分而治之,将程序分为若干模块,先分块编辑、编译,再链接组装。在 C 中,该组装文件就称为项目“Project”,Project 菜单所包含的下拉式子菜单如图 1.5 所示,表 1.5 列出了各菜单选项的功能。



表 1.5 Project 菜单

菜单选项	功 能
Project name	指定一个扩展名为 .prj 的 project 文件名
Break make on	由用户指定终止编译的缺省条件,通常为:警告时(Warning)、有错误时(Errors)、有致命错误时(Fatal Errors)或链接之前
Auto dependencies	开关命令,为 ON 时,自动检查项目组中的每一个 .obj 文件是否与 .c 文件相一致,若二者的日期或时间不一致,将重新编译 .c 文件
Clear project	去除按 project 文件编译的方式
Remove messages	清除消息窗

6. Options 菜单

Options 菜单所包含的下拉式子菜单如图 1.6 所示,表 1.6 列出了各菜单选项的功能。

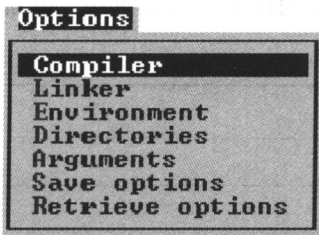


图 1.6 Options 菜单

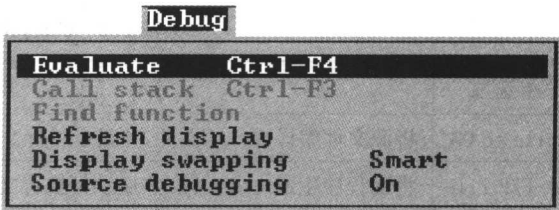


图 1.7 Debug 菜单

表 1.6 Option 菜单

菜单选项	功 能
Compiler	提供设置编译参数的子菜单。包括 Model (存储模式), Defines (宏定义), Code generation (代码生成), optimization (代码优化)等
Linker	提供设置连接参数的子菜单。包括 Map file (选择 map 文件的类型), Initialize segments (段初始化), Default libraries (缺省库)等
Environment	提供对集成环境设置参数的子菜单。包括 Message Tracking (即滚动消息窗口的同时,跟踪编辑窗口中的语法错误), Config auto save (自动保存配置), Backup file (备份文件)等
Directories	设置 TC 搜索路径。包括 include directories (包含文件目录), Library directories (库目录), Output directories (输出目录,即存储 .obj 和 .exe 文件的目录)等
Arguments	设置程序运行时需要命令行参数,通过该命令为运行程序输入命令行参数,使得与 DOS 状态下使用命令行参数一样
Save options	将 option 菜单中所有已作修改的设置保存到配置文件中 (缺省的配置文件为 Tcconfig. tc)。以后启动 TC 时,系统自动寻找 Tcconfig. tc 文件,完成各种新的设置
Retrieve options	加载以前保存过的配置文件



7. Debug 菜单

Debug 菜单所包含的下拉式子菜单如图 1.7 所示,表 1.7 列出了各菜单选项的功能。

表 1.7 Debug 菜单

菜单选项	功 能
Evaluate	计算并显示变量或表达式的值,但表达式中不允许出现函数调用
Call stack	调用堆栈,以显示到目前为止函数的调用情况。Main 函数在栈底,正在运行的函数在栈顶,每项显示调用的函数名和传递给它的参数值
Find function	查找用户程序中的函数,但该程序不能在编译时将 Debug/Source Debugging 和 option/compiler/code generation/obj debug information 设置为 ON
Refresh display	恢复被重写了的屏幕内容
Display swapping	设置显示转换方式。Smart(缺省)时,若执行代码有屏幕输出,就从编辑屏幕切换到用户屏幕,完成输出以后再切换回编辑屏幕;Always 时,执行每一条语句都切换;None 时,任何情况下都不切换
Source debugging	设置源代码的调试方式。ON 时,链接程序能同时用 TC 集成调试程序和独立的 Turbo C 调试程序调试;Standalone 时,就只能用独立的 Turbo C 调试程序调试;None 时两者都不能用

8. Break/watch 菜单

本菜单用于设置断点和添加、删除监视表达式。其下拉式子菜单如图 1.8 所示,表 1.8 列出了各菜单选项的功能。

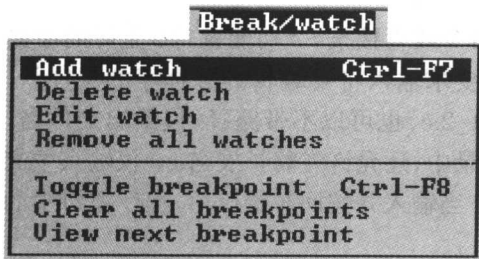


图 1.8 Break/watch 菜单

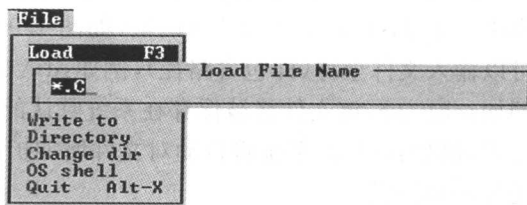


图 1.9 输入文件名对话框

表 1.8 Break/watch 菜单

菜单选项	功 能
Add watch	向监视窗口中输入一个监视表达式,缺省表达式为程序窗口中光标处的单词,若表达式正确,监视窗口就显示表达式的值
Delete watch	删除监视表达式。(只有当监视窗口可见,即程序窗口不能放大时)才能用本命令
Edit watch	编辑监视表达式。执行本命令,调试程序给出一个显示当前监视表达式的拷贝窗口,修改以后,按回车键确认,按 ESC 键取消
Remove all watches	从监视窗口中删除所有的监视表达式
Toggle breakpoint	设置或取消光标处的断点



菜单选项	功 能
Clear all breakpoints	清除程序中的所有断点
View next breakpoint	对断点进行光标定位,即按断点设置的顺序将光标移到下一个断点处

1.2.3 系统状态切换

Turbo C 2.0 不支持鼠标操作,因此光标在菜单和编辑程序之间的切换、菜单的打开与关闭等都必须通过键盘完成,表 1.9 给出了系统状态切换的操作键。

表 1.9 系统状态切换

操 作 键	功 能
F10	光标在菜单和程序窗口之间交替切换
Esc	若某菜单处于打开状态,则关闭该菜单,返回上一级菜单或当前活动窗口;若无菜单打开、而且光标位于菜单行上,则光标跳转到程序窗口

1.2.4 在 Turbo C 2.0 系统上编程

我们对 TC 集成环境已经有了较全面的了解,这里以一个简单程序的编制过程为例,介绍如何在 Turbo C 2.0 系统上完成程序的编辑、编译、链接、运行以及源程序和目标程序的存储等任务。

1. 编辑、编译、运行 C 程序的操作步骤

(1) 装载编辑程序。

假设 TC 已经启动,打开 File 菜单,选择 Load 命令或直接按 F3 功能键,TC 主屏的中央弹出一个“Load File Name”对话框(如图 1.9 所示),要求输入将要编辑的源程序的文件名。可以输入文件全称(即包含路径),例如:d:\test\chl_2.c;也可以不带路径,仅给出文件名,例如:chl_2.c,该文件将被保存在系统默认的输出目录中;还允许省略扩展名.c,仅输入主文件名,例如:chl_2,系统将自动对文件增加扩展名.c。当输入并按回车键以后,程序窗口中的光标即被激活。

(2) 编辑源程序。

如果在上面的“Load File Name”对话框中输入的文件名磁盘上已存在,该文件即被装入编辑窗口,这表示将一个已经存盘的老文件从磁盘调入内存,你就可以对它进行修改与编辑了;若输入的文件名磁盘上不存在(即新的文件),则表示创建一个新文件,此时,编辑窗口如同一张白纸,你可以将事先已编好的源程序输入到该窗口内。假设本例为创建新程序,现输入以下程序内容:

```
main( )
{ char a,b;
  printf("Please input two characters!");
  scanf("%c%c",&a, &b);
  printf("The first character is %c\n",a);
```



```
printf("The second character is %c\n",b);
}
```

注意:在输入过程中若发现有错,可使用光标移动键(↓ ↑ → ←)将编辑光标移动到出错处,此时若按 Del 键就删除了光标处的字符,若按 Backspace 键则删除光标左边的字符;若将当前状态为“Insert”状态(即“插入”状态),则键入的字符就插入在光标处,若为“改写”状态,则输入的字符将替换光标处的字符(插入状态与改写状态的切换是通过按“Insert”键完成的)。常用的编辑操作命令请参见表 1.10。

表 1.10 Turbo C 2.0 常用编辑命令一览表

类 别	编 辑 键	功 能
光 标 移 动	← 键	光标左移一格
	→ 键	光标右移一格
	↑ 键	光标上移一行
	↓ 键	光标下移一行
	Home	快速移动光标至行首
	End	快速移动光标至行末
	Ctrl-Home	快速移动光标至窗口左上角
	Ctrl-End	快速移动光标至窗口右下角
	Ctrl-PgUp	快速移动光标至文件首
	Ctrl-PgDn	快速移动光标至文件末
插 入 与 删 除	Ctrl-QP	快速移动光标至上次位置
	Ins (Ctrl-V)	插入/替换状态切换
	Ctrl-N	在光标处插入一行
	Backspace(Ctrl-H)	删除光标左边的字符
	Del (Ctrl-G)	删除光标处的字符
	Ctrl-T	删除光标右边的字符
块 操 作	Ctrl-Y	删除光标所在行
	Ctrl-KB	将当前光标所在处标记为块首
	Ctrl-KK	将当前光标所在处标记为块尾
	Ctrl-KC	将已经标记的块在当前光标处复制
	Ctrl-KY	将已经标记的块删除
	Ctrl-KV	将已经标记的块移动到当前光标处
	Ctrl-KH	取消已标记的块
	Ctrl-KR	从磁盘上读块文件
	Ctrl-KW	将已经标记的块往磁盘上写



(3) 编译源程序。

完成程序的输入和编辑以后,就应该对源程序进行编译了。方法是:打开 Compile 菜单,选择 Compile 命令或直接按 ALT-F9 热键。编译结束,屏幕中央弹出一个显示有关编译信息的对话框,如果程序没有错误,对话框底部会显示:“Success: Press any key”字样,意思是编译成功(如图 1.10 所示),按任意键即可关闭该对话框,返回编辑窗口;如果程序有错误,对话框底部就会显示:“Warning: Press any key”字样(这表示有警告错误)或“Error: Press any key”字样(这表示有语法错误),有语法错误是不能生成.obj 文件的,当按任意键后,光标跳入消息窗口的第一条出错信息行上(见图 1.11),此时按 F6 功能键,将光标切换到程序窗口的出错行上,修改之,待所有的错误都修改完毕,再重新编译,如此反复直至编译成功。

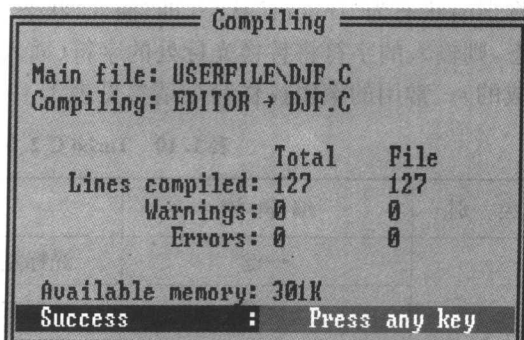


图 1.10 编译成功对话框

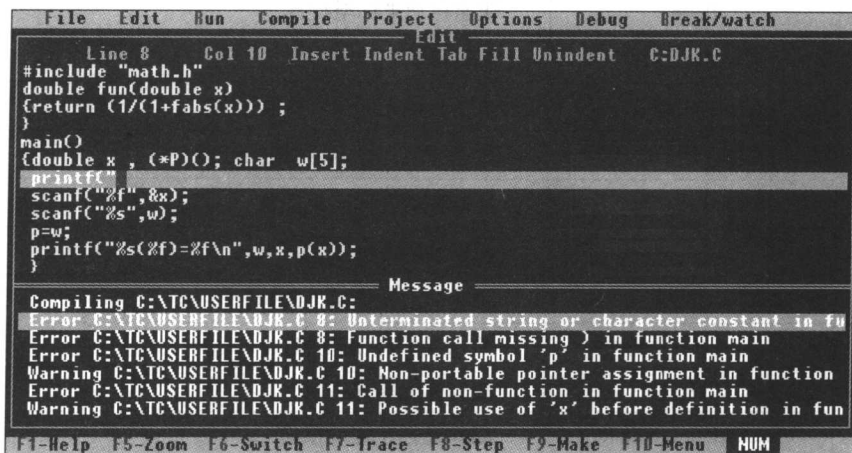


图 1.11 编译报错对话框

(4) 运行目标程序。

编译通过以后,就可以运行目标程序了。方法是:打开 Run 菜单,从中选择 Run 命令或直接按 Ctrl-F9 热键,这时你的程序将被链接并运行。现假设就运行上面输入的那个简单程序,当按下 Ctrl-F9 以后,屏幕立即由程序窗口切换到了用户窗口,并显示提示语:Please input two characters !,这是程序的第一个 printf 语句输出的提示内容,提示你输入两个字符,当输入了两个字符并按回车键以后,程序很快地被运行结束,显示运行结果的用户窗口被快速显示一下以后,便自动返回到程序窗口。图

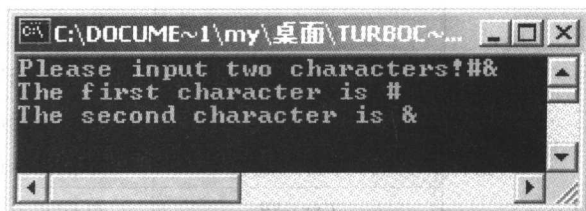


图 1.12 运行 C 程序的用户窗口