

高等学校
计算机教材

面向应用与实践系列

郭有强 等编著

C语言程序设计 实验指导与课程设计

清华大学出版社





郭有强 等编著

C语言程序设计 实验指导与课程设计

清华大学出版社
北京

内 容 简 介

本书作为《C 语言程序设计》一书的配套教材,注重培养实际编程能力,具有很强的通用性和实用性。

本书介绍了两种编译环境,给出了 12 个基础实验、7 个课程设计,基本覆盖了 C 语言程序设计的主要知识点。给出了 4 套模拟试题及相关解答以及主教材中的习题解答和实验教材中思考与练习的参考答案。附录部分介绍了常见的编译错误信息,以帮助读者上机练习。书中通过精心设计的编程内容,使读者体会、消化、掌握和应用相关知识与技术,尽量轻松入门,迅速提高。

本书可作为高等院校相关专业以及其他 C 语言课程的实验教材,也可作为自学 C 语言程序设计的实验参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C 语言程序设计实验指导与课程设计/郭有强等编著. 北京:清华大学出版社,2009.2
(高等学校计算机教材——面向应用与实践系列)

ISBN 978-7-302-19272-5

I. C… II. 郭… III. C 语言—程序设计—高等学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 004587 号

责任编辑:袁勤勇 李玮琪

责任校对:时翠兰

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市清华园胶印厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185×260

印 张:10.5

字 数:245 千字

版 次:2009 年 2 月第 1 版

印 次:2009 年 2 月第 1 次印刷

印 数:1~4000

定 价:16.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:010-62770177 转 3103 产品编号:031241-01

前 言

《C 语言程序设计》是一门实践性很强的课程,上机实验和课程设计是本课程不可缺少的实践环节。上机实验的目的是使读者熟悉 C 语言的基本语法规则、程序结构、编程思路及解题的全过程,加深对 C 语言的理解,得到程序设计基本方法和技巧的训练,从而巩固和深化所学的知识,真正能用 C 语言这个强有力的编程工具去解决实际问题。同时,通过上机实验及其随后的课程设计,读者能够熟练掌握 C 集成开发环境的使用。

本书是集众多长期从事 C 语言教学工作的一线教师的经验和体会,并参考大量的国内外有关资料编写而成。全书共分为 5 部分,另有附录 A 至附录 D。第 1 部分介绍了两种编译环境;第 2 部分给出了 12 个基础实验,基本覆盖了 C 语言程序设计的主要知识点,每个实验都包含“实验目的”、“实验准备”、“实验内容”和“思考与练习”;第 3 部分给出了 1 个课程设计样例和 6 个课程设计题目(供参考使用),通过综合训练,期望读者能够对 C 语言编程思想的进一步理解,进而提升实际应用编程能力;第 4 部分给出了 4 套模拟试题;第 5 部分给出了主教材中的习题解答、模拟试题的参考答案和基础实验部分“思考与练习”的参考答案;附录部分介绍了常见的编译错误信息和标准 ASCII 码表,以帮助读者上机练习;最后还给出了实验报告参考样本和课程设计报告参考样本,以规范文档格式。

本书内容丰富,结构紧凑,选题典型丰富,对初学者具有很强的针对性。

本书是《C 语言程序设计》的配套实验及课程设计指导教材。建有课程网站(<http://computer.bbxy.edu.cn/czdkc/zhongdian.htm>),供参考。

本书由郭有强担任主编,总体设计并统稿。戚晓明、周会萍、何爱华、姚保峰、王磊、谢娜和刘娟在内容编写、程序测试和文字校对工作中付出了辛勤劳动,在此深表感谢。

感谢读者选择使用本书,对本书内容存在的问题,敬请读者批评并提出修改建议,我们将不胜感激。在使用本书时如遇到什么问题需要与作者商榷,或想索取其他相关资料,请与作者联系。

电子邮件地址:bbxyguo@163.com。

郭有强
2008 年 12 月

目 录

第 1 部分 预备知识	1
1.1 Turbo C 2.0 的基本用法	1
1.2 Visual C++ 6.0 的基本用法	9
第 2 部分 基础实验	12
实验 1 C 程序设计基础(1)	12
实验 2 C 程序设计基础(2)	14
实验 3 程序流程控制	17
实验 4 数组(1)	21
实验 5 数组(2)	26
实验 6 模块设计	30
实验 7 指针(1)	34
实验 8 指针(2)	36
实验 9 结构体与共用体类型(1)	39
实验 10 结构体与共用体类型(2)	42
实验 11 位运算及文件操作	47
实验 12 Visual C++ 6.0 集成开发环境和简单 C++ 程序	54
第 3 部分 课程设计	60
3.1 概述.....	60
3.2 总体要求.....	60
3.3 课程设计样例——学生成绩管理系统.....	65
3.4 课程设计题目.....	77
第 4 部分 模拟试题	80
4.1 模拟试题(一).....	80
4.2 模拟试题(二).....	84
4.3 模拟试题(三).....	88
4.4 模拟试题(四).....	92
第 5 部分 参考答案	97
5.1 主教材习题参考答案.....	97
5.2 基础实验部分【思考与练习】参考答案	126

5.3 模拟试题参考答案	142
附录	149
附录 A Turbo C 2.0 编译错误信息	149
附录 B 标准 ASCII 码表	155
附录 C 实验报告参考样本	156
附录 D 课程设计报告参考样本	157
参考文献	159

第 1 部分 预备知识

1.1 Turbo C 2.0 的基本用法

1.1.1 Turbo C 简介

Turbo C 是美国 Borland 公司的产品, Borland 公司是一家专门从事软件开发、研制的大公司。该公司相继推出了一套 Turbo 系列软件, 如 Turbo BASIC, Turbo Pascal, Turbo Prolog, 这些软件很受用户欢迎。该公司在 1987 年首次推出 Turbo C 1.0 产品, 其中使用了全然一新的集成开发环境, 即使用了一系列下拉式菜单, 将文本编辑、程序编译、连接以及程序运行一体化, 大大方便了程序的开发。1988 年, Borland 公司又推出 Turbo C 1.5 版本, 增加了图形库和文本窗口函数库等, 而 Turbo C 2.0 则是该公司 1989 年出版的。Turbo C 2.0 在原来集成开发环境的基础上增加了查错功能, 并可以在 Tiny 模式下直接生成 .com(数据、代码、堆栈处在同一 64KB 内存中)文件。Borland 公司后来又推出了面向对象的程序软件包 Turbo C++, 它继承发展了 Turbo C 2.0 的集成开发环境, 并包含了面向对象的基本思想和设计方法。

1.1.2 Turbo C 2.0 集成开发环境

进入 Turbo C 2.0 集成开发环境后, 屏幕上显示如图 1.1 所示。



图 1.1 Turbo C 2.0 集成开发环境

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。这四个部分构成了 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:

File Edit Run Compile Project Options Debug Break/watch

除 Edit 外,其他各项均有子菜单。只要按下 Alt 键同时按下菜单项的第一个字母,就可进入该项的子菜单中。也可以按下 F10 键转到主菜单,再将光标移到要选择的菜单项后按 Enter 键。

1. File 菜单

按 Alt+F 键可进入 File 菜单:

File 菜单的子菜单共有 9 项,分别叙述如下。

(1) Load: 装载一个文件,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。该项的快捷键为 F3,即只要按 F3 键即可进入该项,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。其快捷键为 Alt+F3。

(3) New: 新建文件,默认文件名为 NONAME.C,存盘时更改文件名。

(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,其快捷键为 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,其快捷键为 Alt+X。

说明:

以上各项可用光标键移动进行选择,回车后执行。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。

(1) 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。

(2) 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:

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 '

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,该菜单有以下各项。

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,其快捷键为 Ctrl+F9。

(1) Program reset: 中止当前的调试,释放分给程序的空间,其快捷键为 Ctrl+F2。

(2) 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。其快捷键为 F4。

(3) 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,其快捷键为 F7。

(4) 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,其快捷键为 F8。

(5) User screen: 显示程序运行时在屏幕上显示的结果。其快捷键为 Alt+F5。

4. Compile 菜单

按 Alt+C 键可进入 Compile 菜单,该菜单有以下几个选项。

Compile to OBJ: 将一个 C 源文件编译生成 .obj 目标文件。其快捷键为 Alt+F9。

(1) Make EXE file: 生成一个 .exe 的文件。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,则是由 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。

(2) Link EXE file: 把当前 .obj 文件及库文件连接在一起生成 .exe 文件。

(3) 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)。

(4) 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。

(5) 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,该菜单包括以下选项。

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 executable 文件,可以先建立一个名为 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 C 语言源文件对待,另外,其中的文件也

可以是库文件,但必须写上扩展名.lib。

(1) Break make on: 由用户选择是否在有 Warning、Errors、Fatal Errors 时或 Link 之前退出 Make 编译。

(2) 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。

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

(4) 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,该菜单有几个内容。

(1) Options: 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。这些子菜单如下。

① Model: 共有 Tiny、small、medium、compact、large 和 huge 六种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。多重定义可用分号,赋值可用等号。

③ 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 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。

Alignonent 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: 它又有许多任选项。

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: 它又有许多任选项。

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

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 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,它有以下内容:

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 Default libraries 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。

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 warning 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、小写字母。

(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,它有以下内容:

① Message tracking

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,其快捷键为 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,有下列各项:

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。

(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, 默认的配置文件为 TCCONFIG. TC。

(7) 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 菜单

按 Alt+D 键可选择 Debug 菜单, 该菜单主要用于查错, 它包括以下内容:

(1) Evaluate

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, 该菜单有以下内容:

(1) Add watch: 向监视窗口插入一个监视表达式。

(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。

1.1.3 Turbo C 2.0 的配置文件

所谓配置文件是包含 Turbo C 2.0 有关信息的文件, 其中存有编译、连接的选择和路径信息。可以用下述方法建立 Turbo C 2.0 的配置。

(1) 建立用户自命名的配置文件: 可以从 Options 菜单中选择 Options/Save options 命令, 将当前集成开发环境的所有配置存入一个由用户命名的配置文件中。下次启动 TC 时只要在 DOS 下键入:

```
tc/c<用户命名的配置文件名>
```

就会按这个配置文件中的内容作为 Turbo C 2.0 的选择。

(2) 若设置 Options/Environment/Config auto save 为 on, 则退出集成开发环境时, 当前的设置会自动存放到 Turbo C 2.0 配置文件 TCCONFIG. TC 中。Turbo C 在启动时会自动寻找这个配置文件。

(3) 用 TCINST 设置 Turbo C 的有关配置,并将结果存入 TC.EXE 中。Turbo C 在启动时,若没有找到配置文件,则取 TC.EXE 中的默认值。

1.1.4 Turbo C 2.0 的快速参考行

F1—HELP 打开 HELP 窗口,提供有关的编辑命令信息。

F5—ZOOM 将信息窗口扩展为全屏幕或恢复原状。

F6—SWITCH 切换编辑窗口、信息窗口和观察窗口为活动窗口。

F7—TRACE 在调试模式下,一次执行一行用户程序,进入所调用的函数。

F8—STEP 在调试模式下,一次执行一行用户程序,不进入所调用的函数。

F9—MAKE 编译连接生成可执行文件。

F10—MENU 活动窗口与菜单切换。

1. File 菜单

Load 装入一个新文件(F3)。

Pick 出现一个选择表,由 NONAME.C---Load file---和前几次装入的几个不同的文件构成。可选择一项装入。

New 建立一个新文件,文件名暂定为 NONAME.C。

Save 把编辑窗口的文件存盘(F2)。

Write to 选择此项,出现一个空白提示框,等待输入用于存盘的文件名。

Directory 选择此项,出现一个提示框: Enter File name *. *,可键入预装入编辑窗口的文件名。

Change dir 选择此项,先显示当前目录,键入不同的驱动器和目录

Os shell 选择此项,可暂时退出 Turbo C。

Quit 选择此项,退出 Turbo C。

2. Edit 选择项

选择此项进入编辑窗口,对调入其中的文件进行全屏幕编辑,也可直接编辑一个文件。

3. Run 菜单

Run 运行用户程序(Ctrl+F9)。

Porgram reset 中止程序的调试(Ctrl+F2)。

Go to cursor 用于程序调试(F4)。

Trace into 执行当前函数的下一条语句(F7)。

Step over 执行当前函数的下一条语句。不能跟踪进入低一级的调用函数(F8)。

User screen 将 Turbo C 2.0 集成开发环境的主屏幕转换到用户屏幕。

4. Compile 菜单

Compile to OBJ 将程序编译成目标文件(.obj),显示所生成的目标文件名(Alt+F9)。

Make EXE file 调用 Project_Make,把 Turbo C 的源程序直接生成一个可执行文件,并显示生成的文件名(F9)。

Link EXE file 把当前的.obj和.lib文件连接成可执行文件。

Build all 把所有的工程文件中的.obj文件的日期和时间设置成零,再重新编译和连接这些文件。

Primary C file 用来编译多个标题文件的单个源文件。

Get info 可以将上次编译或运行的编译信息显示在一个新开辟的窗口。

5. Project 菜单

Project name 工程名。

Break make on 中止生成。

Auto dependencies 自动依赖关系。

Clear project 清除工程名。

Remove messages 清除错误信息。

6. Options 菜单

用来设置集成开发环境的工作方式。

Compiler 编译程序。

Linker 连接程序。

Environment 环境设置。

Directories 目录设置。

Arguments 传递参数设置。

Save options 保存此设置。

Retrieve options 恢复原设置。

7. Debug 菜单

程序的动态调试

8. Break/watch 菜单

1.2 Visual C++ 6.0 的基本用法

1. 启动 Visual C++ 6.0 环境

方法: 选择【开始】|【程序】|【Microsoft Visual Studio 6.0】|【Microsoft Visual C++

6.0】命令,启动 Visual C++ 6.0,主窗口如图 1.2 所示。

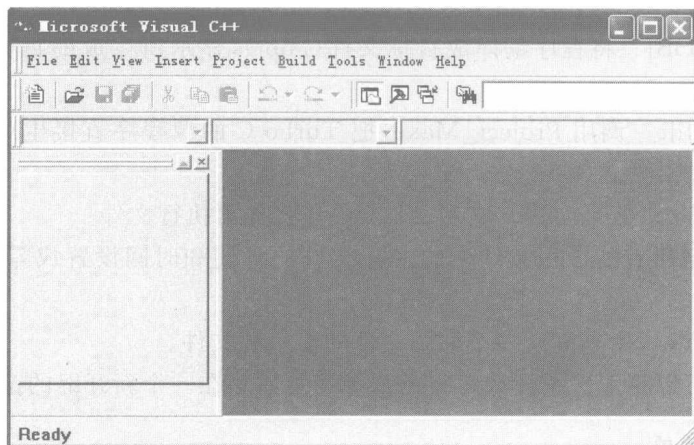


图 1.2 Visual C++ 6.0 环境

2. 编辑源程序文件

1) 创建新项目

(1) 选择 File|New 菜单项,弹出 New 对话框。

(2) 选择 Projects 选项卡,单击 Win32 Console Application 选项,在 Projects name 文本框中输入项目名,如“gcl”,在 Location 文本框中输入或选择新项目所在位置,单击 OK 按钮。弹出 Win32 Console Application-Step 1 of 1 对话框。

(3) 选择 An empty project,单击 Finish 按钮。系统显示 New Project Information 对话框,单击 OK 按钮。

2) 创建新项目中的文件

(1) 选择 File|New 菜单项,弹出 New 对话框。

(2) 选择 Files 选项卡。单击 C++ Source File 选项,在文本框中输入文件名,单击 OK 按钮。系统自动返回 VC++ 6.0 主窗口。

(3) 在文件编辑区窗口输入源程序文件,如图 1.3 所示。

3. 编译和连接

方法 1: 选择 Build | Compile e1_1.cpp 菜单命令。

方法 2: 单击工具栏上的 Build 按钮进行编译和连接。

(1) 系统对程序文件进行编译和连接,生成以项目名称为主文件名的可执行文件。

(2) 编译和连接过程中,系统如发现程序有语法错误,则在输出区窗口中显示错误信息,给出错误的性质、出现位置和错误的原因。如果双击某条错误,编辑区窗口右侧出现一个箭头,指示出现错误的程序行。用户据此对源程序进行相应的修改,并重新编译和连接,直到通过为止。

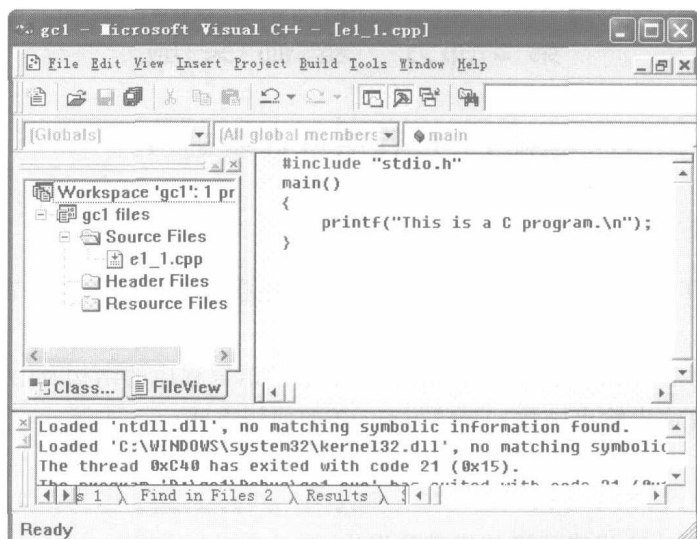


图 1.3 C++ 源程序编辑环境

4. 执行

方法 1: 选择 Build | Execute e1_1.exe 菜单命令。

方法 2: 单击工具栏上的 Execute Program 按钮, 执行编译连接后的程序。

运行成功, 屏幕上输出执行结果, 并提示信息: “Press any key to continue”。此时按任意键系统都将返回 VC++ 6.0 主窗口。如果在执行程序过程中出现运行错误, 则要修改源程序文件并且重新编译、连接和执行。