



普通高等教育“十二五”重点规划教材 公共课系列
中国科学院教材建设专家委员会“十二五”规划教材

C语言程序设计

实践教程

韩志明 王立君◎主编



科学出版社

普通高等教育“十二五”重点规划教材 公共课系列
中国科学院教材建设专家委员会“十二五”规划教材

C 语言程序设计实践教程

韩志明 王立君 主 编
韩旭明 李 楠 副主编
朱丽莉 主 审

科 学 出 版 社

北 京

内 容 简 介

本书是《C 语言程序设计》(张淑华主编, 科学出版社)的配套教材。全书共包括三个部分: 第一部分为 C 语言程序设计上机实验内容, 介绍了 Turbo C 2.0 和 VC++ 6.0 两种集成开发环境的使用方法; 第二部分为《C 语言程序设计》综合练习题及解答; 第三部分为全国计算机等级考试(二级 C 语言)真题及答案汇编, 内容覆盖全部 C 语言程序设计课程。

本书具有较强的实用性和可操作性。本书适合作为高等院校本科生学习“C 语言程序设计”课程的配套教材, 也可作为高职高专学生、高等院校成人教育的教材或参考书。

图书在版编目(CIP)数据

C 语言程序设计实践教程/韩志明, 王立君主编. —北京: 科学出版社, 2012

(普通高等教育“十二五”重点规划教材 公共课系列·中国科学院教材建设专家委员会“十二五”规划教材)

ISBN 978-7-03-033063-5

I. ①C… II. ①韩… ②王… III. ①C 语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 265604 号

责任编辑: 戴 薇 李 瑜 张振华 / 责任校对: 刘玉靖
责任印制: 吕春珉 / 封面设计: 东方人华平面设计部

科 学 出 版 社 出 版

北京东黄城根北街 16 号
邮政编码: 100717

<http://www.sciencep.com>

骏 杰 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2012 年 1 月第 一 版 开本: 787×1092 1/16

2012 年 3 月第二次印刷 印张: 17 1/2

字数: 406 000

定价: 27.00 元

(如有印装质量问题, 我社负责调换<骏杰>)

销售部电话 010-62140850 编辑部电话 010-62135763-2038

版权所有, 侵权必究

举报电话: 010-64030229; 010-64034315; 13501151303

前 言

本书为配合“C 语言程序设计”课程的学习而编写，其内容紧扣“C 语言程序设计”的教学内容与教学进度，结合作者多年的程序设计课程教学经历和全国计算机等级考试辅导的经验，以全面提升学生的应用能力为主要目标，以方便自学为立足点，经过精心策划和准确定位，有助于学习者全面系统地掌握 C 语言的知识 and 提高解决实际问题的能力。

全书共分三部分：

第一部分为 C 语言程序设计上机实验，详细介绍了 Turbo C 2.0 和 VC++ 6.0 两种集成开发环境下的基本操作，有助于提高学生的程序调试能力。本书紧扣《C 语言程序设计》内容，共有 17 个上机实验，每个实验都给出了典型实例，介绍程序的设计思想并给出代码、程序的分析解释和运行结果，使学生对问题有所了解。上机编程是在上机练习的基础上，给出了几个典型的编程题目，要求学生独立完成设计、调试和运行，是对上机练习部分的有效扩展和延伸。同时，通过综合训练题目的上机练习，提高学生对所学知识的综合应用能力和程序的调试能力。

第二部分为 C 语言程序设计练习题及解答，对《C 语言程序设计》所讲述的内容进行了归类，分为 6 章，每章均包括选择、写出程序运行结果和填空三类题型，并配有习题答案。通过对这些习题的练习，读者可以系统地掌握《C 语言程序设计》教材中每章所讲述的知识点。

第三部分为近年全国计算机等级考试（二级 C 语言）笔试真题及答案。

本书中的实例经过反复挑选，既有实用性，又不乏趣味性，使学生在提高兴趣的同时，掌握相关知识。书中全部例题和习题都经上机调试通过。本书遵循由浅入深、循序渐进的原则，适合作为高等院校“C 语言程序设计”课程的实验教材，也适合作为广大计算机爱好者的自学读物。

本书由韩志明、王立君担任主编，由韩旭明、李楠担任副主编。具体编写分工：韩志明编写第一部分和第二部分的第一章、第二章及第三部分，王立君编写第二部分的第三章，韩旭明编写第二部分的第四章和第六章，李楠编写第二部分的第五章，全书由韩志明统稿。

由于时间紧迫，编者水平有限，书中难免有不足之处，恳请读者提出宝贵意见和建议。

编 者

2011 年 10 月

目 录

第一部分 C 语言程序设计上机实验

实验一 Turbo C 2.0 和 Visual C++ 6.0 的基本操作.....	2
思考题一.....	13
实验二 Visual C++ 6.0 动态调试程序的方法.....	15
思考题二.....	19
实验三 数据类型及输入/输出函数.....	20
思考题三.....	26
实验四 顺序结构程序设计.....	27
思考题四.....	28
实验五 选择结构程序设计.....	29
思考题五.....	32
实验六 循环结构程序设计.....	33
思考题六.....	37
实验七 数组的使用.....	38
思考题七.....	42
实验八 字符数组.....	43
思考题八.....	46
实验九 结构体、共用体与枚举类型.....	47
思考题九.....	50
实验十 函数的定义.....	52
思考题十.....	56
实验十一 数组作为函数的参数.....	57
思考题十一.....	61

实验十二 变量的作用域	62
思考题十二	65
实验十三 指针	66
思考题十三	69
实验十四 指针与数组	70
思考题十四	78
实验十五 指针与函数	79
思考题十五	82
实验十六 指针与结构体	83
思考题十六	89
实验十七 文件	91
思考题十七	97

第二部分 C 语言程序设计练习题及解答

第一章 C 语言基础知识	100
第二章 C 语言控制结构	118
第三章 数组	144
第四章 函数与预处理	158
第五章 指针、结构体与共用体	189
第六章 文件操作和位运算	226

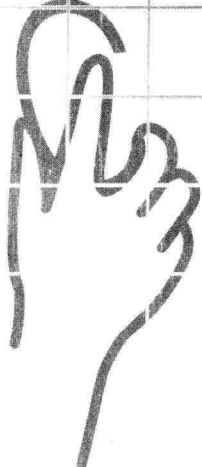
第三部分 历年全国计算机等级考试（二级 C 语言）笔试真题及答案汇编

2010 年 3 月全国计算机等级考试二级笔试试卷	242
2010 年 9 月全国计算机等级考试二级笔试试卷	254
2011 年 3 月全国计算机等级考试二级笔试试卷	264

A large, stylized outline of the letter 'C' is positioned to the left of the title. A thick horizontal line passes through the middle of the 'C'.

第一部分

C 语言程序设计上机实验



实验一

Turbo C 2.0 和 Visual C++ 6.0 的基本操作

一、实验目的

- 1) 掌握 Turbo C 2.0 的主要功能及软件环境的设置等。
- 2) 通过一个简单的实例，使读者了解 C 程序从编写到运行的全过程。
- 3) 掌握 Visual C++ 6.0 的主要功能及软件使用方法。

二、实验内容

1. Turbo C 2.0 的集成开发环境

进入 Turbo C 2.0 系统需要调用 tc.exe 文件，可以在 DOS 平台的 tc 子目录下键入 tc 并回车；也可以在 Windows 平台下打开 tc 文件夹，双击 tc.exe 应用程序图标。

(1) Turbo C 2.0 的工作窗口

打开 Turbo C 2.0 后，会显示 Turbo C 2.0 的版本信息框，用户只需按任意键，此信息框就会被关闭。启动后的 Turbo C 2.0 工作窗口如图 1-1 所示。



图 1-1 Turbo C 2.0 的工作窗口

Turbo C 2.0 窗口中包括如下内容。

① 菜单栏：包括 File（文件）、Edit（编辑）、Run（运行）、Compile（编译）、Project（项目）、Options（选项）、Debug（调试）、Break/watch（断点/监视）主菜单。除 Edit 项外，每一个主菜单还有相应的子菜单，只要在按住 Alt 键的同时再按某项中第一个字母，就可进入该项的子菜单中，通过菜单实现相应的操作。

② 编辑区：正上方有 Edit 字符作为标志，编辑窗口的作用是对 Turbo C 源代码进行输入和编辑。该窗口的上部有一行说明性标志，如 Line 1 和 Col 1，它们表示当前光标的位置。在该行的最右边显示当前正在编辑的源程序的文件名（默认的文件名为 NONAME.C）。

③ 信息区：用 Message 字符标记以下的区域信息。

④ 功能键提示行：提示一些功能键（快捷键）的作用，主要包括如下内容。

- F1-Help: 显示帮助信息。
- F5-Zoom: 分区控制。将当前活动窗口（编辑窗口或信息窗口）全屏显示。
- F6-Switch: 转换活动窗口。交替转换编辑窗口和信息窗口为当前活动窗口，当某个窗口为激活状态时，对应的标志字符（Edit 或 Message）将高亮显示。
- F7-Trace: 跟踪命令。用于跟踪程序的运行情况。
- F8-Step: 按步执行。每按一次 F8 键将仅执行一条语句。
- F9-Make: 进行编译和连接。生成.obj 文件和.exe 文件，但不运行程序。
- F10-Menu: 返回到主菜单。

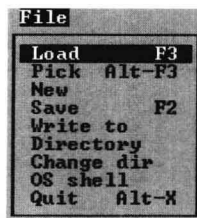


图 1-2 File 菜单

(2) Turbo C 2.0 主要菜单的功能

① File（文件）菜单，如图 1-2 所示。其中各项含义如下。

- Load（加载）：装入一个文件，可用类似于 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。
- Pick（选择）：将最近装入编辑窗口的 8 个文件列成表供用户选择，选择后将程序装入编辑区，并将光标置在上次修改过的位置。其热键为 Alt+F3。
- New（新文件）：说明文件是新的，默认文件名为 NONAME.C，存盘时可改名。
- Save（存盘）：将编辑区中的文件存盘，若文件名为 NONAME.C 时，将询问是否改名。其热键为 F2。
- Write to（另存为）：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。
- Directory（目录）：显示目录及目录中的文件，并可由用户选择。
- Change dir（改变目录）：显示当前目录，用户可以改变显示的目录。
- OS shell（暂时退出）：暂时退到 DOS 操作系统下，此时可以运行 DOS 命令；若想再回到 Turbo C 中，只要在 DOS 状态下输入 EXIT 即可。
- Quit（退出）：退出 Turbo C 2.0，返回到 DOS 操作系统中。其热键为 Alt+X。

说明：Turbo C 2.0 所有菜单均可通过移动方向键进行选择，按回车键则执行，也可用每一项的第一个大写字母直接选择。要退到主菜单或从下一级菜单退回，均可用 Esc 键。

② Edit（编辑）菜单，进行文本编辑（用 F1 键可获得有关编辑方法的帮助信息）。编辑命令如表 1-1 所示。

表 1-1 C 语言中的编辑命令

编辑命令	含 义	编辑命令	含 义
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	删除光标处的一个词
Ctrl+KB	设置块开始	Ctrl+KK	设置块结尾

续表

编辑命令	含 义	编辑命令	含 义
Ctrl+KH	块取消	Ctrl+KC	块复制
Ctrl+KV	块移动	Ctrl+KR	读文件
Ctrl+KY	块删除	Ctrl+KP	块文件打印
Ctrl+KW	存文件	Ctrl+Q[查找双界符的后匹配符
Ctrl+Q]	查找双界符的前匹配符	Ctrl+F1	如光标所在处为 Turbo C 2.0 库函数, 则获得有关该函数的帮助信息

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 为自动缩进的控制键。

③ Run (运行) 菜单, 如图 1-3 所示。其中各项的含义如下。

- Run (运行程序): 运行由 Project/Project name 项指定的文件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); 否则在进行编译、连接后才运行。其热键为 Ctrl+F9。

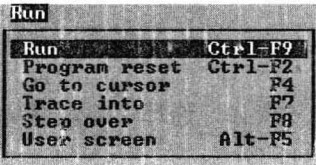


图 1-3 Run 菜单

- Program reset (程序重启): 中止当前的调试, 释放分给程序的空间。其热键为 Ctrl+F2。
- Go to cursor (运行到光标处): 在调试程序时, 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。
- Trace into (跟踪进入): 在调用子函数时, 若用该项, 将跟踪到该子函数内部去执行。其热键为 F7。
- Step over (单步执行): 执行当前语句, 即使当前是函数调用, 也不会跟踪进入函数的内部, 而是将其作为一条普通语句来执行。其热键为 F8。
- User screen (用户屏幕): 程序运行时, 在屏幕上显示的结果。其热键为 Alt+F5。

④ Compile (编译) 菜单, 如图 1-4 所示。其中各项的含义如下。

- Compile to OBJ (编译生成目标码): 将一个 C 源文件编译生成 .obj 目标文件。其热键为 Alt+F9。
- Make EXE file (生成执行文件): 此命令生成一个 .exe 的文件。其中 .exe 文件名是下面几项之一。
 - ◆ 由 Project/Project name 说明的项目文件名。
 - ◆ 若没有项目文件名, 则为由 Primary C file 说明的源文件。
 - ◆ 若以上两项都没有文件名, 则为当前窗口的文件名。

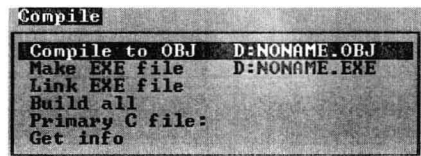


图 1-4 Compile 菜单

- **Link EXE file** (连接生成执行文件): 把当前.obj 文件及库文件连接在一起生成.exe 文件。
- **Build all** (建立所有文件): 重新编译项目中的所有文件, 并进行装配生成.exe 文件。该命令不做过时检查(上面的几条命令要做过时检查, 如果目前项目中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)。
- **Primary C file**: 当没有定义工程文件时, 使用此选项来指定哪一个文件将被编译成.obj 文件或生成.exe 文件。
- **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。

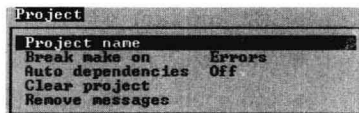


图 1-5 Project 菜单

⑤ **Project** (项目) 菜单, 如图 1-5 所示。其中各项的含义如下。

- **Project name** (项目名): 项目是扩展名为.prj 的文件, 其中包括将要被编译和连接文件的文件名。例如, 一个程序由 file1.c 和 file2.c 组成, 要将这两个文件编译装配成一个 file.exe 的执行文件, 可以先建立一个名为 file.prj 的项目文件, 其内容为: file1.c、file2.c。此时将 file.prj 放入 Project name 项中, 以后进行编译时系统将自动对项目文件中规定的两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, 将不对 file1.c 进行编译, 而直接连接。

! 注意: 当项目文件中的文件无扩展名时, 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。

- **Break make on** (中止编译): 由用户选择是否在有 Warning (警告)、Errors (错误)、Fatal Errors (致命错误) 时或 Link (连接) 之前退出 Make 编译。
- **Auto dependencies** (自动依赖): 开关为 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。
- **Clear project** (清除项目文件): 清除 Project / Project name 中的项目文件名。
- **Remove messages** (删除信息): 把错误信息从信息窗口中清除掉。

⑥ **Options** (运行环境设置菜单), 如图 1-6 所示。对初学者来说, 要谨慎使用该菜单。其中各项的含义如下。

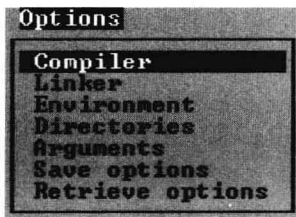


图 1-6 Options 菜单

- **Compiler** (编译器): 该项中又有许多子菜单, 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, 有下列各项。
- ◆ **Model**: 共有 tiny, small, medium、compact、large 和 huge 6 种不同模式可由用户选择。
- ◆ **Calling convention**: 可选择 C 或 Pascal 方式传递参数。
- ◆ **Floating point**: 可选择仿真浮点、数字协处理器浮点或无浮点运算。
- ◆ **Identifier length**: 说明标识符有效字符的个数, 默认为 32 个。

- ◆ 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。
 - Linker (连接器): 本菜单设置有关连接的选择项。
 - Environment (环境): 本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, 有下列各项。
 - ◆ Edit auto save: 是否在 Run 或 Shell 之前, 自动存储编辑的源文件。
 - ◆ Backup file: 是否在源文件存盘时产生后备文件 (.bak 文件)。
 - ◆ Tab size: 设置制表键大小, 默认为 8。
 - ◆ Zoomed windows: 将现行活动窗口放大到整个屏幕, 其热键为 F5。
 - Directories (路径): 规定编译、连接所需文件的路径, 有下列各项。
 - ◆ Include directories: 包含文件的路径, 多个子目录用“;”分开。
 - ◆ Library directories: 库文件路径, 多个子目录用“;”分开。
 - ◆ Output directoried: 输出文件 (.obj、.exe、.map 文件) 的目录。
 - ◆ Turbo C directoried: Turbo C 所在的目录。
 - Arguments (命令行参数): 允许用户使用命令行参数。
 - Save options (保存设置): 使用户可以将 Compiler、Linker、Environment、Debug 及 Project options 中所进行的设置保存起来, 生成配置文件, 默认的文件名是 TCCONFIG.TC。
 - Retrieve options(恢复设置): 调用以前保存的配置文件, 使用其中的设置来配置 Turbo C 系统。
- ⑦ Debug (调试) 菜单, 如图 1-7 所示。该菜单主要用于程序调试、查错、设置等。
- Evaluate (计算): 当程序运行时, 此命令可以使用户查看各变量或表达式的值, 并且可以修改这些值。执行此命令会弹出一个对话框, 如图 1-8 所示。

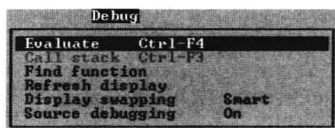


图 1-7 Debug 菜单

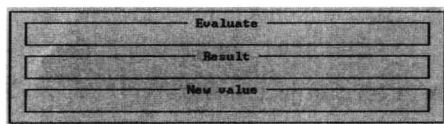


图 1-8 Evaluate 对话框

各项的含义如下。

- ◆ Evaluate: 要计算结果的表达式。

- ◆ Result: 显示表达式的计算结果。
- ◆ New value: 赋新值。

⑧ Break / watch (断点及监视表达式), 如图 1-9 所示。其中各项的含义如下。

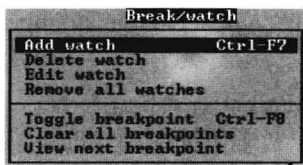


图 1-9 Break/watch 菜单

- Add watch: 向监视窗口插入一个监视表达式。
- Delete watch: 从监视窗口中删除当前的监视表达式。
- Edit watch: 在监视窗口中编辑一个监视表达式。
- Remove all watches: 从监视窗口中删除所有的监视表达式。

式。

- **Toggle breakpoint:** 对光标所在的行设置或清除断点。
- **Clear all breakpoints:** 清除所有断点。
- **View next breakpoint:** 将光标移动到下一个断点处。

(3) Turbo C 2.0 的配置文件

所谓配置文件, 是包含与 Turbo C 2.0 有关信息的文件, 其中存有编译、连接的选择和路径等信息。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 下输入: tc / c<用户命名的配置文件名>, 就会将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 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 文件中。Turbo C 启动时, 若没有找到配置文件, 则取 TC.EXE 文件中的默认值。

【实例】下面以建立名为 hello.c 的 C 语言源程序为例, 介绍在 Turbo C 的集成开发环境中建立一个新程序并编译、运行的步骤。

通常建立并执行一个 C 程序有以下几个步骤。

- ① 在编辑器中编写源文件。
- ② 编译和连接。
- ③ 生成可执行文件。

进入 TC 工作窗口后, 按 F3 键, 在随之出现的框中输入文件名, 文件名可以带.c, 也可以不带(此时系统会自动加上)。输入文件名后, 按回车, 即可将文件调入, 如果文件不存在, 就建立一个新文件(也可用下面例子中的方法输入文件名)。系统随之进入编辑状态, 就可以输入或修改源程序了。源程序输入或修改完毕以后, 按 Ctrl+F9 组合键, 则进行编译、连接和执行, 这 3 项工作是连续完成的。

(1) 建立并执行 C 程序的步骤

① 进入 Turbo C 集成开发环境, 通过键盘输入如下程序, 如图 1-10 所示。

```
void main()                /* 主函数 */
{
    printf("Hello, world!\n"); /* 输出语句 */
    printf("I am a student."); /* 输出语句 */
}
```

② 程序存盘: 在编辑窗口下, 可以使用 File 菜单中的 Save 命令存盘, 也可直接按功能键 F2 将文件存盘。这时, 系统将弹出一个 Rename NONAME 对话框, 这里将建立一个名为 hello.C 的文件, 如图 1-11 所示。

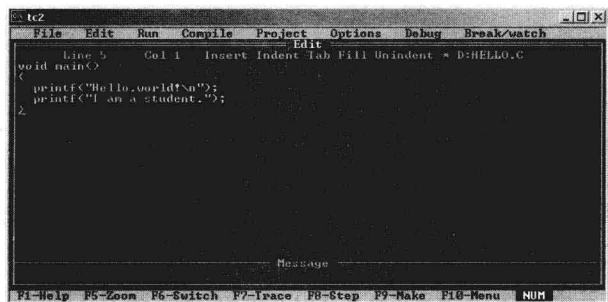


图 1-10 在 Turbo C 中输入源程序

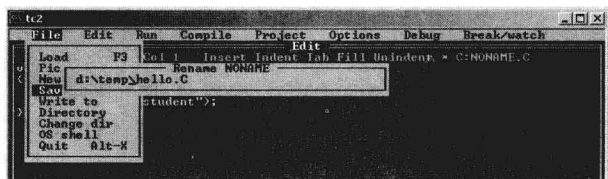


图 1-11 程序命名存盘

③ 编译一个程序：按 Alt+F9 组合键，进入编译状态后，会弹出一个编译窗口，几秒钟后，若显示信息为 Success:press any key，则表示编译成功。如果编译时产生警告（Warning）或出错（Error）信息，会显示在屏幕下部的信息窗口中，可按提示对源程序进行修改，重新进行编译。

④ 运行程序：源程序经编译无误后，按 Ctrl+F9 组合键运行，这时屏幕会出现一个连接窗口，显示 Turbo C 正在连接程序所需的库函数。连接完毕后，屏幕会突然一闪，又回到编辑窗口，此时可按 Alt+F5 组合键切换到程序输出窗口，查看输出结果。再按任意键，即可又回到编辑窗口。

这时，在磁盘上生成了 3 个文件：hello.c、hello.obj、hello.exe。其中文件 hello.c 是 C 语言源文件；文件 hello.obj 是 Turbo C 编译程序产生的二进制机器指令（目标代码）；文件 hello.exe 是 Turbo C 连接程序产生的可执行文件。

(2) 程序分析与解释

① 在上面的程序中，main() 表示“主函数”，每一个 C 语言程序都必须有一个主函数 main()；void 表示主函数 main() 没有返回值；函数体用一对大括号括起来。

② 本例中主函数有两条输出语句，printf 是 C 语言中的输出函数；双引号（双撇号）内的字符串按原样输出；“\n”是换行符，相当于在输出“Hello, world!”后按 Enter 键换行；语句的最后有一个分号，在 C 语言中是一行语句的结束标志。

③ “/*”和“*/”括起来的是注释部分，将有助于程序的阅读和理解，要注意这个符号的两部分缺一不可。

(3) 程序运行结果

```

Hello, world!
I am a student.

```

2. Microsoft Visual C++ 6.0 的集成开发环境

Visual C++ 6.0 是 Microsoft 公司出品的基于 Windows 环境的 C/C++ 开发工具,它是 Microsoft Visual Studio 套装软件的一个组成部分。C 语言源程序可以在 Visual C++ 6.0 集成环境中进行编译、连接和运行。

在使用 Visual C++ 6.0 之前,先要把它安装到系统中。该软件的安装很容易,只要从 Visual Studio 的光盘中运行其相应的安装程序 setup.exe,然后按照提示一步步操作即可。

安装成功后,选择“开始/程序/Microsoft Visual Studio 6.0/Microsoft Visual C++ 6.0”或双击桌面上的 Visual C++ 6.0 快捷图标就可以启动 Visual C++ 6.0,进入 Visual C++ 6.0 的编程环境。

(1) Visual C++ 6.0 的用户界面

如图 1-12 所示为 Visual C++ 6.0 用户集成开发环境的主界面。该界面由标题栏、菜单栏、工具栏、工作区窗口、源程序编辑窗口、输出窗口和状态栏组成。

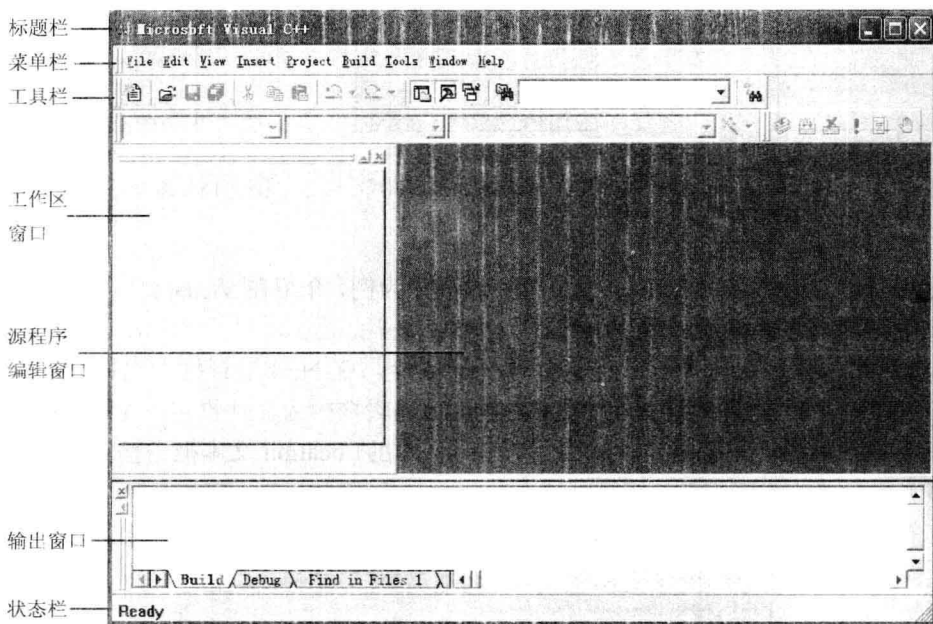


图 1-12 Visual C++ 6.0 的开发环境

屏幕最上端是标题栏,用于显示应用程序名和所打开的文件名。标题栏的下面是菜单栏和工具栏,工具栏的下面是两个窗口:一个是工作区窗口,一个是源程序编辑窗口。工作区窗口的下面是输出窗口,用于显示项目建立过程中所产生的各种信息。屏幕底端是状态栏,列出当前操作或所选择命令的提示信息。

(2) Visual C++ 6.0 集成开发环境常用命令的功能

① File (文件) 菜单,如图 1-13 所示。

- New: 弹出 New 对话框。
- Open: 弹出“打开”对话框,选择需要的程序并加载到内存中。

- Save: 将编辑区中的文件存盘。
 - Save As: 弹出“保存为”对话框, 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。
- ② Build (编译) 菜单, 如图 1-14 所示。
- Compile hello.c: 编译当前编辑区正在编辑的程序, 即 hello.c。当编译通过后, Build 菜单会发生变化, 如图 1-15 所示。
 - Build hello.exe: 创建 hello.exe 可执行文件。
 - Execute hello.exe: 执行 hello.exe 可执行文件。

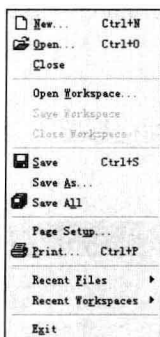


图 1-13 File 菜单

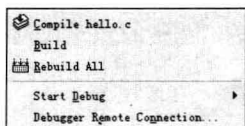


图 1-14 Compile 菜单

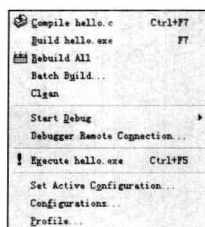


图 1-15 编译成功后的 Build 菜单

(3) 编辑并运行 C 程序

下面仍以建立一个名为 hello.c 的 C 语言源程序为例, 介绍在 Visual C++ 6.0 集成开发环境中建立一个新程序并编译、运行的过程。

① 建立 C 源程序。选择 File 菜单中的 New 命令, 在 New 对话框中选择 Files 选项卡, 单击 C++ Source File, 如图 1-16 所示。在右边的 File 编辑框中为文件指定一个名称, 如 hello.c, 如果不指定文件后缀, 系统将自动加上后缀.cpp。再在 Location 文本框中输入或选择文件保存的路径, 最后单击 OK 按钮。这样就成功建立了一个名为 hello.c 的空白源程序文件, 在源程序编辑窗口输入如图 1-17 所示的程序。

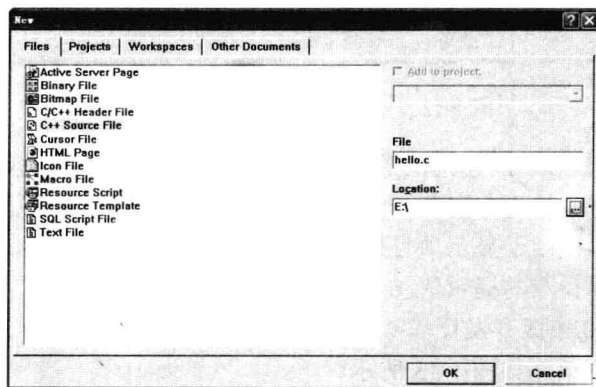


图 1-16 建立源程序

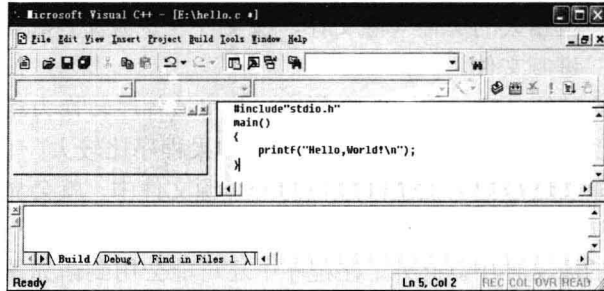




图 1-17 输入源程序

② 编译连接程序。编辑结束后，检查输入的内容。确认无误后，选择 Build 菜单中的 Compile hello.c 命令（直接按 Ctrl+F7 组合键，或单击工具栏上的  图标），开始编译。但在正式编译之前，VC 会先弹出一个如图 1-18 所示的对话框，询问是否建立一个默认的项目工作区。VC 必须有项目才能编译，所以这里必须单击“是”按钮。如果编译没有错误，在屏幕下方的输出窗口中将会显示 hello.obj - 0 error(s), 0 warning(s) 信息。如果在编译时得到错误或警告，用鼠标双击输出窗口中显示的编译错误信息，该错误信息所对应的程序语句所在行将会出现一个箭头标识，据此改正错误，再重新编译，直到编译通过为止。这时得到目标程序 hello.obj，就可以对程序进行连接了，方法是选择 Build 菜单中的 Build 命令（直接按 F7 键，或单击工具栏上的  图标），生成与源程序同名的可执行文件 hello.exe。

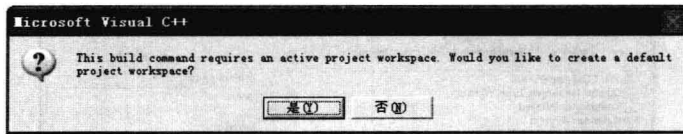



图 1-18 询问是否建立项目工作区

③ 运行程序。连接成功后，选择 Build 菜单中的 Execute hello.exe 命令（直接按 Ctrl+F5 组合键，或单击工具栏上的  图标）执行程序，程序将在一个类似于 DOS 的窗口中输出结果，如图 1-19 所示。在输出结果的下面显示 Press any key to continue，提示按任意键退出当前运行的程序，返回到开发环境中。按任意键，窗口关闭，返回 Visual C++ 6.0 的开发环境。

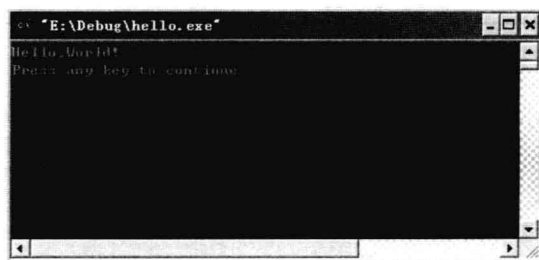


图 1-19 执行 hello.exe 程序的输出结果

(4) 建立工程

在 VC 中可以通过建立工程（project）来管理多个源程序文件。其实在 VC 中必须有工程（或项目），程序才能编译。那么，什么是工程呢？