

C语言程序设计教程

周启海 著



机械工业出版社
China Machine Press

计算机基础课程系列教材

C语言程序设计教程

周启海 著



机械工业出版社
China Machine Press

本书是一本创意新颖、体系独特的C语言程序设计教材，它首创了“同构化涵盖结构化，结构化映证同构化”的科学方法，采用最新的VC++环境作为上机实习平台；是从零起步，学习程序设计基本原理、主要思想、一般方法、基本技能及其C语言实现的良师益友。

全书行文浅显易懂，具有高中以上文化程度即可阅读，也可作为大专院校相关课程的教材或教学参考书以及计算机等级考试的培训用书。

版权所有，侵权必究。

图书在版编目（CIP）数据

C语言程序设计教程 / 周启海编著. - 北京：机械工业出版社，2004.1

（计算机基础课程系列教材）

ISBN 7-111-12310-7

I. C… II. 周… III. C语言-程序设计-教材 IV. TP312

中国版本图书馆CIP数据核字（2003）第100320号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：蒋 玮

北京瑞德印刷有限公司印刷·新华书店北京发行所发行

2004年1月第1版第1次印刷

787mm×1092mm, 1/16·17.25印张

印数：0 001-4 000册

定价：25.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

本社购书热线电话：（010）68326294

前 言

以程序设计为中心的计算机技术，是上世纪末以来发展最迅猛、应用最广泛的现代科学技术之一。计算机技术的开发与应用，说到底无非是计算机程序的设计与运用。“如何使中国学生轻松、愉快地学好计算机程序设计，怎样让全国教师愉快、轻松地教好计算机程序设计”，无疑具有重要的学术理论价值和社会实践意义。

有分析，才能比较；有比较，才能鉴别；有鉴别，才能发展；有发展，才有未来。为此，本书作者积极探索出了一条有中国特色的计算机程序设计教育的新捷径。其核心特征与主要特色是：

一、基础坚实——创立了“算法设计同构化，程序编码结构化”新理论。

从根本上讲，所有计算机语言都同构于一种同构化算法设计语言，并且这一同构化算法设计语言宛如各种计算机语言的“世界语”，或者恰似我国的“普通话”。同构化算法设计语言是所有计算机语言的公共原象，而且这一公共原象是从它的各种映像——所有计算机语言中抽象、简化、浓缩、提炼、升华而成的。同数学中 n 维欧氏空间与任何一个 n 维空间的数学结构关系相类似，用同构化算法设计语言（例如作者创造的算法周图、周码等）表述的同构化算法，与它所对应的任何一种具体计算机语言编写的程序之间的精神内核关系是同构的，即：它们的结构本质与构造实质是完全一致的。同样，所有计算机语言下的程序设计，显然同构于用同构化算法设计语言表征的同构化程序设计。所以，从学习同构化程序设计入手来学习和把握程序设计，无疑是计算机程序设计普及、推广、应用的新捷径；这不但因为从结构化程序设计基础发展起来的同构化程序设计，比通常特定的具体计算机语言下的程序设计简单易学、便于入门、易于开发，而且因为在掌握同构化程序设计之后，把一个同构化算法转换成它所对应的任一具体计算机语言下的同构化程序，将易如反掌。从1985年迄今，接受这种先进教学法教育的学生们的共同体会是：从算法到编码，其简单容易犹如将具体数据代入数学公式。

二、思想科学——突出了“算法设计为主导，语言编程为主体”新思想。

在计算机科学中，“计算机程序设计 = 算法设计 + 计算机语言”，而“算法设计 = 算法 + 数据结构 + 程序设计方法学”，且其核心和基础是算法。这表明，“算法乃程序之母，程序为算法之子”：算法是“源”，是程序行为的主导思想；而程序是“流”，是算法描述的主体形式。

然而，必须强调指出：算法设计，由于算法及其设计所依赖的思维方法的特殊规律性，大大地增加了人们学习它、掌握它的困难性。算法设计的思维方法，本质上不同于人们在从小学开始的数学教育与训练中所熟悉的公理系统思维方法（尽管也与它有所联系）。人们在初学计算机程序设计时，往往有茫然不知所措之感，概源于此。努力让人们去理解、熟悉这一整套新的算法构造思维方法，无疑是计算机程序设计普及教育的核心内容、首要难点与根本重点，而离开了算法及其设计这个计算机程序设计的基础、重点和关键，程序及其编码就成了无本之木、无源之水。只有突出了算法设计这个核心与重点，才能使学习程序设计的初学者“入门容易，提高自然，适应自如”。所以，在程序设计教学中，作者倡导“算法设计为主导，语言编程为主

体”的新思想。

三、方法独到——创造了“既授人以鱼，更授人以渔”新方法。

古人云：“授人以鱼，不如授人以渔”。同理，让人们学懂某种计算机语言的程序设计，远不如教会人们掌握可指导各种计算机语言程序设计和程序移植的算法设计的基本原理和一般方法。所以，应当“既授人以鱼（某种计算机语言的程序设计技术），更授人以渔（可适用于各种计算机语言的算法设计原理）”，以培养和形成学生的程序设计综合素质与拓展能力。为此，必须先“注重算法分析，突出算法设计，强化算法注释，点拨算法疑难”，再“结合某种语言（例如C语言），针对具体算法（例如秦九韶算法），实施程序编码（例如编写C语言秦九韶程序），上机调通程序（本书主选VC++ 6.0作为编程平台）”。

四、模式新颖——推出了“理论联系实际，算法制导程序，上机验证编程”新模式。

长期实践表明，这种新模式可切实做到授人以开启计算机程序设计宝库的金钥匙——进行计算机程序设计和程序移植的基本原理、一般方法与基础技术，使学生从根本上具备当今信息化社会所必需的计算机文化基本素质——“举一反三，触类旁通；各种语言，无师自通”的同构化程序设计应变能力和适应能力。

五、工具先进——进一步新创了“思想新颖，体系完备，技术先进，使用简便”的新型算法设计描述工具“周图、周码”，可极大地提高人们的算法（或程序）设计效率。

总之，大量事实证明：以同构化算法设计为指南来指导所有具体计算机语言下的程序设计，不但在思想上是先进的，在理论上是可靠的，在方法上是可行的，在模式上是创新的，而且在实践上是成功的。毋庸置疑，这一新思路和新捷径，必将有利于消除我国同发达国家在计算机普及、推广、应用上的差距。

常言道：“人无完人，金无赤金”。故本书倘有疏误，则恳请读者不吝赐教指正（来信请寄：610074 成都市 西南财经大学经济信息工程学院 周启海 教授；电话联系：028-87354172；E-mail: zhouqh@swufe.edu.cn），以便再版时予以更正。

周启海 教授

2003年11月11日，于西南财经大学

目 录

前言

第1章 引论

1.1 计算机应用概要	1
1.2 计算机语言、算法、程序概要	1
1.3 C语言及其编程平台概要	2
1.4 C语言常用编程平台入门——VC++、 Turbo C	3
1.4.1 VC++ 6.0及其进入、工作、退出	3
1.4.2 Turbo C 2.0的进入、工作、退出	9
1.5 算法与C程序的总体结构、注释结构 概述	13
1.6 C的字符集与标识符	14
1.6.1 C字符集	14
1.6.2 C标识符	15
1.7 数据的类型及其形式与运算	16
1.7.1 计算机数据的类型、操作概述	16
1.7.2 C语言的数据类型概述	17
1.7.3 数据的输入、输出操作概述	17
1.7.4 整数类型数据的形式与运算	25
1.7.5 实数类型数据的形式与运算	34
1.7.6 字符型、字符串型数据的形式与 运算	35
1.7.7 准逻辑型数据的形式与运算	40
1.7.8 各种变量的特殊运算——取地址(&) 与取内容(*)	43
1.7.9 指针型数据的形式与运算概述	44
1.8 程序设计基本过程概要	45
1.8.1 问题分析——程序设计的基础	46
1.8.2 算法设计——程序设计的核心	49
1.8.3 程序编码——程序设计的关键	50
1.8.4 分析调试——程序设计的保障	50
1.8.5 运行维护——程序设计的后援	51

1.8.6 人的主导作用与计算机的主体作用	51
-----------------------------	----

1.9 同构化程序设计方法概要	51
-----------------------	----

习题	52
----------	----

第2章 顺序结构

2.1 顺序结构的算法表示	55
---------------------	----

2.2 顺序结构设计	56
------------------	----

习题	60
----------	----

第3章 选择结构

3.1 选择结构概述	63
------------------	----

3.2 双分支选择结构设计	64
---------------------	----

3.2.1 一般条件型双分支选择结构设计 (if、if-else)	64
--	----

3.2.2 情况条件型双分支选择结构设计 (switch、break)	69
--	----

3.3 多分支选择结构设计	71
---------------------	----

3.3.1 一般条件型多分支选择结构中的 if嵌套	71
------------------------------------	----

3.3.2 情况条件型多分支选择结构中的 switch	77
--------------------------------------	----

3.3.3 混合条件型多分支选择结构中的 if、switch嵌套	80
---	----

习题	81
----------	----

第4章 循环结构

4.1 循环结构概述	85
------------------	----

4.2 单重循环结构设计	85
--------------------	----

4.2.1 当型单重循环结构设计(while)	85
-------------------------------	----

4.2.2 直到型单重循环结构设计(do-while)	95
-----------------------------------	----

4.2.3 步长型单重循环结构设计(for)	100
------------------------------	-----

4.2.4 强制缩短与中止循环过程设计 (break、continue以及goto)	109
---	-----

4.2.5 三种循环结构的比较	113
-----------------------	-----

4.3 双重循环结构设计	114
--------------------	-----

4.4 多重循环结构设计	115	7.2 数组及其下标变量的定义与使用	161
习题	117	7.2.1 数组及其下标变量的有序化与宜用 原则	161
第5章 模块化结构与子算法结构	123	7.2.2 数组型与数组的定义	162
5.1 模块化与子算法概述	123	7.2.3 下标变量和下标的定义	163
5.2 过程子算法结构设计	123	7.2.4 数组和下标变量的运算与使用	163
5.2.1 过程子算法的定义与调用	123	7.3 一维数组与单下标变量的应用	166
5.2.2 无参过程子算法结构设计	126	7.3.1 顺序存储与顺序标记	166
5.2.3 有参过程子算法结构设计	128	7.3.2 下标变量型计数器、累加器、累 乘器	168
5.2.4 子算法调用关系与定义位置	130	7.3.3 排序分类	170
5.2.5 标识符的作用域、生存期与误使用	131	7.3.4 查找检索	176
5.3 函数子算法结构设计	132	7.3.5 有序插入	179
5.4 嵌套、递推与递归	139	7.3.6 有序删除	181
5.4.1 嵌套	139	7.3.7 有序进退	181
5.4.2 递推	139	7.4 二维数组与双下标变量的应用	181
5.4.3 递归	140	7.5 多维数组与多下标变量的应用	184
5.5 过程参数与函数参数	144	7.6 一维字符型数组及其特殊用法示例	184
5.6 外部函数、内部函数与大型程序开发 方法	146	习题	185
5.6.1 外部函数、内部函数的定义方法与 基本特点	146	第8章 记录结构与共用型	191
5.6.2 内部函数与外部函数的综合应用	146	8.1 记录结构概述	191
习题	148	8.2 原象、实体、记录	191
第6章 C语言预处理与类型定义及其应用	151	8.3 记录的定义与使用	191
6.1 预处理的基本特点与使用用法	151	8.3.1 记录及其字段的秩序化与宜用原则	192
6.1.1 预处理、核心语言、标准库的主要 作用与基本特点	151	8.3.2 记录型与记录的定义	192
6.1.2 文件包含命令(#include)	152	8.3.3 记录和字段的运算与使用	194
6.1.3 宏处理——宏定义和宏替换	153	8.4 一般记录的应用方法	195
6.1.4 条件编译	155	8.5 数组记录与记录数组的应用方法	196
6.2 类型定义的一般方法与基本模式	156	8.5.1 数组的记录应用示例	196
6.3 枚举型结构程序设计	157	8.5.2 记录的数组应用示例	197
6.3.1 枚举型的类型定义与变量定义	157	8.6 共用型的定义与使用	198
6.3.2 枚举型量的运算与使用	157	习题	200
习题	158	第9章 文件结构	201
第7章 数组结构	161	9.1 文件结构概述	201
7.1 数组结构概述	161	9.2 对象数据、实体数据、内存数据、外存 数据	201
		9.3 文件及其数据的定义与使用	201

9.3.1 文件型与文件的定义	202	10.2.1 连通事物	225
9.3.2 文件的创建与重建	204	10.2.2 指针结构	226
9.3.3 文件数据的读入与写存	206	10.3 指针及其指针元素的定义与使用	227
9.4 文件的应用	210	10.3.1 指针型与指针的定义	227
9.4.1 文件的察看	210	10.3.2 指针值和指针元素的定义与使用	228
9.4.2 文件的复制	211	10.4 指针结构与半动态数据结构——栈、队	232
9.4.3 文件的修改	212	10.4.1 栈	232
9.4.4 文件的增删	213	10.4.2 队	240
9.4.5 文件的合并	216	10.5 指针结构与动态线性数据结构——链表	245
9.4.6 文件的排序	217	10.5.1 单向链表	245
9.5 文本文件	219	10.5.2 双向链表	252
习题	221	习题	258
第10章 指针结构及其应用	225	附录	261
10.1 指针结构概述	225		
10.2 连通事物与指针结构	225		

第1章 引 论

在以计算机科学技术为龙头和核心的21世纪,人类生产、工作、生活、休闲的基本方式和主要特征,是“社会信息化,经济知识化,世界网络化,竞争全球化”。无疑,学好、用好计算机,掌握至少一种计算机语言的程序设计基本技能,是当代社会各类人才必须具备的基本素质之一。为此,本书以“与时俱进”精神,对传统C语言程序设计入门课程进行深层改革、全面改进和教育创新:以“同构化涵盖结构化,结构化映证同构化”,用VC++作为C语言上机实习平台,走“算法制导程序,上机验证编程,学用促成应用”的编程新路,为无任何计算机知识的初学者尽快迈入现代程序设计殿堂,敞开了大门。

1.1 计算机应用概要

计算机系统由硬件系统与软件系统共同构成。计算机硬件如同计算机的躯体,由存储器、运算器、控制器与输入设备、输出设备、外存(储)设备构成。计算机软件(即所有计算机程序及其相关文件的总称)恰似计算机的灵魂,它由系统软件(如:操作系统、计算机语言、数据库系统)与应用软件(如:工具软件、杀毒软件、用户应用软件)构成。没有思想的人,只能是不会生活的植物人(僵尸);没有软件的计算机,只配称不能工作的“植物机”(裸机)。可见,计算机的软件系统的确比硬件系统更为重要。

计算机的应用正日益广泛地渗入和深入到人类社会经济生活的各个领域,它已经并仍将不断地影响和改变人类的生产方式、工作方式和生活方式。通常,计算机应用可分为:数值计算,数据处理,自动控制,辅助应用(如:辅助设计CAD、辅助制造CAM、辅助教育CAI、辅助测试CAT),人工智能等五大类。

1.2 计算机语言、算法、程序概要

计算机语言即程序设计语言(programming language),已走过了4代(即:机器语言、汇编语言、高级语言、非过程语言)的发展历程,现在正向着第5代(即:智能语言)积极迈进。然而,应当强调指出:计算机语言的科技进步,与计算机硬件的科技进步性质全然不同:新一代计算机语言的产生,只表明它为更多的计算机用户提供了又一种更为便捷的新语言工具,而丝毫不意味着其他计算机语言历史使命的结束,因为每一代计算机语言(哪怕是机器语言)至今仍各有其特殊专长与用武之地。

算法是人们为了借助计算机解决给定问题而命令计算机按照“人-机”系统所认可的操作方式和控制方式,对一步一步具体施行的有穷操作过程的描述。算法就是计算机的“解题思想”,它颇类似于人们解决给定问题的特殊方法、想法、思路、规程、处方,等等。

计算机程序简称程序(program)。通俗地讲,程序是人们在所设计的、能控制计算机正确解决给定问题的算法的基础上,进一步用某种选定的计算机语言,把算法翻译成计算机可接受、读懂和执行的“计算机语言文章”。换言之,程序是人交给计算机具体实施的工作任务书和行为

指南针。应当强调指出：有如“同一事物，可用不同自然语言（例如：中文、英文、法文、俄文）表述而说法不同”，同一个算法，完全可以根据用户的实际需要和编程爱好而采用不同的计算机语言来进行编码，并得到关于同一算法的具有不同计算机语言表现形式的程序。这些不同计算机语言的程序，其执行效果是完全一致的（例如，同一算法可以采用C语言、QBASIC语言、FORTRAN语言进行翻译，而其效果相同）。由此可见，算法远比程序深刻得多、基本得多、重要得多，两者的关系是“算法是程序之母，程序是算法之子”。

1.3 C语言及其编程平台概要

20世纪70年代初，为高效开发UNIX操作系统，美国贝尔实验室的Dennis Ritchie完成了“源于ALGOL（ALGOritm Language, 1960年）、始于CPL（Combined Programming Language, 1963年）、继于BCPL（Basic Combined Programming Language, 1967年）、承于B（Basic combined programming language, 1970年）”的技术改进，成功开发出“把高级语言根本优势与汇编语言主要特长巧集于一身”的结构化程序设计语言——C语言（basic Combined programming language, 1972年）。

C语言有“中级语言”美称，属于编译型语言，故只有在编译、连接通过后其程序才能运行。它具有“语言精干，功能强大；数据丰富，使用灵活；结构简洁，易于共享；硬件控制，简单可靠；软件开发，简练高效；程序快速，便于移植”的独特优点；但与其他高级语言相比也有“语言过活，容易出错；规则过多，不易纠错；出错自纠，尚逊不足；编程要求，似嫌偏高”的弱点。因此，国内外计算机教育界和计算机科技界普遍认为：C语言是计算机系统软件、应用软件开发的最主要工具语言之一，也可作为学习计算机程序设计的常用教学语言之一。

程序设计学习必须以理论学习为主体和基础，以上机实践为辅助和补充。因此，任何计算机语言程序设计，都需要自己特定的上机编程平台环境；否则，就只能成为纸上谈兵的纯理论学习。典型的C语言编程平台，早期是Turbo C，中期是C++，如今是VC++（Visual C++的简称）。

Turbo C是美国Borland公司于20世纪80年代末开发成功、90年代不断改进的著名Turbo系统软件（包括Turbo BASIC, Turbo PASCAL, Turbo C, Turbo C++等）之一。它继承和发扬了标准C语言的全部优势，而且新增了“开发环境友好，语言功能更强；设计风格优雅，编程调试简便；执行效率高，兼容性能优良（注意：C++、VC++均向下兼容C语言）；可扩充新设计方法（如Turbo C++已支持“对象化程序设计”或称“面向对象程序设计”），符合软件工程规范”等一系列新优势，曾是20世纪风靡一时的C语言上机编程平台。

但Turbo C、Turbo C++毕竟是主要基于DOS操作系统的“老”编程平台，其编程环境远远无法与当今主流操作系统WINDOWS支持的后起之秀——VC++编程平台相媲美。VC++为了向下兼容C/C++语言而特意提供了主/辅两套编译器：对扩展名为“.CPP”的C++、VC++源程序，会自动选用Visual C++本身的VC++主编译器；而对扩展名为“.C”的C源程序，则自动选用专向下兼容C语言的C辅编译器。

然而，应当指出：虽然VC++向下兼容C语言，但绝不应因此而把VC++降格成纯结构化语言来使用，而应主要遵循对象化风格来使用它，以充分体现VC++的对象化优势。本书为了便于读者提高C语言理论学习与上机实践效率，利于读者向C++、VC++过渡和升级，故特意把VC++当作C语言的主编程平台来使用。

1.4 C语言常用编程平台入门——VC++、Turbo C

本节只简述VC++ 6.0编程平台的基本用法，也略讲Turbo C 2.0编程平台的一般使用。

1.4.1 VC++ 6.0及其进入、工作、退出

VC++ 6.0是第4代计算机语言（包括C++、VB、VC++、Delphi、Java、C#）的典型代表之一，是C、C++家族中最时新的佼佼者。它的硬、软件上机条件要求较高，但界面美观、环境友好、操作简便、效果可视、利于实习。下面简述VC++ 6.0的进入、工作、退出过程。

1. VC++ 6.0的进入

在WINDOWS操作系统状态下，确认安装有VC++ 6.0后，可用如下方法之一打开VC++。

(1) “开始”按钮方法

施行“开始 → 程序 → Microsoft Visual Studio 6.0 → Microsoft → Visual C++ 6.0”操作后，即可进入VC++ 6.0初始主界面，如图1-1所示。其中，右箭头“→”表示“单击、选项、输入、打开”有关按钮、图标、菜单项等人-机交互界面操作之一；下同。

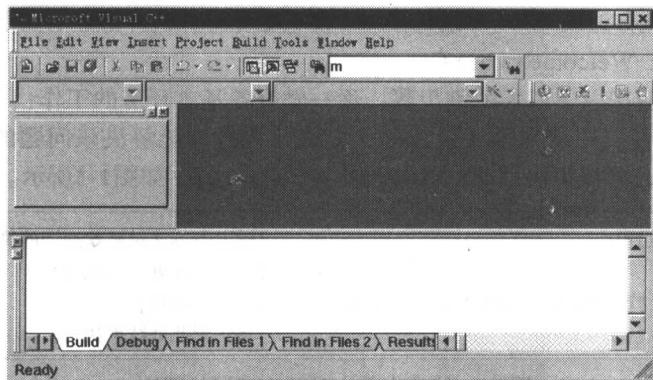


图1-1 VC++ 6.0的初始主界面

(2) “运行”程序方法

施行“开始 → 运行 → C:\Program Files\Microsoft Visual Studio\Common\MSDev98\Bin\MSDEV.EXE → 打开”操作后，即可进入VC++ 6.0初始主界面，如图1-1所示。

(3) “快捷”图标方法

确认建有VC++ 6.0的快捷方式后，施行“开始 → Microsoft Visual C++ 6.0快捷图标”操作后，即可进入VC++ 6.0初始主界面，如图1-1所示。

(4) “打开”C语言程序方法

确认建有C语言程序文件（*.C）后，找到并打开该C语言程序文件（例如：Eg0101.C），即可进入VC++ 6.0的编辑主界面，如图1-2所示。

2. Visual C++ 6.0的工作

进入VC++ 6.0工作状态后，就可以在由它的“窗口-菜单”系统构成的工作环境下，方便地施行C语言程序的指定、输入、编辑、修改、编译、调试、运行等处理工作。而C语言的各项工

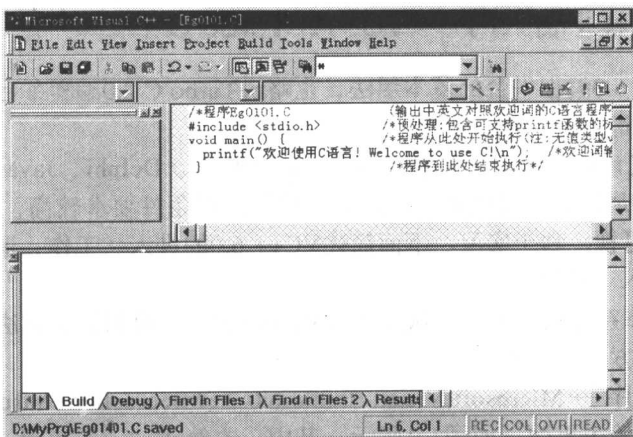


图1-2 VC++ 6.0的C++旧程序编辑主界面

例1.1 试用结构化方法设计一个C语言程序，使它能让计算机屏幕显示如下中英文对照欢迎词：
欢迎使用C语言！ Welcome to use C!

为此，人们通常应按以下三大关键步骤，逐一做好各基本环节的工作：

第一步，算法设计。用结构化（或其他）方法，设计出能解决该问题的算法，并用某种表现方式描述之。为此，可得解决该问题的结构化算法Eg0101，如图1-3所示。

算法Eg0101	{输出中英文对照欢迎词的算法}
>>>	{算法从此处开始执行}
行输出 "欢迎使用C语言！ Welcome to use C!";	{欢迎词输出}
!!!	{算法到此处结束执行}

图1-3 输出中英文对照欢迎词的算法

（注意：在算法描述中，凡花括号“{}”及其内容，以及C语言程序中注释号“/*”、“*/”及其后的内容，均为帮助人们理解的注释内容，注释的有无、多寡与算法及其C语言程序功能无关！）

第二步，程序编码。用C语言对算法进行编码，得程序Eg0101.C如下（注意：实际上，它就是把算法的C语言对应表示形式“代入”算法！）：

```
/*程序Eg0101.C          (输出中英文对照欢迎词的C语言程序)*/
#include <stdio.h>        /*预处理:包含可支持printf函数的标准输入输出头文件stdio.h*/
void main() {            /*程序从此处开始执行(注:无值类型void说明主函数没有返回值)*/
    printf("欢迎使用C语言! Welcome to use C!\n"); /*欢迎词输出*/
}                          /*程序到此处结束执行*/
```

顺便说明，上述程序Eg0101.C虽可在语法上简化为完全等效的如下最简形式，但不宜提倡：

```
main() { printf("欢迎你使用语言C! Welcome to use C!\n"); }
```

这是因为：程序注释均可略去，而书写格式可较自由；标准输入、输出函数scanf、printf可缺省用以指定标准（或称系统）头文件stdio.h的预处理命令“#include <stdio.h>”（详见第6章）；主函数main()只能由C语言系统本身所调用，且往往无需返回其函数值（简称返回值），而可省略其返回值类型——无值类型或称空类型void。

第三步，上机调试。只设计出上述C语言程序仍然是不够的，因为：要在VC++状态下（或者操作系统状态下），让计算机输出这段欢迎词，还必须在VC++系统环境下进一步完成“由人指定、输入、编辑程序，由计算机编译、连接、运行程序，由人察看、分析、利用程序运行结果，由人保存、评价、复用（即重复利用的简称）程序文件”。

指定当前程序

首先，由于VC++的当前项目只支持自己的相关文件，因此：在指定新当前程序之前，应确保VC++的旧当前项目及其文件均已关闭（不然，就务必在VC++主界面下，先施行关闭文件操作“File → Close”与关闭项目操作“File → Close Workspace”）；否则，不可正常指定和使用新当前程序。然后，按所需C语言程序是否存在，而分别采用不同方式进行指定当前程序的操作。

(1) 方法1——“主界面File”菜单方式。在VC++主界面下：施行操作“File → New”，进入VC++的创建对象对话框，如图1-4所示。在VC++的创建对象对话框内：施行操作“File → C++ Source File”，进入VC++的创建C++程序对话框，如图1-5所示；在VC++的创建C语言程序对话框的文件名输入文本框内，施行操作“D:\MyPrg\Eg0101.C → OK”（注意：有下划线者的内容，必须由用户自己输入；下同），进入VC++的C语言新程序编辑主界面，如图1-6所示。

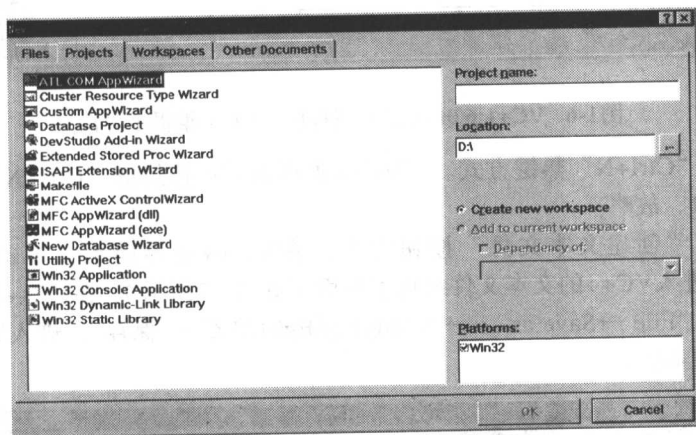


图1-4 VC++ 6.0的创建对象对话框

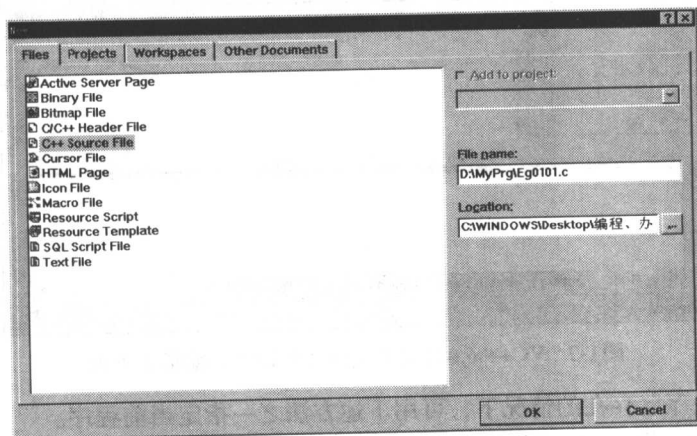


图1-5 VC++ 6.0的创建C语言程序(*.C)对话框

警告：图1-5中，其文件名输入对话框内，用户所输入的C语言程序文件名必须是全称，即不但要有文件的基本名（或称主名），而且一定要包括该文件的扩展名（.C）；否则，VC++所创建的程序，只会是系统默认的C++程序（*.CPP），而不会是所希望的C语言程序（*.C）！

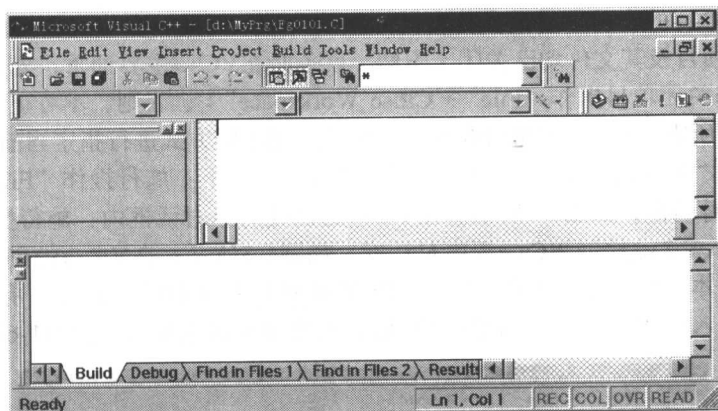


图1-6 VC++ 6.0的C语言新程序（*.C）编辑主界面

(2) 方法2——“Ctrl+N”热键方式。在VC++主界面下：按热键“Ctrl+N”后，其余操作和效果完全同于方法1，故略。

(3) 方法3——“创建文本文件”按钮方式。在VC++主界面下：单击“创建文本文件（*.TXT）”按钮，进入VC++的文本文件编辑主界面（如图1-7所示）。在VC++的文本文件编辑主界面内：施行操作“File → Save as ... → D:\MyPrg\Eg0101.C → 保存”，进入VC++的C语言程序编辑主界面（如图1-6所示）。

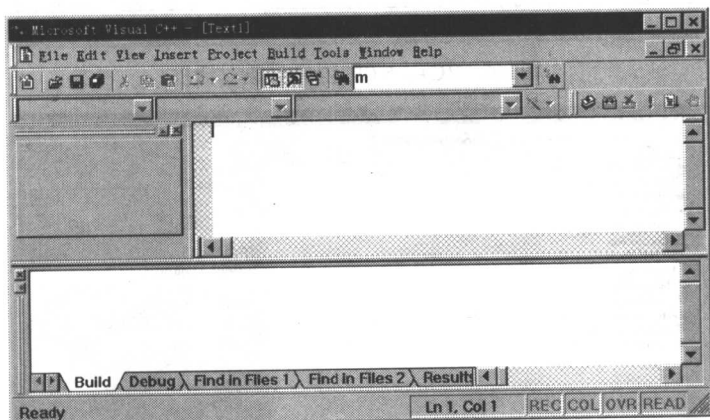


图1-7 VC++ 6.0的文本文件（*.TXT）编辑主界面

在所需C语言程序已存在的情况下，可用下述方法之一指定当前程序。

(1) 方法1——“主界面File”菜单方式。在VC++主界面下，施行操作“File → Open”，进入

VC++的打开旧文件对话框（如图1-8所示）。在VC++的打开旧文件对话框内，单击“查找范围”的▼按钮，选定旧文件所在路径；在文件夹/文件窗内，选定旧文件名；单击“打开”按钮，进入VC++的C语言旧程序编辑主界面（如图1-2所示）。

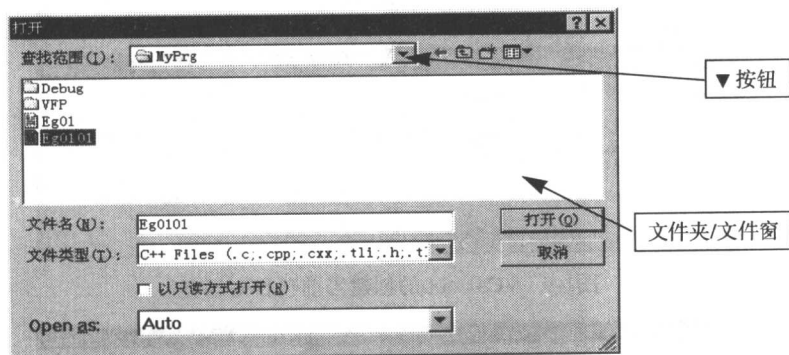


图1-8 VC++ 6.0的打开C++旧程序 (*.C) 对话框

(2) 方法2——“Ctrl+O”热键方式。在VC++主界面下：按热键“Ctrl+O”后，其余操作和效果完全同于上述方法1，故略。

(3) 方法3——“打开旧文件”按钮方式。在VC++主界面下，单击“打开旧文件”按钮后，其余操作和效果完全同于上述方法1，故略。

输入、编辑程序内容

在当前C语言程序编辑主界面内，正确键入、编辑C语言程序（例如Eg0101.C）全部内容。（请初学者特别注意：每一个程序行的自身内容输入完毕时，都务必以回车键作为该程序行输入的结束，绝不可犯“用若干个空格来造成纯视觉意义上的假换行”的错误。否则，可能会出现你意料不到的错误！）

编译当前程序 (*.C)

计算机只能实际运行C程序的可执行程序“*.EXE”，而不能直接运行扩展名为“.C”的文本状态的C程序。为此，C语言程序（例如Eg0101.C）键入、编辑完毕后，可采用如下两种方法之一进行编译，生成目标程序 (*.OBJ)。

(1) 方法1——“主界面Build”菜单方式。在VC++主界面下：施行操作“Build → Compile Eg0101.C”，如果出现VC++ 6.0创建当前项目的对话框（如图1-9图所示），再都选按钮“是”，就进入完成C语言程序编译及其目标程序 (*.OBJ) 生成主界面（如图1-10所示）。如果用户当前程序出现编译错误（即C语言的语法错误），则系统将会在下方的信息窗口自动指示出错位置与错误性质，以提醒用户及时修改；改正后，再重复本步操作，以重新编译修正后的当前程序，直至编译成功为止。（请初学者小心：真正导致错误的根源处，未必就一定就是系统指示出错位置！）

(2) 方法2——“Ctrl+F7”热键方式。在VC++主界面下：按热键“Ctrl+F7”后，其余操作和效果完全同于“编译当前程序”的方法1，故略。

在VC++下生成可执行程序 (*.EXE)

可采用如下两种方法之一，把目标程序 (*.OBJ) 连接成可执行程序 (*.EXE)。

(1) 方法1——“主界面Build”菜单方式。在VC++主界面下：施行操作“Build → Build Eg0101.exe”，就进入完成C语言程序可执行程序文件 (*.EXE) 生成主界面（如图1-11所示）。

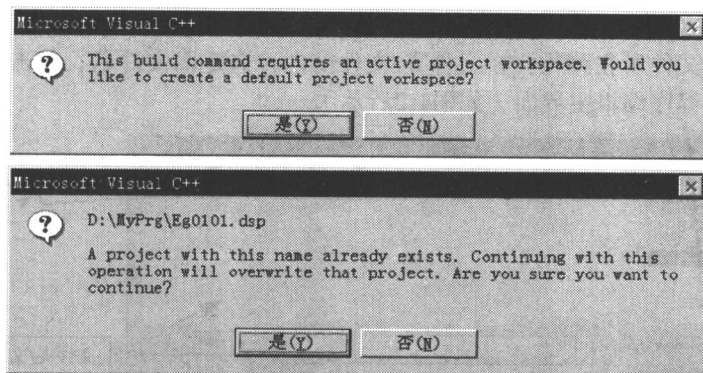


图1-9 VC++ 6.0的创建当前项目对话框

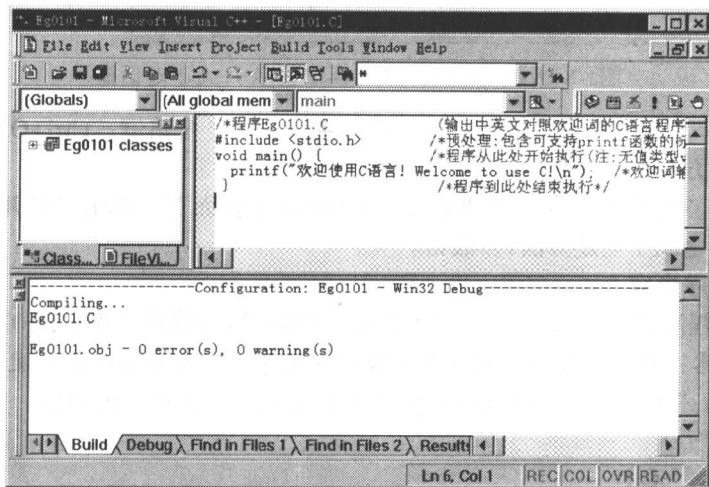


图1-10 VC++ 6.0的C语言程序编译及其目标程序(*.OBJ)生成主界面

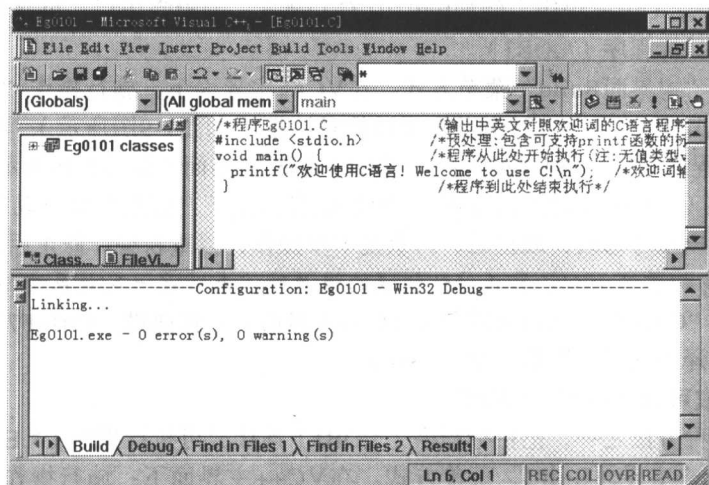


图1-11 VC++ 6.0的C语言可执行程序(*.EXE)生成主界面

(2) 方法2——“F7”热键方式。在VC++主界面下：按热键“F7”后，其效果完全同于“在VC++下生成可执行程序”的方法1，故略。

在VC++下运行当前可执行程序 (*.EXE)

在VC++下运行可执行程序 (*.EXE)，可采用如下两种方法之一。

(1) 方法1——“主界面Build”菜单方式。在VC++主界面下，施行操作“Build → Execute Eg0101.exe”，就运行C语言程序可执行程序文件 (*.EXE)，并得到其运行结果（如图1-12所示）。

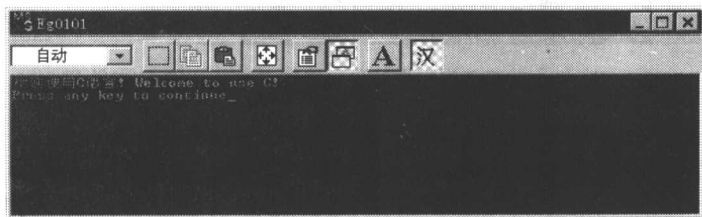


图1-12 VC++ 6.0下C语言可执行程序 (*.EXE) 运行结果示例

(2) 方法2——红色“!”按钮方式。在VC++主界面工具栏内，单击红色“!”程序运行按钮后，其效果完全同于“在VC++下生成可执行程序”的方法1。顺便说明：只要编译并生成过当前目标程序，单击红色“!”按钮，就可“编译、生成、运行可执行程序”一并完成（故简化了程序修改后的运行操作）。

保存当前程序（以文本文件方式存盘）

在程序编辑的全过程中，为了防止当前程序的意外丢失，并能够在日后重复利用它，必须及时将它存盘。其存盘操作方法，同于一般文件的存盘，故略。

在操作系统下运行可执行程序 (*.EXE)

在操作系统下，运行C语言的可执行程序 (*.EXE)的操作方法，同于运行一般可执行程序“的开始”按钮、“运行”程序、“快捷”图标、“打开”可执行程序等多种方法，故略。

3. Visual C++ 6.0的退出

当计算机处于VC++的工作状态时，如果确认各项工作已全部完成，则应结束使用，从VC++状态退出，并返回到操作系统状态下。其退出操作，可采用如下两种方法之一。

(1) 方法1——“主界面File”菜单方式。在VC++主界面下：施行操作“File → Exit”，即退出VC++，回到WINDOWS。

(2) 方法2——“关闭主界面总窗口”按钮方式。在VC++主界面下，单击“关闭主界面总窗口”按钮，便退出VC++，回到WINDOWS。

1.4.2 Turbo C 2.0的进入、工作、退出

开始使用Turbo C 2.0之前，必须首先进入它的工作状态；结束使用Turbo C 2.0之时，必须及时退出它的工作状态。

1. Turbo C 2.0的进入

当计算机处于DOS工作状态时，首先进入Turbo C 2.0所在子目录，然后直接执行主文件——可执行文件“TC.EXE”，便从DOS工作状态进入Turbo C 2.0工作状态，并使计算机立即出现如图1-13所示的Turbo C 2.0的集成开发环境（Integrated Development Enviroment，以下简称IDE，它由一系列菜单与窗口构成）的初始编辑窗口。