

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

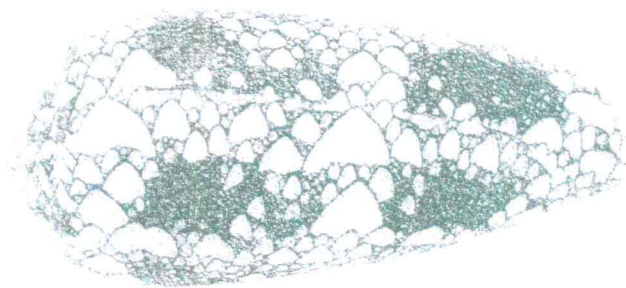
C语言程序设计 实践教程

C Language Programming Practice Tutorial

王电化 王忠友 主编

鲁屹华 杨胜 副主编

- 基础知识理论系统完整
- 编程和算法思想性鲜明
- 案例项目编写趣味生动



高校系列



人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等学校计算机规划教材

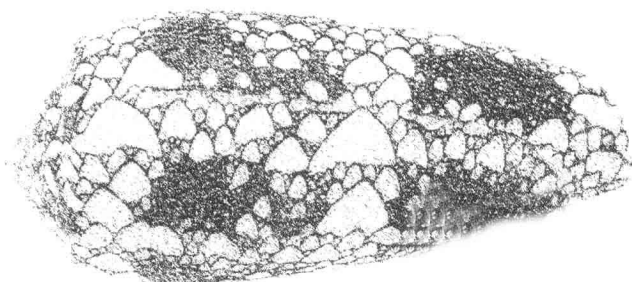
21st Century University Planned Textbooks of Computer Science

C语言程序设计 实践教程

C Language Programming Practice Tutorial

王电化 王忠友 主编

鲁屹华 杨胜 副主编



人民邮电出版社
北京

图书在版编目 (C I P) 数据

C语言程序设计实践教程 / 王电化, 王忠友主编. --
北京: 人民邮电出版社, 2013. 3
21世纪高等学校计算机规划教材. 高校系列
ISBN 978-7-115-30069-0

I. ①C… II. ①王… ②王… III. ①
C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(2013)第025270号

内 容 提 要

本书是《C语言程序设计项目教程》一书的配套教学参考书。书中详细介绍了 Visual C++6.0 的开发环境,并设计了覆盖《C语言程序设计项目教程》所有知识点的基础实验,按照软件工程学方法设计了一个覆盖本书主要章节的综合实训项目;本书还详细地解答了《C语言程序设计项目教程》一书中设计的习题,最后列出了全国计算机等级考试 C 语言考试大纲和 2012 年全国计算机等级考试 C 语言试题及答案。

本书中所有的实验代码在 Visual C++6.0 中测试通过,习题大多数来自历年的全国计算机等级考试试卷和一些大公司 C 语言测试试卷,实验及题目都是精心设计和安排的。

本书是学习 C 语言和上机实验的必备参考书,可以作为高等院校计算机应用的 C 语言程序设计实验教学用书,也可以作为备考全国计算机等级考试二级 C 语言的参考书。

21 世纪高等学校计算机规划教材——高校系列

C 语言程序设计实践教程

-
- ◆ 主 编 王电化 王忠友
副 主 编 鲁屹华 杨 胜
责任编辑 韩旭光
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京铭成印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 15 2013 年 3 月第 1 版
字数: 398 千字 2013 年 3 月北京第 1 次印刷

ISBN 978-7-115-30069-0

定价: 32.00 元

读者服务热线: (010)67132746 印装质量热线: (010)67129223
反盗版热线: (010)67171154

前言

C 语言程序设计是一门实践性很强的课程,要想学好 C 语言就得通过大量的实践,在实践中发现问题、研究问题,并且解决问题,这样才能更好地理解 C 语言,并最终学会使用 C 语言。本书从 3 个方面组织教材帮助读者提高对 C 语言的进一步掌握和理解。首先,书中安排了大量的基础实训实验,这类实验基本覆盖了教材中 C 语言知识点,让读者通过实验的引导理解 C 语言知识的一些细节;其次,安排了贯穿本书的一个综合实训项目,该综合实训项目的目的是为了让读者明白所学的知识在真实项目中大致能做的事是什么,同时也可以进一步提高读者分析和解决问题的能力;最后,本书对《C 语言程序设计项目教程》书中的所有的习题都给出了答案,并对大部分题目做了分析,给出解题思路,以便读者进一步掌握各知识点,并提高分析问题和解决问题的能力。

本书是《C 语言程序设计项目教程》配套用教材,内容包括五部分:第一部分,C 语言上机环境和基本操作方法介绍,主要为第 1 章,介绍 Visual C++集成开发环境和基本的使用方法,并且介绍了在 Visual C++上编辑、编译、调试 C 语言程序的基本步骤和方法;第二部分,C 语言上机实验,包括 11 个实验项目,每个实验大体上又分基础实训项目和综合实训项目,基础实训项目目的是帮助读者解决对应的知识点,综合实训项目是从《学生成绩管理系统》中分离出来的子项目,目的是帮助读者了解所学知识能处理哪些实际问题,进一步提高分析问题和解决问题能力;第三部分,习题和参考解答,习题来自于《C 语言程序设计项目教程》一书,本书中对所有的习题都给了解答,对部分习题还给出了提示,帮助读者找到正确的解题方法;第四部分,全国计算机等级考试大纲和全国计算机等级考试真题一套。学习完本书之后,读者可以利用真题考查自己 C 语言学习的程度;第五部分,学生成绩管理系统的源代码,阅读源码、研究源码可以快速提高 C 语言程序设计能力。

本书由王电化、王忠友、鲁屹华、杨胜编写。本书第 1 章,第 2 章 2.1 节~2.3 节,第 3 章 3.1 节~3.3 节、3.12 节~3.13 节、附录 I 由王电化编写;第 2 章 2.4 节~2.6 节、第 3 章 3.4 节~3.6 节,附录 II 由王忠友编写;第 2 章 2.7 节~2.9 节、第 3 章 3.7 节~3.9 节由鲁屹华编写;第 2 章 2.10 节~2.11 节、第 3 章 3.10 节~3.11 节由杨胜编写;本书实验部分的综合实训项目由王电化、王忠友和鲁屹华共同设计并完成附录 III 的代码编写工作,本书最后由王电化统稿。

由于时间紧迫,以及编者水平有限,书中难免有错误或者不妥之处,恳请广大读者批评指正。

编 者
2012 年 12 月

目录

第1章 Visual C++6.0

集成开发环境 1

- 1.1 Visual C++6.0 概述 1
- 1.2 Visual C++6.0 的安装和卸载 1
- 1.3 Visual C++6.0 的功能介绍 3
 - 1.3.1 Visual C++6.0 的菜单栏 4
 - 1.3.2 Visual C++工具栏 8
- 1.4 Visual C++6.0 编写 C 语言程序过程 9
 - 1.4.1 编写 C 语言源码程序 10
 - 1.4.2 编译源程序, 构建应用程序 11

第2章 C 语言项目实战 13

- 2.1 概述 13
 - 2.1.1 基础实训项目 13
 - 2.1.2 综合实训项目 16
- 2.2 数据类型、运算符和表达式 18
 - 2.2.1 基础实训项目 18
 - 2.2.2 综合实训项目 21
- 2.3 顺序结构程序设计 22
 - 2.3.1 基础实训项目 22
 - 2.3.2 综合实训项目 23
- 2.4 选择结构程序设计 25
 - 2.4.1 基础实训项目 25
 - 2.4.2 综合实训项目 31
- 2.5 循环结构程序设计 36
 - 2.5.1 基础实训项目 36
 - 2.5.2 综合实训项目 42
- 2.6 数组 43
 - 2.6.1 基础实训项目 43
 - 2.6.2 综合实训项目 48
- 2.7 结构体和共用体 49
 - 2.7.1 基础实训项目 49
 - 2.7.2 综合实训项目 56
- 2.8 函数及编译预处理 62
 - 2.8.1 基础实训项目 62

- 2.8.2 综合实训项目 68

2.9 指针 70

- 2.9.1 基础实训项目 70
- 2.9.2 综合实训项目 75
- 2.10 链表 78
 - 2.10.1 基础实训项目 78
 - 2.10.2 综合实训项目 91
- 2.11 文件 92
 - 2.11.1 基础实训项目 92
 - 2.11.2 综合实训项目 98

第3章 习题和参考解答 100

- 3.1 概述 100
- 3.2 数据类型、运算符和表达式 102
- 3.3 顺序结构程序设计 108
- 3.4 选择结构程序设计 115
- 3.5 循环结构程序设计 126
- 3.6 数组 138
- 3.7 结构体和共同体 145
- 3.8 函数及编译预处理 154
- 3.9 指针 162
- 3.10 链表 172
- 3.11 文件 178
- 3.12 算法与数据结构 184
- 3.13 软件开发基础知识 188

附录 I 全国计算机等级 考试大纲 193

附录 II 全国计算机等级考试 二级笔试 C 语言程序 设计试卷及答案 197

附录 III 学生成绩管理系统 源代码清单 206

参考文献 236

第 1 章

Visual C++6.0 集成开发环境

1.1 Visual C++6.0 概述

Visual C++6.0 集成开发环境是美国 Microsoft 软件公司推出的目前应用极为广泛的可视化软件开发环境。自 1993 年 Microsoft 开发出 Visual C++1.0 版本以来,随着 Microsoft 不断地对该工具的改进和优化,到 Visual C++6.0 以后,因其功能强大、灵活性好、完全可扩展性,以及强有力的 Internet 支持,从众多的 C++开发环境中脱颖而出,成为目前最为流行的、应用广泛的 C++语言集成开发环境之一。

虽然 Microsoft 后面开发出 Visual C++.net(又可称为 Visual C++7.0),但是该集成开发环境只能用于 Windows 2000 以上的操作系统中,在配置较低的 Windows 98 等操作系统上不能运行。

Visual C++6.0 是一个集成开发环境,不是一个简单的源码编写、编译的软件系统。Visual C++6.0 主要由 3 个部分组成。

(1) Developer Studio

这是 Visual C++6.0 的外壳,是我们编写和运行各种菜单功能的框架,它主要给我们开发软件时提供一个编辑器环境和一些 Wizard(向导)功能,源码调试、编译、链接成目标代码是由 Platform SDK 提供的。

(2) MFC

由 Microsoft 提供的一个强大的 Windows 文档视图软件类库。当我们开发 Windows 窗口应用程序时,借助该类库可以大大地提高我们的开发速度。

(3) Platform SDK

这部分才是 Visual C++6.0 的核心,所有的编译、链接和调试程序是由这部分提供。同时还提供汇编模块、大量的帮助文档和一些辅助工具。

本书是面向 C 语言的实验教程,由于 C 语言是过程化语言,所以在使用 Visual C++6.0 时,不会用到 MFC。

1.2 Visual C++6.0 的安装和卸载

软件安装对硬件环境和操作系统环境都有一定要求,要想将 Visual C++6.0 安装成功,对系统

的最低要求是，硬件环境为 Pentium 处理器，32M 内存和至少 200M 的硬盘空间，计算机要配置光驱，显示器的最低分辨率为 800×600，软件环境要求是 Windows 95 以上的操作系统。

安装过程如下。

(1) 将 Visual C++6.0 安装光盘插入光驱，打开安装向导出现如图 1.1 所示界面。



图 1.1 “安装向导”界面

(2) 单击“下一步”按钮，出现如图 1.2 所示对话框。

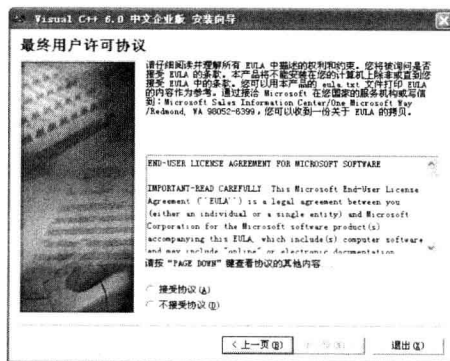


图 1.2 “最终用户许可协议”对话框

(3) 选择“接受协议”单选按钮，然后单击“下一步”按钮，系统提示输入序列号、用户名和所在单位名称，单击“下一步”按钮，出现如图 1.3 所示对话框。

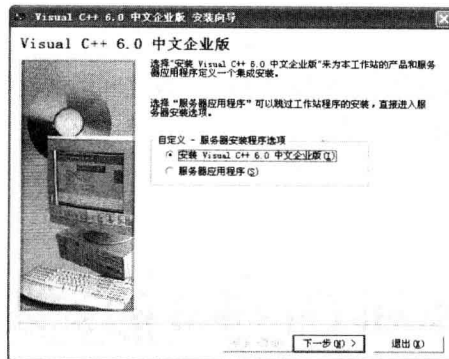


图 1.3 “安装程序选项”对话框

(4) 选择“安装 Visual C++6.0 中文企业版”然后单击“下一步”按钮, 出现如图 1.4 所示对话框。

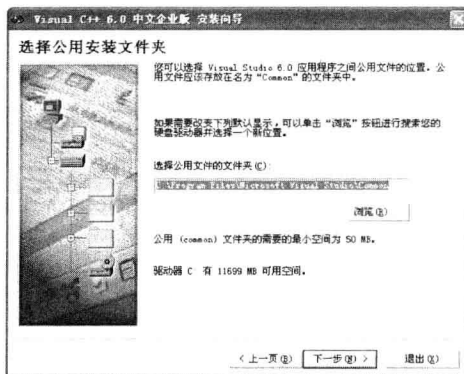


图 1.4 “选用公用安装文件夹”对话框

(5) 按提示选好公用文件的文件夹后, 点击“下一步”按钮, 在后续的对话框中直接选择“确定”按钮即可, 然后就出现文件移动动画对话框, 当动画对话框到 100%后就出现安装 msdn, 由于我们是在其上编写 C 语言程序, 这个安装不用理会, 直接点击“下一步”按钮或“确定”按钮, 到最后点击“完成”按钮则表示 Visual C++安装完成。

(6) 在桌面上“开始”菜单中选择“程序”然后选择“Microsoft Visual C++6.0”, 出现如图 1.5 所示窗口则表示安装成功。

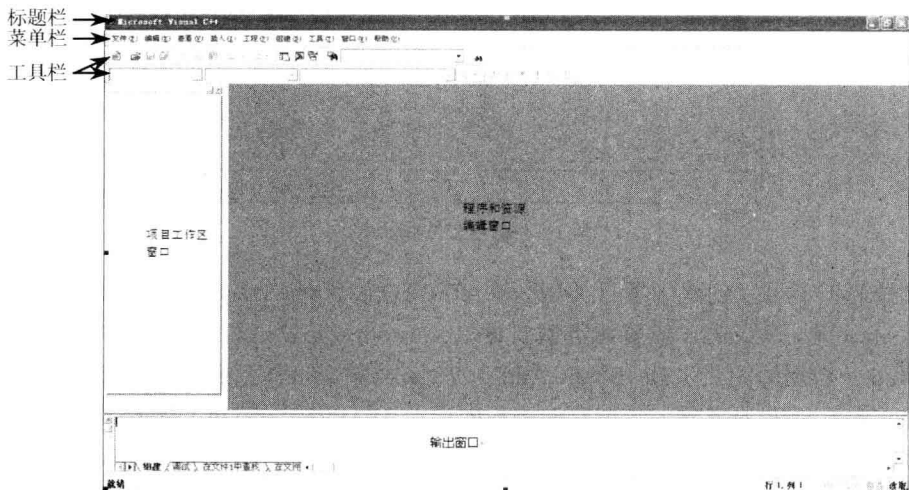


图 1.5 Visual C++6.0 窗口

1.3 Visual C++6.0 的功能介绍

在使用 Visual C++6.0 时, 大部分功能是通过菜单栏和工具栏按钮完成的, 下面对 Visual C++6.0 菜单栏和工具栏按钮做一简介。

1.3.1 Visual C++6.0 的菜单栏

Visual C++6.0 的菜单按功能分为几大菜单栏，它们分别是文件、编辑、查看、插入、工程、组建、工具、窗口和帮助。

1. 文件 (File)

这个菜单栏是专门负责对源文件、资源文件和工程文件管理的菜单，主要完成文件的创建、保存、打开、关闭以及打印等工作。常见的命令如下。

(1) **新建 (New)**。新建命令可以创建菜单文件 (Files)、工程项目 (Projects)、工作区 (Workspaces) 和其他文档 (Other documents)。

其中，文件选项卡中可以创建的文件类型为：Active Server Page (asp 文件)、Binary File (二进制文件)、Bitmap File (位图文件)、C/C++ Header File (C/C++头文件)、C++ Source File (C++源文件，在 Visual C++6.0 中，C 语言源文件也同时是 C++源文件，C++语法兼容 C 语法)、Cursor File (光标文件)、HTML Page (HTML 文件)、Icon File (图标文件)、Macro File (宏文件)、SQL Script File (SQL 脚本文件)、Resource Script (资源脚本文件)、Resource Template (资源模板文件)、Text File (文本文件) 等，如图 1.6 所示。



图 1.6 新建文件选项卡

工程项目选项卡可以创建的项目类型为：ATL COM AppWizard (ATL 应用程序向导)、Cluster Resource Type Wizard (簇资源类型向导)、Custom AppWizard (自定义应用程序向导)、Database Project (数据库项目)、DevStudio Add-in Wizard (集成工作室插件向导)、Extended Stored Proc Wizard (基于 SQL 服务器的扩展存储进程向导)、ISAPI Extension Wizard (ISAPI 服务器扩展动态链接库向导)、Makefile (旧版 C/C++生成文件)、MFC ActiveX Control Wizard (ActiveX 控件程序向导)、MFC AppWizard(dll) (MFC 动态链接库)、MFC AppWizard(exe) (MFC 可执行程序)、New Database Wizard (SQL 服务器数据库向导)、Utility Project (公共服务基础项目)、Win32 Application (Win32 应用程序)、Win32 Console Application (Win32 控制台应用程序，C 语言编写应用程序基本上都是控制台应用程序，因此我们建立新的 C 语言工程时都是选择这个项目类型)、Win32 Dynamic-Link Library (Win32 动态链接库)、Win32 Static Library (Win32 静态库)，如图 1.7 所示。

工作区选项卡可以建立各种类型的工作区，其他文档选项卡可以创建各种 Office 文件。

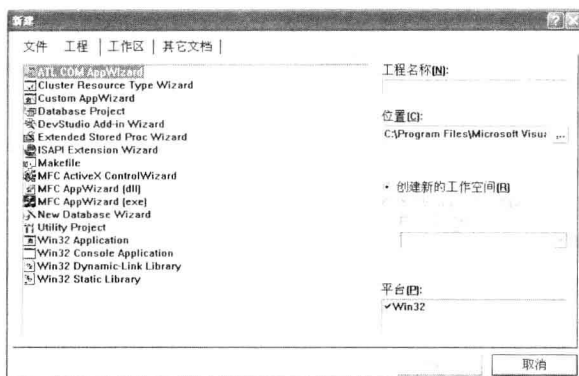


图 1.7 新建工程选项卡

(2) **打开 (Open)**。选择打开命令可以打开已经保存的文件。可以打开的文件类型有 C++ 文件、Web 文件、宏文件、资源文件、定义文件、图像文件、工作区文件和项目文件等。具体的在打开对话框中文件类型选择框中可以查看。

(3) **关闭 (Close)**。关闭活动窗口中打开的文件。

(4) **打开工作区 (Open Workspace)**。打开工作区管理的各种文件。

(5) **保存工作区 (Save Workspace)**。保存工作区管理的各种文件的修改。

(6) **关闭工作区 (Close Workspace)**。将会关闭当前工作区管理的各种文件。

(7) **保存文件 (Save)**。保存当前正在编辑的文件，如果保存的文件是第一次编辑，则会打开另存为对话框。

(8) **另存为文件 (Save As)**。将当前文件保存为另一个文件，原来的文件不会删除，也不做任何修改，新修改的内容只会保存到新文件中。

(9) **保存所有文件 (Save All)**。将所有打开的文件都保存，如果有文件是第一次保存，则会打开另存为对话框。

(10) **页面设置 (Page Setup)**。该命令可以设置打印页面的格式，如页面的上下左右边距和页眉页脚等内容。

(11) **最近打开的文件列表 (Recent Files)**。这里显示上次关闭 Visual C++ 集成开发环境前所打开的文件。该列表只显示最近打开的 4 个文件。

(12) **最近打开的工作区列表 (Recent Workspace)**。这里显示上次关闭 Visual C++ 集成开发环境前所有打开的工作区。该列表只显示最近打开的 4 个工作区列表。

2. 编辑 (Edit)

撤销 (Undo)：撤销最后一次的操作。

重做 (Redo)：重做最后一次撤销的操作。

剪切 (Cut)：将选中的内容删除，并在内存的剪贴板中复制一份。

复制 (Copy)：将选中的内容复制一份到剪贴板中，选中的内容不删除。

粘贴 (Paste)：将剪贴板中的内容复制到编辑区中光标所在处，如果剪贴板中没有内容，则该按钮是灰色的。

删除 (Delete)：删除选中内容。

全选 (Select All)：选择所有内容。

查找 (Find)：在当前文件中查找指定内容。查找是一个很重要的功能，在这里对查找对话框

做一个介绍。对话框如图 1.8 所示, Find what 中输入的是要查找的内容, Direction 是查找的方向, Up 是指从下向上查找, Down 是指从上向下找。

Match whole word only: 只匹配整个单词。

Match case: 区分大小写匹配。

Regular expression: 按照正则表达式规则匹配文本。

Search all open documents: 在所有打开的文档中查找。

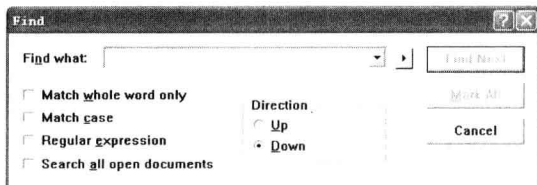


图 1.8 Find 对话框

Regular expression: 按照正则表达式匹配文本, 是指用特殊的字符序列去匹配文本字符串的模式, 通常称这些特殊字符为通配符, 表 1.1 列出了部分常用通配符及其含义。

表 1.1 Find what 对话框中正则式常用的通配符

通 配 符	含 义
.	匹配任何单个字符包括空格。例如, 正则表达式 “r.t” 匹配这些字符串: “rat”、“rut”、“rt”, 但是不匹配 “root”
\$	匹配行结束符。例如, 正则表达式 “weasel\$” 能够匹配字符串 “He's a weasel”, 但是不能匹配字符串 “They are a bunch of weasels”
^	匹配一行的开始。例如, 正则表达式 “^When in” 能够匹配字符串 “When in the course of human events”, 但是不能匹配 “What and When in the”
*	匹配 0 个或多个正好在它之前的那个字符。例如, 正则表达式 “*”, 意味着能够匹配任意数量的字符串
\	这是引用符, 用来将通配符当作普通的字符来进行匹配。例如, 正则表达式 “\\$” 被用来匹配美元符号, 而不是行尾, 类似的正则表达式 “\.” 用来匹配点字符, 而不是任何字符的通配符
[] [c1-c2] [^c1-c2]	匹配括号中的任何一个字符。例如, 正则表达式 “r[au]t” 匹配 “rat”、“rot” 和 “rut”, 但是不匹配 “ret”。可以在括号中使用连字符 “-” 来指定字符的区间, 例如, 正则表达式 “[0-9]” 可以匹配任何数字字符; 还可以指定多个区间; 例如, 正则表达式 “[A-Za-z]” 可以匹配任何大小写字母。另一个重要的用法是 “排除”, 要想匹配除了指定区间之外的字符, 也就是所谓的补集, 可以在左边的括号和第一个字符之间使用 “^” 字符, 例如, 正则表达式 “[^269A-Z]” 将匹配除了 2、6、9 和所有大写字母之外的任何字符
\(\)	将 “(” 和 “)” 之间的表达式定义为 “组” (group), 并且将匹配这个表达式的字符保存到一个临时区域 (一个正则表达式中最多可以保存 9 个), 它们可以用 “\1” ~ “\9” 的符号来引用
+	匹配 1 个或多个正好在它之前的那个字符。例如, 正则表达式 “9+” 匹配 “9”、“99”、“999” 等。注意: 这个元字符并不是所有的软件都支持的

Find in files: 在给定目录、给定类型的所有文件中查找指定的内容。

Replace: 在指定的文件中替换查找到的内容。

Go to: 光标移到指定的位置。

Breakpoints: 在指定的位置设置断点。

3. 查看菜单 (View)

查看菜单用来设置和改变窗口和工具栏的工作方式, 可以设置窗口按全屏显示, 打开工作区窗口, 打开信息输出窗口和各种调试窗口等。

4. 插入菜单 (Insert)

插入菜单主要用于项目及资源的创建和添加, 可以将文本插入文件中, 也可以插入一个新的 ATL 对象, 主要功能如下:

New class: 插入新类;

New Form: 新建窗体;

Resource: 新建资源;

Resource copy: 对选定的资源备份;

File As Text: 插入文本;

New ATL Object: 插入新的 ATL 对象。

本菜单在学习 C++ 面向对象开发用得较多, 对于 C 语言程序设计, 基本上不怎么用。

5. 项目菜单 (Project)

管理项目和工作区, 所谓项目是指一些彼此相关联的源文件, 经过编译、链接后产生为一个可执行文件的 Windows 程序或者是动态链接库函数。该菜单可以把选定的项目指定为工作区中的活动项目, 可以把一些文件、文件夹、数据链接以及可再用部件添加到项目中, 也可以编辑或修改项目间的依赖关系。

6. 编译菜单 (Build)

编译菜单包括用于编译、建立和执行应用程序的命令。主要的命令介绍以下。

Compile: 编译源文件, 在编译的时候能判断源程序的语法错误。在编译过程中出现的语法错误或警告会在信息输出窗口显示。可以向前或者向后浏览错误信息, 通过点击<F4>键会在源文件中显示错误的相关代码行。

Build: 构建项目中的所有文件。如果在构建项目过程中出现了错误或者警告信息都会在信息输出窗口显示。如果文件没有编译过, 菜单先编译文件, 然后构建生成可执行文件或动态链接库。所谓构建是指把编译的二进制文件、程序中引用的库函数或其他用户自定义的函数库函数链接在一起形成可执行文件或新的动态链接函数库。

Rebuild all: 重新构建所有的源文件。

Batch build: 批构建文件, 可以指定构建 Release 版, 或者 Debug 版的, 或者两者都构建。Release 版本是软件的发布版本, 没有调试信息, 最终生成的可执行文件或动态链接库是优化过的, 代码长度一般较小; Debug 版本是调试版本, 编译的二进制可执行文件或动态链接函数库代码中含有调试信息, 代码没有优化, 长度比较长。

Clean: 清除构建的文件。

Start debug: 该菜单项下有几个子菜单, 都是用于调试, 分别如下。

Go: 执行程序。

Step into: 单步调试, 碰到函数调用, 则进入函数体。

Run to cursor: 执行到光标处。

Step over: 单步调试时, 碰到函数调用, 跳过函数体。

Step out: 该命令和 Step into 配合使用。如果使用 Step into 在调试某一函数体时, 发现该函数体不需要调试, 可以使用 Step out 退出来。

Profile: 该命令是用于检查程序运行行为的强有力的工具。它不是为了检查程序的错误, 而是为了使程序更好地运行。

7. Tools 菜单

用于选择或定制开发环境中的一些实用工具、打开一些调试窗口、改变窗口的显示模式等。

1.3.2 Visual C++工具栏

工具栏是一系列的命令组合, 他们以图形的方式显示在屏幕上, 是一种直观快捷的方式。使用 Visual C++6.0 系统提供的操作命令, 熟悉工具栏按钮, 可以提高使用 Visual C++6.0 的开发效率。下面就对一些常用的工具栏做介绍。

1. 标准工具栏 (Standard)

标准工具栏如图 1.9 所示, 其按钮说明如表 1.2 所示。

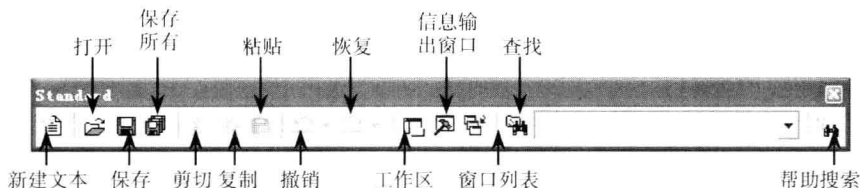


图 1.9 Visual C++6.0 的工具栏

表 1.2

标准工具栏按钮说明

按钮命令	含 义
新建文件	新建一个文本文件
打开	打开已经存在的文件
保存	保存当前活动文件
保存所有	保存所有打开的文件
剪切	将选中的内容删除掉, 并复制到剪贴板中
复制	将选中的内容复制到剪贴板中
粘贴	将剪贴板中的内容复制到指定的位置
撤销	撤销上次的操作, 点击旁边的小三角, 可以直接撤销已做过的指定步骤
恢复	恢复刚刚撤销的步骤, 点击旁边的小三角, 可以直接恢复指定的步骤
工作区	显示或隐藏工作区窗口
信息输出窗口	显示或隐藏信息输出窗口
窗口列表	显示已打开的窗口列表
查找	在文件中查找指定的内容
帮助搜索	在当前文件中查找指定的字符串

2. 向导工具栏

向导工具栏如图 1.10 所示，其按钮说明见表 1.3。



图 1.10 向导工具栏

表 1.3 向导工具栏按钮说明

按钮命令	功能描述
类	显示当前编辑的类。通过此下拉列表可以迅速的定位到指定的类
过滤器	显示正在操作的资源标示
成员函数	显示当前正在操作的成员函数名，和前两个配合，可以快速的定位到指定的函数中
功能按钮	帮助快速找到当前编译的代码的相关位置，如果当前编辑的是成员函数，通过此按钮可以快速的定位到类的定义处或成员函数的声明处

3. 小型构建工具栏

小型构建工具栏如图 1.11 所示，其按钮说明如表 1.4 所示。

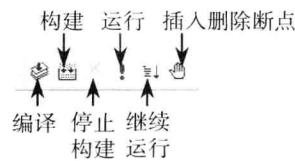


图 1.11 小型构建工具栏

表 1.4 小型构建工具栏按钮说明

按钮命令	含 义
编译	编译 C 或 C++源文件
构建	从项目中构建出应用程序的 exe 文件
停止构建	在构建过程中按该按钮可以停止构建项目
运行	执行应用程序，如果程序没有构建，则先构建程序，再执行
继续运行	单步执行
插入删除断点	插入或删除断点

1.4 Visual C++6.0 编写 C 语言程序过程

编写 C 语言程序的第一步是建立 C 语言源程序。在以往的 Turbo C 环境下，C 源程序的扩展名是.c，而在 Visual C++6.0 环境下建立的 C 源程序扩展名默认是.cpp，Visual C++6.0 也能识别.c 的源文件。不论是.c 文件还是.cpp 文件，他们都是字符文件，所以建立源程序的方法就有很多了，可以使用任何编写字符文件的工具，比如常用的记事本就可以用来编写 C 语言源程序，只是文件在保存时，一定要将扩展名改为.c 或.cpp 而不是.txt。

也可以用记事本帮助编写 C 语言源程序,然后在 Visual C++6.0 环境下面编译链接。但是 Visual C++6.0 提供了更好的编写 C 语言源程序的环境。AppWizard 能帮助用户迅速地生成应用程序框架,如 Windows 应用程序、控制台应用程序等。在本书中编写的程序都是控制台类程序,所谓控制台类程序是指运行在 DOS 环境下的程序,这类程序一般没有很好的用户界面。

下面介绍利用 Visual C++6.0 建立一个简单的 C 语言源程序步骤。本程序就是在控制台输出一个“hello world!”。

1.4.1 编写 C 语言源码程序

准备工作:先在 d 盘的根目录下面建立一个名为“MyProject”的文件夹。

(1) 打开 Visual C++6.0,选择“文件”(File)菜单,在其中选择“新建”(New)子菜单项。

(2) 在新建对话框中选择“项目”(Project)属性卡。在该属性页选择“Win32 Console Application”,如图 1.12 所示。

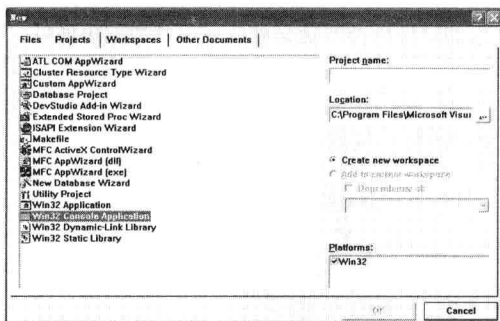


图 1.12 新建项目对话框

(3) 在“Project name”文本框中输入控制台应用程序的工程文件名,如“helloworld”。在“Location”文本框中选择工程保存的路径,在这里选择刚刚建立的文件夹路径。如果是第一次使用,则在“Location”文本框中会出现默认的工程项目路径,可以点击“Location”文本框右边的 3 个小黑点的按钮,会出现工程路径选择对话框。在那里可以修改默认的工程路径,选择“d:\MyProject”。单击“OK”按钮。

(4) 这时进入到“Win32 Console Application Step 1 of 1”对话框,如图 1.13 所示,选中“An empty project”项。

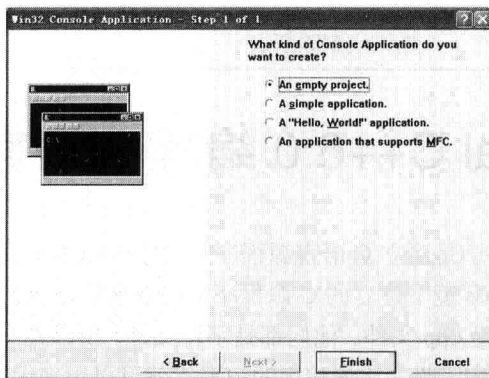


图 1.13 Win32 Console Application Step 1 of 1 对话框

(5) 单击“Finish”按钮，系统将出现 AppWizard 的创建信息，直接单击“OK”按钮。系统会建立所要求的应用程序框架。

(6) 如图 1.14 所示，在该窗口的项目工作区窗口中，可以看到“ClassView”和“FileView”两个属性标签。本书中使用的都是“FileView”属性标签。

(7) 在“FileView”属性标签下，选中“Source Files”，然后在菜单“File”下选择“New”，出现如下“New”对话框，如图 1.15 所示。



图 1.14 工作区对话框

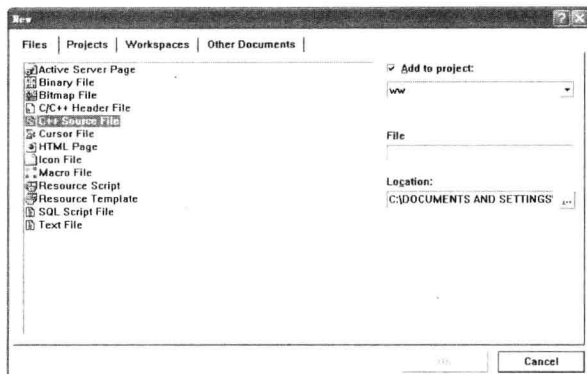


图 1.15 “New”对话框

在“Files”属性列中选择“C++ Source File”，在 File 文本框中输入“Helloworld.cpp”，在“Location”文本框中输入文件保存的位置，一般选择默认位置，不做修改，然后点击“OK”按钮。

(8) 在“FileView”属性列中选择“Source Files”，打开“Helloworld.cpp”，在“Helloworld.cpp”文件中输入下列代码：

```
#include "stdio.h"
void main()
{
    printf("hello world!\n");
}
```

然后保存项目，这样，一个 C 语言源程序就编好了。

1.4.2 编译源程序，构建应用程序

按下<Ctrl+F7>键或者选择构建菜单中的编译（Compile）helloworld.cpp 命令，系统开始对 helloworld.cpp 进行编译。或者按下<F7>键或选择构建菜单中的构建（Build）helloworld.exe 命令，系统开始对 helloworld 进行编译、链接。在编译结束时会发现在信息输出窗口中出现如下信息：

```
-----Configuration: helloworld - Win32 Debug-----

Compiling...
Helloworld.cpp
D: \MYPROJECT\helloworld\helloworld.cpp(4) : error C2001: newline in constant
D: \MYPROJECT\helloworld\helloworld.cpp(5) : error C2143: syntax error : missing ')'
before '}'D: \MYPROJECT\helloworld\helloworld.cpp(5) : error C2143: syntax error : missing
';' before '}'Error executing cl.exe.

helloworld.obj - 3 error(s), 0 warning(s)
```

这些信息的含义如下。

(1) 配置: helloworld-win32 debug。表示现在编译的配置构建输出结果是 Win32 debug 输出模式。配置应用程序构建输出的模式有两种, Debug 模式和 Release 模式, 其中 Debug 模式是调试模式, Release 模式是发布模式, 在 Debug 模式中代码并没有优化, 最终代码行和源代码行具有一定的对应关系, 而 Release 模式, 最终代码经过了优化, 产生的最终代码的长度、执行时间和空间都是最优的。在开始编程时, 一般采用 Debug 模式生成目标代码, 当测试项目无误, 准备发布时就改为 Release 模式。

(2) 编译开始, 准备编译 helloworld.cpp。

(3) d:\myproject\helloworld.cpp 的第 4 行有错误, 代号为 C2001: 在常量中有新行, 第 5 行有错误, 错误代码 C4508: 语法错误: 在 “}” 前少了 “);”;

(4) 编译 helloworld.obj 出现 3 个错误, 0 个警告。

出现了错误, 下面就改正错误。从错误出现的最开始行 4 行开始, 检查代码, 发现第 4 行为: printf(“hello world!\n);这句, 在这句的括号内少了一个 ‘)’’, 加上这个引号, 继续看代码, 没有错误, 改正代码后保存, 再次编译。输出窗口中出现如下信息:

```
-----Configuration: helloworld - Win32 Debug-----
Compiling...
Helloworld.cpp
helloworld.obj - 0 error(s), 0 warning(s)
```

看最后一行 helloworld.obj-0 错误, 0 警告。说明代码语法没有任何问题。

然后构建应用程序, 点<F7>键或者在构建菜单中选择构建 (Build) helloworld.exe 命令, 信息输出窗口出现:

```
-----Configuration: helloworld - Win32 Debug-----
helloworld.exe - 0 error(s), 0 warning(s)
```

表示构建应用程序成功没有错误, 没有警告。

按<Ctrl+F5>键运行程序出现如图 1.16 所示结果。

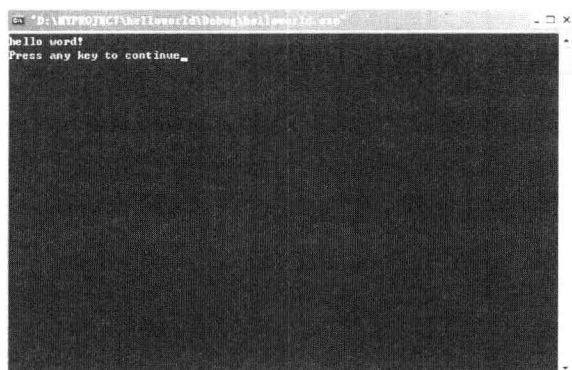


图 1.16 程序运行结果

表示程序输出的结果也符合要求, 至此有关在 Visual C++6.0 下编写、编译、调试和链接 C 语言程序的基本步骤就结束了。