21世纪高等学校计算机规划教材

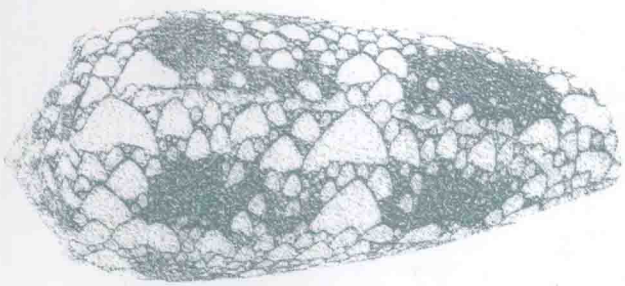
21st Century University Planned Textbooks of Computer Science

C语言程序设计项目教程

C Language Programming Project Tutorial

段善荣 厉阳春 钱涛 陈博 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

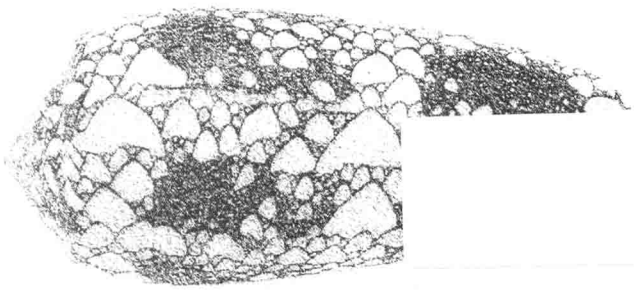
21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

C语言程序设计项目教程

C Language Programming Project Tutorial

段善荣 厉阳春 钱涛 陈博 主编



高校系列

人 民 邮 电 出 版 社

北 京

图书在版编目（CIP）数据

C语言程序设计项目教程 / 段善荣等主编. -- 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
ISBN 978-7-115-30068-3

Ⅰ. ①C… Ⅱ. ①段… 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(2013)第024321号

内 容 提 要

本书共分四部分 13 章，其中第一部分基础篇由 C 语言概述，数据类型、运算符和表达式，顺序结构、选择结构、循环结构程序设计组成；第二部分提高篇由数组、结构体和共用体、函数及编译预处理组成；第三部分高级篇由指针、链表、文件组成；第四部分扩展篇由算法与数据结构、软件开发基础知识组成。全书结合“学生成绩管理系统”这个典型项目讲解了 C 语言所有知识点，使读者能够较快地掌握 C 语言程序设计的基础知识、基本算法和编程思想，同时还提供了内容丰富的、趣味性较强的案例，能有效提高读者的学习兴趣。

本书既可以作为非计算机专业本科学生的计算机 C 语言教材，也可以作为高等院校计算机专业本科和专科学生的基础教材，还可以作为自学者和教师的参考教材。

21 世纪高等学校计算机规划教材

C 语言程序设计项目教程

◆ 主　　编　段善荣　厉阳春　钱　涛　陈　博
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
北京隆昌伟业印刷有限公司印刷

◆ 开本：787×1092　1/16
印张：24.75　　2013 年 3 月第 1 版
字数：657 千字　　2013 年 3 月北京第 1 次印刷

ISBN 978-7-115-30068-3

定价：45.00 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 语言是目前较流行的程序设计语言之一。它既具有高级语言程序设计的特点，又具有汇编语言的功能，是当今世界上最具有影响力的程序设计语言之一，也是程序设计者应当熟练掌握的一种语言工具。

本书结合《全国计算机等级考试二级 C 语言考试大纲》，并结合作者多年来的教学经验和软件开发实践，对 C 语言知识点的编排进行了细致的策划和组织，精心选择和设计了趣味性和实用性较强的案例，能有效提高读者的学习兴趣，激发读者的求知欲望。通过一个典型项目“学生成绩管理系统”将分散的知识点进行有机联系，并由浅入深地应用每章所涉及的知识点，强调了知识的层次性和技能培养的渐进性，学习者可以借鉴项目中的经验，最终用于开发其他项目，真正达到学以致用的目的。每章均附有习题，有利于巩固和提高学习者的学习水平。

本书由段善荣和厉阳春主编，第 1 章～第 5 章由钱涛编写，第 6 章～第 8 章由段善荣编写，第 9 章～第 11 章由陈博编写，第 12 章、第 13 章、附录由厉阳春编写，全书由段善荣统稿和主审。

由于编者水平有限，书中难免出现疏漏或处理不当之处，恳请读者批评指正。

编　者

2012 年 12 月

目录

第 1 章 C 语言概述

电子计算机自从诞生以来，在人类的各个领域都取得了丰硕的成果。但计算机本身是无生命的，要使它能够运行起来，为人类完成各种各样的工作，就必须让它执行相应的程序，这些程序都是依靠程序设计语言编写出来的。

C语言就是众多程序设计语言中的一种，是国际上广泛流行的、很有发展前途的计算机高级语言。它具备方便性、灵活性和通用性等特点，同时还向程序员提供了直接操作计算机硬件的功能，既可用来写系统软件，也可以用来写应用软件，深受软件工作者欢迎。

本章从C语言的发展入手，从程序设计的角度，介绍了C语言程序的基本结构和开发环境等内容。

1.1 C 语言的发展及主要特点

本节主要介绍 C 语言的发展史、主要特点及 C 语言程序的基本结构。

1.1.1 C 语言的发展史

C语言是当今社会应用广泛的、并受到众多程序开发人员欢迎的一种计算机算法语言。C语言的出现是与 UNIX 操作系统紧密联系在一起的。

从历史发展来看，C语言起源于 1968 年发表的 CPL 语言(Combined Programming Language)，它的许多重要思想来自于 Martin Richards 在 1969 年研制的 BCPL 语言，以及以 BCPL 语言为基础的由 Ken Thompson 在 1970 年研制成的B语言。Ken Thompson 用B语言写了第一个 UNIX 操作系统，用在 PDP-7 计算机上。D.M.Ritchie 1972 年在B语言的基础上研制了C语言，并用C语言写成了第一个在 PDP-11 计算机上实现的 UNIX 操作系统。1977 年出现了独立于机器的C语言编译文本《可移植C语言编译程序》，从而大大简化了把C语言编译程序移植到新环境所需做的工作，使 UNIX 操作系统迅速地在众多的机器上实现。随着 UNIX 操作系统的广泛使用，C语言也迅速得到了推广。

1983 年美国国家标准化协会（ANSI）根据 C 语言问世以来的各种版本，对 C 语言的发展和扩充制定了新的标准，称为 ANSI C。1987 年 ANSI 又公布了新的标准——87 ANSI C。1990 年，国际标准化组织（ISO）接受 87 ANSI C 为 ISO C 的标准。目前流行的 C 编译系统都是以它为基础的。

目前在微型机上使用的有 Quick C、Turbo C、BORLAND C、Visual C++等多种版本。这些不

同的 C 语言版本，基本部分是相同的，但在有关规定上又略有差异。本书结合全国计算机等级考试二级（C 语言）的要求，在 Visual C++的环境下运行 C 语言程序，也使读者在继续学习时，能更快地熟悉面向对象程序设计的环境。

1.1.2 C 语言的主要特点

任何一种程序设计语言，都有其特点和主要的应用领域。在有众多程序设计语言存在的环境中，一种语言之所以能存在和发展，并具有生命力，总是有其不同于（或优于）其他语言的特点。事实证明，C 语言是一种极具生命力的语言，它的特点是多方面的，主要特点归纳如下。

（1）语言简洁、紧凑，使用方便、灵活。

C 语言一共只有 32 个关键字，9 种控制语句，程序书写形式自由，主要用小写字母表示，压缩了一切不必要的成分，使程序设计人员在输入源程序时，尽可能地减少工作量。

（2）C 语言运算符丰富。

C语言运算符包含的范围很广泛，共有 34 种运算符。C语言把括号、赋值、强制类型转换等都作为运算符处理。从而使C语言的运算类型极其丰富，表达式类型多样化，灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

（3）C 语言数据结构丰富，具有现代化语言的各种数据结构。

C语言的数据类型有：整型、浮点型、字符型、数组类型、指针类型、结构体类型、共用体类型等，能用来实现各种复杂的数据结构（如链表、树、栈等）的运算，尤其是指针类型数据，使用起来比其他语言更为灵活、多样。

（4）C 语言具有结构语言的特点。

它具有结构化的流程控制语句（如 if-else 语句、while 语句、do-while 语句、for 语句），支持若干种循环结构，允许编程者采用缩进书写形式编程。因此，用 C 语言设计出的程序层次结构清晰。

（5）C 语言程序的基本单位是函数，函数可以在程序中完成独立的任务，独立地编译成代码，以实现程序的模块化，符合现代编程风格要求，并且程序之间很容易实现共享。

（6）语法限制不太严格，程序设计自由度大。

例如，对数组下标越界不做检查，由程序编写者自己保证程序的正确。对变量的类型使用比较灵活。例如，整型数据、字符型数据和逻辑型数据可以通用，一般的高级语言语法检查比较严，能检查出几乎所有的语法错误。而C语言允许程序编写者有较大的自由度，因此放宽了语法检查。这样使 C 语言能够减少对程序员的束缚。程序员应当仔细检查程序，保证其正确，而不要过分依赖C编译程序去查错。“限制”与“灵活”是一对矛盾体，限制严格，就失去灵活性；而强调灵活，就必然放松限制。一个不熟练的人员，编一个正确的C程序可能会比编一个其他高级语言程序难一些。从这个角度来说，对用C语言的人，要求其对程序设计更熟练一些。

（7）C语言允许直接访问物理地址。

C 语言可直接对硬件进行操作，实现汇编语言的大部分功能，因此C语言既具有高级语言的功能，又具有低级语言的许多功能。C语言的这种双重性，使它既是成功的系统描述语言，又是通用的程序设计语言。有人把C称为“高级语言中的低级语言”，也有人称它为“中级语言”，意为兼有高级和低级语言的特点。

（8）生成目标代码质量高，程序执行效率高。

一般只比汇编程序生成的目标代码效率低 10%~20%。

（9）用C语言写的程序可移植性好（与汇编语言比）。

基本上不做修改就能用于各种型号的计算机和操作系统。

上面只介绍了C语言的最容易理解的一般特点，至于C语言内部的其他特点将结合以后各章的内容做介绍。由于C语言的这些优点，使C语言应用面很广。许多软件都用C语言编写，这主要是由于C语言的可移植性好、硬件控制能力高、表达和运算能力强。许多以前只能用汇编语言处理的问题现在可以改用C语言来处理了。

C 语言优点很多，但是它也存在一些缺点，如运算符优先级太多，数值运算能力方面不像其他的一些语言那样强，语法定义不严格等。尽管 C 语言目前还存在一些不足之处，但由于它目标代码质量高、使用灵活、数据类型丰富、可移植性好而得到广泛的普及和迅速的发展，成为一种受到广大用户欢迎的实用的程序设计语言，同时也是一种在系统软件开发、科学计算、自动控制等各个领域被广泛应用的程序设计语言。

1.1.3　C 语言程序的基本结构

在学习 C 语言的具体语法之前，我们先通过两个简单的 C 语言程序示例，来初步了解 C 语言程序的基本结构。

【例 1.1】有两个瓶子 A 和 B，分别装着水和酒，要求将两个瓶子中的液体交换。

【问题分析】

上例可以抽象为将两个数 a 和 b 的值交换。

【程序代码】

```
#include <stdio.h>
void main()                          /*主函数*/
{
    int a,b,c ;
    printf("请输入两个整数：");
    scanf("%d%d",&a,&b);             /*输入两个数*/
    /*下面三行语句实现两个数交换*/
    c=a;
    a=b;
    b=c;
    printf("a=%d,b=%d\n",a,b);       /*输出交换后的结果*/}
}
```

【运行结果】（见图 1.1）

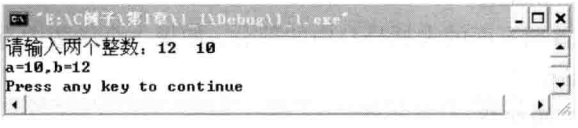


图 1.1 【例 1.1】运行结果

【例 1.2】输入 3 个整数，求最大的数。

【程序代码】

```
#include <stdio.h>
int max(int x,int y)  /*定义函数，用来求两个数当中较大的一个数*/
{
```

```
    int  z;
    if (x>y)  z=x;
    else  z=y;
    return (z);   /*将 z 的值返回*/
}
void main()                          /*主函数*/
{
    int a,b,c,p;                     /*声明部分，定义变量*/
    printf("请输入三个整数：");
    scanf("%d%d%d",&a,&b,&c);        /*输入 3 个整数*/
    p=max(a,max(b,c));               /*调用 max 函数，得到最大值赋给 p*/
    printf("最大值为%d\n",p);        /*输出 p 的值*/
}
```

【运行结果】（见图 1.2）

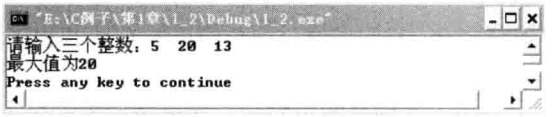


图 1.2 【例 1.2】运行结果

上面两个例子虽然都比较简单，但我们可以从中看出 C 语言程序的基本结构和书写格式。

（1）C 语言程序的基本单位是函数。

一个 C 语言源程序可以由一个或多个源文件组成，每个源文件可以由一个或多个函数组成，每个函数可以由一条或多条语句组成。程序是运行单位，文件是编译单位，每个文件可以单独编译，而函数是 C 语言的基本单位。每一个 C 语言程序都必须有且仅有一个 main()函数，称为主函数。main()函数可以在程序的开头，也可以在其他位置，但不能包含在其他函数内部。它的作用是标识整个程序开始执行的位置和程序运行结束的位置。其他函数是为了实现程序的功能而设置的小模块，C 语言的这种结构符合现代程序设计中模块化的要求。C 语言的函数除了标准库函数外，用户还可以根据需要自己定义函数，如上面例 1.2 中的 max 函数。

（2）一个 C 语言函数由以下两部分组成。

① 函数的首部，即函数的第一行。包括函数名、函数的类型、函数属性、函数参数（形参）名、参数类型。上面例子中，第一个 int 为函数的类型，max 为函数的名称，括号内的 x、y 为函数的参数（形参），用来定义 x、y 的 int 为参数的类型。一个函数名后面必须跟一对小括号（），函数的参数可以没有，如 main()。

② 函数体，即函数首部下面的大括号{}内的部分，如果一个函数内有多个大括号，则最外一层的一对大括号为函数体的范围。

函数体一般包括以下两部分。

a．声明部分：在这部分中定义本函数内部所用到的一些变量，如例 1.2 中的 max()函数中的 int z;，main()函数中的 int a,b,c,p;。

b．执行部分：由若干条语句组成，是函数功能的实现过程的描述。

在某些情况下，函数可以没有声明部分。甚至可以既无声明部分，也无执行部分。这样的函数称为空函数。

（3）C 程序书写格式自由，一行内可以写几个语句，一个语句可以分写在多行上。C 程序没

有行号。

（4）每个语句和数据定义的最后都必须有一个分号。分号是 C 语句的必要组成部分，是构成语句所必不可少的，即使是程序中最后一个语句也应包含分号。

（5）“/*”与“*/”之间的内容构成 C 语言程序的注释部分。

“/*”和“*/”之间的内容可以是一行，也可以是多行。注释部分不参与程序的编译和运行，只是起到说明作用，增强程序的可读性。一个好的、有使用价值的源程序都应当加上必要的注释。

了解了 C 语言程序的基本结构之后，在进行程序设计编写时，就可以先进行各个功能函数的编写，然后通过主函数的调用或函数与函数之间的调用，将整个程序组装起来，实现完整的、复杂的功能。因此，在编写程序时，读者首先不要有畏惧感，任何一个大程序都可以把它划分成大小不等的各种功能，每个功能由一个函数来实现，再将这些函数通过合理的方式组合起来，就构成了一个大型的程序。一个较完整的 C 语言程序大致包括：头文件、用户函数说明部分、全局变量定义、主函数、若干用户自己定义编写的函数。

1.2　C 语言上机过程

编写好的 C 语言程序要经过编辑（输入）、编译和链接后才能形成可执行的程序。图 1.3 表示了 C 语言程序设计的上机步骤。

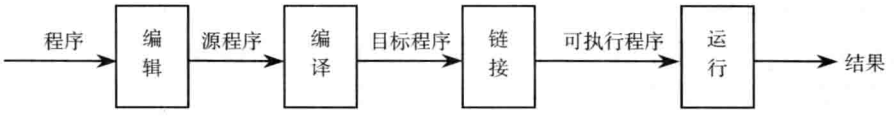


图 1.3　C 语言程序的上机步骤

C 语言的编译系统有多种，本书采用 Visual C++ 6.0 集成开发环境，本节介绍在此环境下如何编辑、编译、链接和运行 C 程序。

1.2.1　启动 VC++6.0

启动 Visual C++6.0 的操作步骤如下。

（1）单击任务栏中的开始菜单按钮，选择“程序”菜单项，选择该菜单项下的“Microsoft Visual Studio6.0”下的“Microsoft Visual C++6.0”，即可启动 Visual C++6.0。

（2）如果是第一次启动或出现 Tip of　Day 对话框，里面会显示一些使用 Visual C++6.0 的一些小技巧。点击“Next Tip”会显示下一条技巧，点击“Close”关闭小技巧对话框。如果不想下次启动 Visual C++6.0 时出现该对话框，则取消复选框“show Tips at startup”，则下次启动 Visual C++6.0 时就不会出现这个对话框。

（3）关闭 Tips of Day 对话框后，就进入了 Visual C++6.0 的主窗口，如图 1. 4 所示。主窗口主要由标题栏、菜单栏、工具栏、项目工作区窗口、程序和资源编辑窗口、信息输出窗口、状态栏等组成。

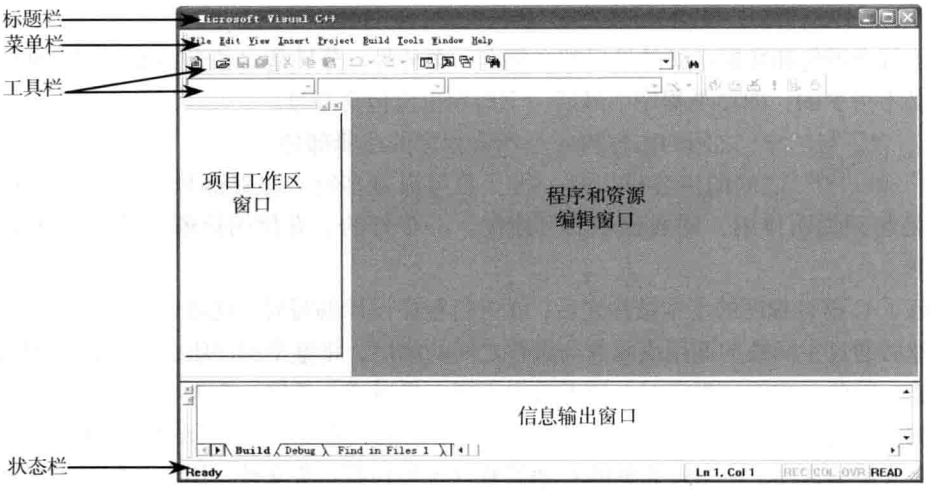


图 1.4　Visual C++6.0 主窗口

从图 1.4 可以看到，主窗体主要由 3 个窗体组成：项目工作区窗口、程序和资源编辑窗口、信息输出窗口。其中，项目工作区窗口和信息输出窗口是可停靠窗口，也就是说，这两个窗口可以放到主窗体的任意位置，将鼠标放到这两个窗口的停靠控制区，就可以拖动窗口到任意位置。

1.2.2　Visual C++6.0 的菜单栏

菜单是使用 Visual C++6.0 的主要操作方式，所以下面就逐一地介绍 Visual C++6.0 的各个菜单。

1. 文件菜单（File）

文件菜单是处理与文件操作相关的命令菜单，主要包括以下的菜单项。

New：提供新建文件、项目、工作区和其他文档功能。

Open：打开已存在的文件。

Close：关闭当前打开的活动文件。

Open Workspace：打开工作区文件。

Save Workspace：保存工作区文件。

Close Workspace：关闭工作。

Save：保存当前打开文件，如果该文件是第一次编辑，则会打开“Save As”对话框。

Save As：另存当前打开文件。

Save All：保存所有打开文件。

Page Setup：打印页设置。

Print：打印当前文件内容。

Recent Files：最近打开文件列表。

Recent Workspace：最近打开工作区列表。

Exit：退出系统。

2. 编辑菜单（Edit）

Undo：撤销上次操作。

Redo：重做上次操作。

Cut：剪切。

Copy：复制。

Parse：粘贴。

Delete：删除。

Select All：选择所有内容。

Find：在当前文件中查找指定内容。查找是一个很重要的功能，下面对查找对话框做一下介绍，如图 1.5 所示。

（1）Find what：输入要查找的内容。

Direction：查找的方向，向上是 Up，向下是 Down。

Match whole word only：只匹配整个单词。

Match case：区分大小写。

Regular expression：按照正则表达式匹配文本。

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

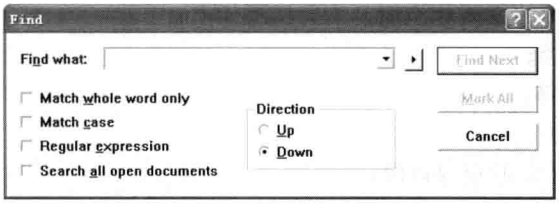


图 1.5　Find 对话窗口

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模式，通常称这些特殊字符为通配符，表 1.1 列出了这些通配符及其含义。

表 1.1　　通配符及其含义

通　配　符	含　　义
.	匹配任何单个字符。例如，正则表达式 r.t 匹配这些字符串：rat、rut、r t，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"的末尾，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” 的开始，但是不能匹配 “What and When in the” 。
*	匹配 0 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任何字符
\	这是引用府，用来将这里列出的这些元字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，类似地，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[aou]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；还可以制订多个区间，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：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

续表

通 配 符	含 义
+	匹配 1 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。

（2）查看菜单（View）

查看菜单可以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。

（3）插入菜单（Insert）

主要用于项目及资源的创建和添加，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。

（4）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，也可以把一些文件、文件夹、数据链接以及可重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。

（5）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。主要的命令如下。

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，通过双击或点击<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。

Rebuild all：重新构建所有的文件。

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。

Clean：清除构建的文件。

Start debug：该菜单项下有几个子菜单，都是用于调试时用的。分别是，Go：调试时，进入函数体；Step into：调试时，进入函数体；Run to cursor：执行到光标处；Step over：单步调试时，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。

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。

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。

1.2.3　Visual C++6.0 的工具栏

工具栏是一序列的命令组合，它们以图形的方式显示在屏幕上，如图 1.6 所示。工具栏是以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）

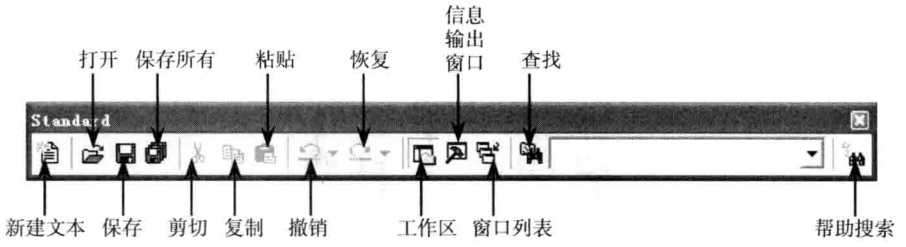


图 1.6　Visual C++6.0 的工具栏

表 1.2　　工具栏按钮命令及功能

按 钮 命 令	功 能 描 述
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 1.7 所示，其按钮命令及功能见表 1.3。

图 1.7　向导工具栏

表 1.3　　向导工具栏按钮命令及功能

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地定位到指定的类
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，可以快速地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地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 1.8 所示，其按钮命令功能见表 1.4。

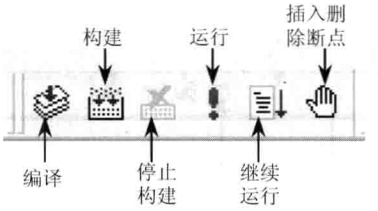


图 1.8　小型构建工具栏

表 1.4　　小型构建工具栏按钮命令及功能

按 钮 命 令	功 能 描 述
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

1.2.4　Visual C++6.0 编辑、编译、链接和运行程序的步骤

1. 建立源程序

编写 C 语言程序的第一步就是建立 C 语言源程序。在以往的 Turbo C 环境下，C 语言源程序的扩展名是.c，而在 Visual C++6.0 环境下面建立的 C 语言源程序的扩展名默认是.cpp，Visual C++6.0 也可以识别.c 的源文件。不论是.c 文件还是.cpp 文件，它们都是字符文件，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作编写 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。

虽然可以用记事本编写 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 语言源程序的方法。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程序，所谓控制台程序是指运行在 DOS 环境的程序，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 语言源程序的步骤。本程序是在控制台输出“hello world”。