

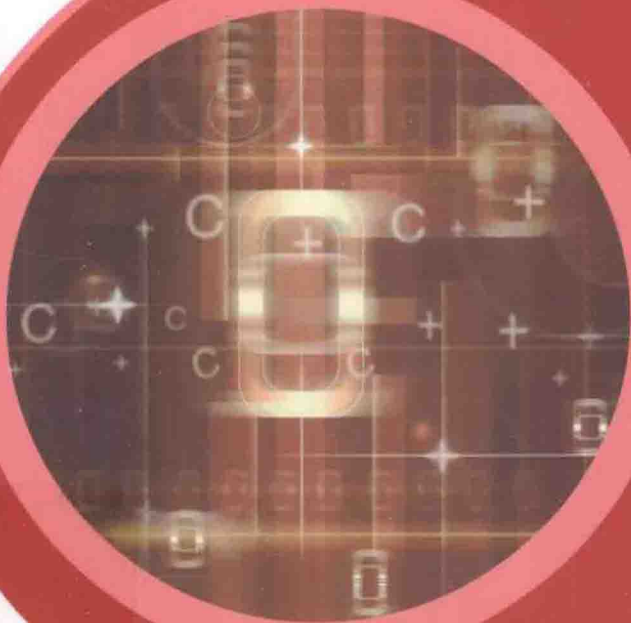
C YUYAN CHENGXU SHEJI ANLI YU SHIXUN JIAOCHENG



21 世纪高职高专规划教材
高等职业教育规划教材编委会专家审定

C语言程序设计 案例与实训教程

主 编 周 丰
副主编 黄 庆 熊 慧
主 审 杨海澜



北京邮电大学出版社
www.buptpress.com



21 世纪高职高专规划教材

高等职业教育规划教材编委会专家审定

C 语言程序设计案例与实训教程

主 编 周 丰
副主编 黄 庆 熊 慧
主 审 杨海澜



北京邮电大学出版社
www.buptpress.com

内 容 提 要

C 语言是一种应用非常广泛的高级程序设计语言,它既可以用于编写应用软件,又可以用于编写系统软件。

本书是针对高职高专教学特点而编写的 C 语言教学用书。本书以学习任务驱动,通过典型案例,系统地介绍了 C 语言的基本概念、语法规则及结构化程序设计的基本方法。本书特别注重技能训练和学习检测评价,强调培养学生掌握 C 语言的基本知识和解决编程过程中遇到的问题的动手能力,提高了学习效果和学习效率。

本书既注重理论又注重实践,内容丰富,概念清晰,深入浅出,特别适合作为高职院校 C 语言程序设计课程教材。本书兼顾了计算机二级考试的要求,可以作为计算机等级考试(二级 C)的参考书,也可以供从事计算机技术与应用工作的科技人员使用。

图书在版编目(CIP)数据

C 语言程序设计案例与实训教程 / 周丰主编. -- 北京:北京邮电大学出版社, 2014. 1
ISBN 978-7-5635-3746-4

I. ①C… II. ①周… III. ①C 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 261643 号

书 名: C 语言程序设计案例与实训教程

主 编: 周 丰

责任编辑: 王晓丹

出版发行: 北京邮电大学出版社

社 址: 北京市海淀区西土城路 10 号(邮编:100876)

发 行 部: 电话: 010-62282185 传真: 010-62283578

E-mail: publish@bupt. edu. cn

经 销: 各地新华书店

印 刷: 北京联兴华印刷厂

开 本: 787 mm×1 092 mm 1/16

印 张: 20.25

字 数: 532 千字

版 次: 2014 年 1 月第 1 版 2014 年 1 月第 1 次印刷

ISBN 978-7-5635-3746-4

定 价: 39.80 元

· 如有印装质量问题,请与北京邮电大学出版社发行部联系 ·

前 言

C 语言以其通用性和无限制性等特点成为当今软件开发的主流。C 语言是一种编译型程序设计语言,它兼顾了多种高级语言的特点,并具备汇编语言的功能。C 语言具有丰富的数据类型、灵活方便的表达式和独具特色的数据结构,可移植性好,兼具高级语言和低级语言的特点,既可用于编写系统软件,又可用于编写应用软件,其应用范围十分广泛。因此,C 语言是目前各类大专院校学习程序设计的首选语言。

为了配合 C 语言的教学,我们根据多年从事 C 语言教学的经验,根据高职高专教学的特点,精心组织编写了《C 语言程序设计案例与实训教程》一书。本书对学生学习 C 语言中容易出现的问题、难点进行了详细的解析,对 C 语言程序设计的应用技巧进行了重点讲评。读者可以通过书中提供的“技能实训”任务和“检测评价”习题,在实践中进一步理解 C 语言的语法语义,掌握 C 语言程序设计的一般过程和方法,开拓思路,培养编程能力。希望本书能成为每一位 C 语言学习者的良师益友。

此外,本书还有如下特点。

(1) 本书按照“任务驱动”的教学方法编排。各章节均有明确可行的学习任务和目标要求,对学习活动的导向非常明晰。

(2) 学习活动以“知识导学→案例程序分析→疑难辨析→技能实训→学习效果的检测评价→收获反思”为主线进行,遵循了由浅入深、循序渐进的原则,将技能培养和专业素质提升放在首位。

(3) 本书重视上机实践。根据我们在实践教学中的体会,一是对上机实训前的准备阶段学习,提出了明确的要求,引导学习者积极思考和实践;二是除了技能实训中的编写程序任务,其他实训任务均不要求现场录入程序,这样就节省了学习者的上机时间,有利于有针对性地辅导,大大提高了单位时间内的实训学习效率。

(4) 本书的内容以及各章节的案例程序、实训任务和检测习题,均经过字斟句酌、深思熟虑、反复推敲,力求不留遗漏和遗憾给读者。书中所有案例和习题都已经上机调试并通过。

(5) 本书对每个案例程序都添加了适当的注释,以分析和解说程序语句,帮助读者阅读案例程序。同时,也起到了潜移默化的作用,让学习者愿意和习惯书写注释,增加程序的可读性。

(6) 本书兼顾了“全国计算机等级考试(二级 C 语言程序设计)”的要求,精选了历年二级 C 语言程序设计试题中有代表性的题目。也适合作为全国等级考

试(二级`C)得力的复习资料。

本教材共分为 8 章和附录,由周丰担任主编。第 1、6、7、8 章和附录由周丰编写,第 2 章由熊慧编写,第 3 章及第 2 章第 8 节由姜静思编写,第 4 章由黄庆编写,第 5 章由曾怡编写,邢晓敏编写了涉及单片机 C 程序问题的部分内容。周丰负责全书的修改和统稿工作。参与本书编写的作者都是从事计算机教学 10 年以上的中高级教师,书中包含了他们多年的工作和思考。

由于时间仓促,错误和不当之处在所难免,敬请读者和专家批评指正。

编 者

目 录

第 1 章 C 语言概述	1
1.1 C 语言程序结构和书写格式	1
1.2 C 语言程序的开发步骤	6
第 2 章 C 语言基础知识	14
2.1 常量和变量	14
2.2 数值型数据	20
2.3 字符型数据	27
2.4 算术表达式及数据类型转换	32
2.5 赋值表达式、增量表达式及逗号表达式	39
2.6 关系表达式、逻辑表达式及条件表达式	45
2.7 位运算符	52
2.8 数据的输入输出	57
第 3 章 程序的基本结构	66
3.1 顺序结构程序设计	66
3.2 选择结构程序设计	71
3.3 多分支选择语句	80
3.4 循环结构程序设计	89
3.5 循环嵌套语句	97
3.6 循环控制语句	103
3.7 循环程序设计举例	108
第 4 章 数组	115
4.1 一维数组	115
4.2 二维数组	121
4.3 字符数组和字符串	128
4.4 字符串处理函数	135
4.5 数值处理程序举例	142
4.6 字符处理程序举例	153

第 5 章 函数与编译预处理	159
5.1 函数的定义与调用	159
5.2 函数调用的数据传递	167
5.3 函数的嵌套调用与递归调用	176
5.4 变量的存储类型与作用域	185
5.5 编译预处理	195
第 6 章 指针	204
6.1 变量的地址和指针	204
6.2 指针与函数参数	212
6.3 指针变量与一维数组	219
6.4 指针与字符串	228
6.5 数组指针与指针数组	236
6.6 函数型指针	246
6.7 带参数的 main() 函数	252
第 7 章 结构体、共用体和枚举类型	256
7.1 结构体的定义和结构变量的引用	256
7.2 指向结构的指针	267
7.3 共用体	278
7.4 枚举类型	283
第 8 章 文件	287
8.1 文件及其打开与关闭	287
8.2 文件的字符读写函数和字符串读写函数	293
8.3 文件的数据块读写函数和格式化输入输出函数	297
8.4 文件操作中的其他函数	303
附录 A ASCII 码表	308
附录 B 运算符的优先级和结合性	309
附录 C 常用 C 语言库函数	310
附录 D C 语言新老版本的差别	316
参考文献	318

第 1 章 C 语言概述

随着电子计算机的迅速发展和广泛应用,C 程序设计语言(简称 C 语言)在计算机软件开发中的作用日益重要。当前几乎各种型号的微机 and 小型计算机,以及某些大中型计算机上都配置有 C 语言编译系统。良好的开发环境,使 C 语言成为一种深受程序设计者欢迎并广泛使用的程序设计语言。

1.1 C 语言程序结构和书写格式

C 语言是一种深受程序设计者欢迎并广泛使用的程序设计语言,可以在 Unix、DOS 等环境下运行。常见的版本有 Turbo C、Microsoft C、lattice C、Quick C 和 borland C 等。不同版本略有差异,因此读者在使用具体 C 编译系统时,还应通过阅读有关手册了解它的具体规定。本书用 Visual C++ 编译系统作为实现的版本。

【学习任务】

认识 C 语言,体验 C 语言程序的基本结构。

【学习目标】

- (1) 了解 C 语言及其发展概况。掌握 C 语言源程序的基本结构特点。
- (2) 了解 C 语言语法、语义及 C 语言源程序的一般书写规则。
- (3) 熟练掌握 Visual C++ 6.0 集成开发环境下 C 语言程序的开发步骤。

【知识导学】

1. C 语言及其发展概况

C 语言是目前世界上流行最广的一种程序设计语言。它的发展过程可粗略地分为三个阶段:1970 年至 1973 年为诞生阶段,1973 年至 1988 年为发展阶段,1988 年以后为成熟阶段。

C 语言既适合于编写系统程序,又适合于编写应用程序。以前的系统程序主要是用机器语言或汇编语言编写的。例如,Unix 操作系统就是由美国贝尔实验室 K. Thompson 和 D. M. Richie 在 1969 年用机器语言编写的,随后又用汇编语言编写。由于汇编语言不可移植,并且描述问题的效率不如高级语言,特别是可读性差,所以 K. Thompson 决定开发一种高级语言来描述 Unix 系统。早在 20 世纪 60 年代,英国剑桥大学的 M. Richards 就在 CPL 语言的基础上开发了早期的编程语言 BCPL(Basic Combind Programming Language)。1970 年,K. Thompson 以 BCPL 语言为基础,作了进一步简化,设计出既简单又接近硬件的 B 语言(取

BCPL 的第一个字母),在 PDP-11/20 上实现 B 语言并用 B 语言编写了 Unix 操作系统和绝大多数应用程序。由于 B 语言过于简单,数据无类型(或者说只是一种机器字类型),描述能力有限,B 编译程序产生的是解释执行的代码,运行速度慢,故没有流行起来,而导致 D. M. Ritchie 于 1971 年在 B 语言的基础上开始设计 C 语言(取 BCPL 第二个字母)。最初的 C 语言只是为了描述和实现 Unix 操作系统提供一种工作语言而设计的。1973 年 K. Thompson 和 D. M. Ritchie 两人合作把 Unix 的 90% 以上部分用 C 语言进行了重新编写(即 Unix 第 5 版)。以后 C 语言又作了多次改进,但主要还是在贝尔实验室内部使用。直到 1975 年 Unix 第 6 版面世后,C 语言的突出优点才引起人们的普遍关注。1977 年出现了不依赖于具体机器的 C 语言编译文本——可移植 C 语言编译程序,使 C 移植到其他机器的工作大大简化,这也推动了 Unix 操作系统在各种机器上得以实现。Unix 操作系统在全球范围内取得巨大的成功与 C 语言的使用是分不开的。随着 Unix 操作系统的应用日益广泛,C 语言也迅速得到推广。1978 年以后,C 语言已先后移植到大、中、小型机及微机上,已独立于 Unix 系统了。现在 C 语言风靡全世界,成为世界上应用最广泛的几种计算机语言之一。1983 年,美国国家标准化协会(ANSI)根据 C 语言诞生以来的各种版本对 C 语言进行了扩充、规范,制定了新的标准,称为 ANSI C。1987 年,ANSI 又公布了新标准:87ANSI C。

在 C 的基础上,1983 年又由贝尔实验室的 Bjarne Stroustrup 博士推出了 C++。C++ 进一步扩充和完善了 C 语言,成为一种面向对象(Object Oriented)的程序设计语言。C++ 提出了一些更为深入的概念,为程序员提供了一种与传统结构程序设计不同的思维方式和编程方法,因而也增加了整个语言的复杂性。但是 C 是 C++ 的基础,C++ 语言和 C 语言在很多方面是兼容的。因此,学习 C 语言,不仅是进行一种编程设计方法的训练,也是为进一步学习 C++ 打下良好的基础。

2. C 语言编程的风格、特点

一种语言之所以能存在和发展,并具有生命力,总是有些不同于或优于其他语言的风格特点。C 语言有以下几个基本特点。

(1) C 语言简洁、紧凑,使用方便、灵活。C 语言一共只有 32 个保留字、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,相对其他计算机语言而言,源程序较短,因此输入程序时工作量少。

(2) C 语言既具有高级语言的特点,又具有低级语言的一些功能。它允许直接访问地址,能进行位(bit)运算,可以直接对硬件进行操作。

(3) C 语言是一种结构化程序设计语言,它具有结构化控制语句(if else、while、do while、switch、for 等语句)。C 语言用函数作为程序模块,以实现程序的模块化。因此,C 语言十分有利于实现结构化、模块化程序设计。

(4) C 语言运算符丰富。C 语言的运算符范围很广泛,共有 34 种之多。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 语言的运算符类型极其丰富,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。

(5) C 语言的数据类型丰富,具有现代化语言的各种数据类型。C 语言的数据类型有整型、实型、字符型、数组型、指针型、结构型、联合型和枚举型等。能用来实现各种复杂的数据结构。因此,C 语言具有很强的数据处理能力。

(6) C 语言程序中可以使用如 #define、#include 等编译预处理语句,能进行字符串或特定参数的宏定义,以及实现对外部文本文件的读取和合并,同时还具有 #if、#else 等条件编译

预处理语句。这些功能的使用有利于提高程序质量和软件开发的工作效率。

(7) C语言生成的代码质量高。高级语言能否用来描述系统软件,特别是像操作系统、编译程序等,除了决定于语言表达能力以外,还有一个很重要因素就是该语言的代码质量。实验表明,C语言代码效率只比汇编语言代码效率低10%~20%,C语言是描述系统软件和应用软件比较理想的工具。

(8) C语言程序的可移植性好。C语言程序本身不依赖于机器硬件系统,从而便于在硬件结构不同的机种间和各种操作系统中实现程序的移植。

C语言的优点很多,但也有不足之处,应引起注意。C语言语法限制不太严格,程序设计时自由度大。例如,对数组下标越界不作检查,由程序编写者自己保证程序的正确。C语言对变量的类型使用比较灵活。例如,整型与字符型和逻辑型数据可以通用。C语言允许程序编写者有较大的自由度,放宽了对语法的检查。为此,程序员应当仔细检查程序,保证其正确性,而不要过分依赖C语言编译程序去查错。

3. C语言程序的书写格式

任何一种程序设计语言都有其特定的语法规则和表达方式。用某种程序设计语言进行程序设计和编程时,必须按照该语言规定的语法和表达方式编写程序。只有这样,才能保证所编写的程序可以正确地执行。

(1) C程序是由函数构成的。每个程序由一个或多个函数组成,其中必须有且仅有一个主函数main()。例如,后面“案例1.1”中的程序由一个main()函数组成,而“案例1.2”中的程序则由main()和max()两个函数组成。每一个函数均由函数首部和函数体两部分构成。其中,函数体写在一对花括号里面,而函数首部(又称函数头)写在这对花括号前。

一个可执行的C语言程序总是从main()函数开始执行的,而不论其在整个程序中的位置如何。函数使程序的模块化容易实现。

(2) 提倡在编写程序时多使用注释语句。VC中注释语句有两种写法,一是可用/*……*/对C程序中的任何部分作注释,注释可以写在程序的任何位置上,“/*”与“*/”也可不在同一行上;二是可用“//”将该行它后面的部分作为注释。建议使用后者。

注释的内容是给程序员自己看的,对程序执行不产生影响,主要用作对程序或语句的功能进行说明。注释可放在程序中的任何地方,但一般放在程序的开头或语句的后面。一个好的、有使用价值的源程序,都应当加上必要的注释,以增加程序的可读性。

(3) 一个C语言源程序开头一般都有以“#”号打头的预处理命令(上述案例中的include命令仅为其中的一种),预处理命令通常应放在源文件或源程序的最前面。C语言对输入输出实行函数化。例如,输出时可以调用printf()函数,需要由键盘输入数据时可以调用scanf()函数。printf()函数和scanf()函数的声明格式,均在“stdio.h”文件中,因此必须先做预包含“#include <stdio.h>”。

(4) 在函数体中,每条语句和变量定义的最后必须有一个英文分号“;”,但在以下三种情况下不允许有分号:

- 所定义的函数的名称后面不使用分号(即函数头后面不写“;”号);
- 定义函数时函数体的右花括号“}”后面不使用分号;
- 预处理语句后面不使用分号。

(5) C程序书写格式自由,一行内可以写几个语句,一个语句也可以分写在多行上。若一个语句过长而需要换行书写时,C语言允许使用续行标志“\”(注意“\”后面不能有回车换行以

外的其他任何字符)。如果在一行代码的行尾放置一个反斜杠, C 语言编译器会忽略行尾的换行符, 而把下一行的内容也算作是本行的内容。续行标记常用于书写较长的字符串(见第 4 章)或定义跨行的宏(见第 5 章)。

(6) 在 C 语言中大小写字母是有区别的(C 语言习惯用小写字母)。

【案例与程序】

【案例 1.1】计算两个数的和。

/* 本行为注释语句, 下面程序中使用了很多注释。请注意注释语句的书写格式。

(这种格式的注释语句可以占用很多行) */

```
# include<stdio.h>          /* 本行为包含预处理命令 */
void main( )                //主函数的函数首部(这种格式的注释语句应写在一行的后面)
{
    // 以下为函数体, 函数体用一对花括号括起来
    int m, n, sum;           // 本行为定义变量, 定义了 m、n、sum 三个整型变量
    printf("Welcome, everyone! \n"); // 本行调用输出函数输出双引号里的信息
    m = 321; n = 567;        // 本行写有 2 个语句, 分别给变量 m 和 n 赋予了整数值
    sum = m + n;             // 本行语句为求和, 计算结果赋予变量 sum 中
    printf("Sum is  %d \n", sum); // 本语句输出计算的结果
}                             // 这是括起函数体的右花括号。函数到此结束
```

程序的运行结果:

Welcome, everyone!

Sum is 888

【案例 1.2】从键盘输入两个整数, 并将最大的数显示出来。

// 本程序由 main() 和 max() 两个函数构成。

```
# include <stdio.h>
int max(int, int);          // 本行为函数声明, 将在第 5 章学习
void main( )
{
    int a, b, c;
    printf("Please input two integers:  ");
    scanf("%d, %d", &a, &b); // 本行调用键盘输入函数 scanf()
    c = max(a, b);          // 本行先调用 max() 函数, 再将 max() 的计算结果赋予变量 c
    printf("max = %d \n", c); // 本行语句输出结果
}                             // main() 函数到此为止, 请注意其函数首部和函数体的格式
int max(int x, int y)        // max() 函数的功能是找出 x、y 中较大的一个数
{
    int z;
    if(x > y)
        z = x;
    else
        z = y;
```

```

        return(z);
    }           // max()函数到此为止,请注意其函数首部和函数体的格式

```

程序的运行结果为(程序先输出提示信息给用户,然后等待用户从键盘输入用逗号间隔的2个整数“36,19”,用户敲回车键后——本书用<CR>代表回车,程序继续完成后续工作):

```

Please input two integers:    36,19<CR>
max = 36

```

【疑难辨析】

从书写清晰,便于阅读、理解和维护的角度出发,在书写程序时一般应遵循以下规则。

- 一个说明或一个语句占一行。
- 用{}括起来的部分,通常表示了程序的某一层结构。{}一般与该结构语句的第一个字母对齐,并单独占一行。
- 低一层次的语句或说明可比高一层次的语句或说明缩进若干格后书写,以便看起来更加清晰,增加程序的可读性。即所谓书写程序的“缩进格式”。

对初学者,在编程时应力求遵循这些规则,以养成良好的编程风格。

【技能实训】

认识“案例 1.1”和“案例 1.2”中程序的书写格式。请写出或说出 C 语言源程序的基本书写格式。

【检测评价】

请完成下列检测题(时间 10 分钟)。

选择题

1. (二级 2009 年 9 月选择第 11 题)以下叙述中正确的是()。
 - A. 程序设计的任务就是编写程序代码并上机
 - B. 程序设计的任务就是确定所用数据结构
 - C. 程序设计的任务就是确定所用算法
 - D. 以上三种说法都不完整
2. (二级 2008 年 9 月选择第 11 题)以下叙述中正确的是()。
 - A. C 程序的基本组成单位是语句
 - B. C 程序中的每一行只能写一条语句
 - C. 简单 C 语句必须以分号结束
 - D. C 语句必须在一行内写完
3. (二级 2007 年 9 月选择第 14 题)以下叙述中正确的是()。
 - A. C 语言程序将从源程序中第一个函数开始执行
 - B. 可以在程序中由用户指定任意一个函数作为主函数,程序将从此开始执行
 - C. C 语言规定必须用 main 作为主函数名,程序将从此开始执行,在此结束
 - D. main 可作为用户标识符,用以命名任意一个函数作为主函数
4. (二级 2008 年 4 月选择第 11 题)以下叙述中正确的是()。
 - A. C 程序中的注释只能出现在程序的开始位置和语句的后面
 - B. C 程序书写格式严格,要求一行内只能写一个语句
 - C. C 程序书写格式自由,一个语句可以写在多行上

D. 用 C 语言编写的程序只能放在一个程序文件中

5. (二级 2006 年 4 月选择第 18 题) 以下四个程序中, 完全正确的是()。

- A. `# include<stdio.h>`
`main( )`
`{`
`/* programming */`
`printf("programming! \n");`
`}`
- B. `# include<stdio.h>`
`main( )`
`{`
`/* /programming/ */`
`printf("programming! \n");`
`}`
- C. `# include<stdio.h>`
`main( )`
`{`
`/* /* programming */ */`
`printf("programming! \n");`
`}`
- D. `include<stdio.h>`
`main( )`
`{`
`/* programming */`
`printf("programming! \n");`
`}`

☆请评价自己本节内容的学习效果, 并将学习与实训的收获、体会或疑问记录下来:

1.2 C 语言程序的开发步骤

计算机不能直接识别高级语言, 要让计算机能执行高级语言程序, 需要将高级语言程序翻译成等价的机器语言, 生成相对应的可执行程序。高级语言编译、连接源程序的过程如图 1.1 所示。

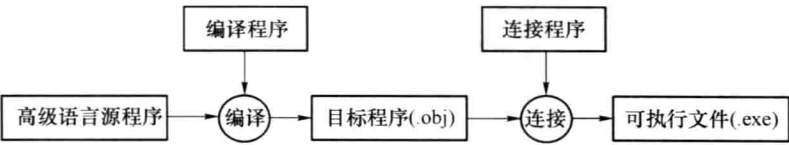


图 1.1 高级语言编译、连接源程序的过程

用 C 语言编制程序到完成运行, 一般要经过编辑、编译、连接、运行几个阶段。本节对在 Visual C++6.0 环境下运行 C 程序作一简单介绍。

【学习任务】

体验 Visual C++6.0 集成开发环境下 C 语言程序的开发步骤。

【学习目标】

- (1) 学会安装 Visual C++6.0 集成开发环境。
- (2) 认识 Visual C++6.0 集成开发环境的工作界面。
- (3) 初步掌握 Visual C++6.0 环境下 C 程序的上机步骤。

(4) 初步学习在 Visual C++ 6.0 信息输出窗口中查看 C 程序构形的编译说明。

【知识导学】

1. Visual C++ 6.0 集成开发环境及其安装与启动

Visual C++ 6.0 是运行在 Windows 平台上的交互式的可视化集成开发环境,它是 Microsoft Visual Studio 套装软件的一个组成部分。作为一种通用且功能强大的编程语言,一方面,它完美地与 Windows 平台进行结合,从而保证程序具有强大的功能;另一方面,其无可比拟的与 Windows 同步更新的优势对程序员也具有极强的吸引力。

在使用 Visual C++ 6.0 之前,先要在系统中安装该软件。该软件的安装很容易,只要从 Visual Studio 光盘中运行相应的安装程序 setup.exe,然后按照提示一步步操作即可。

安装成功后,单击桌面任务栏左侧的“开始”菜单按钮,选择“程序”菜单项,选择该菜单项下的“Microsoft Visual Studio 6.0”下的“Microsoft Visual C++ 6.0”级联菜单,即可启动 Visual C++ 6.0。

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 的小技巧。单击“Next Tip”会显示下一条技巧,单击“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,可取消复选框“Show Tips at Startup”。

关闭“Tip of Day”对话框后,就进入了 Visual C++ 6.0 工作界面的主窗口。

2. Visual C++ 6.0 集成开发环境的工作界面

启动 Visual C++ 6.0 进入 Microsoft Visual C++ 编译环境的工作界面。如图 1.2 所示。

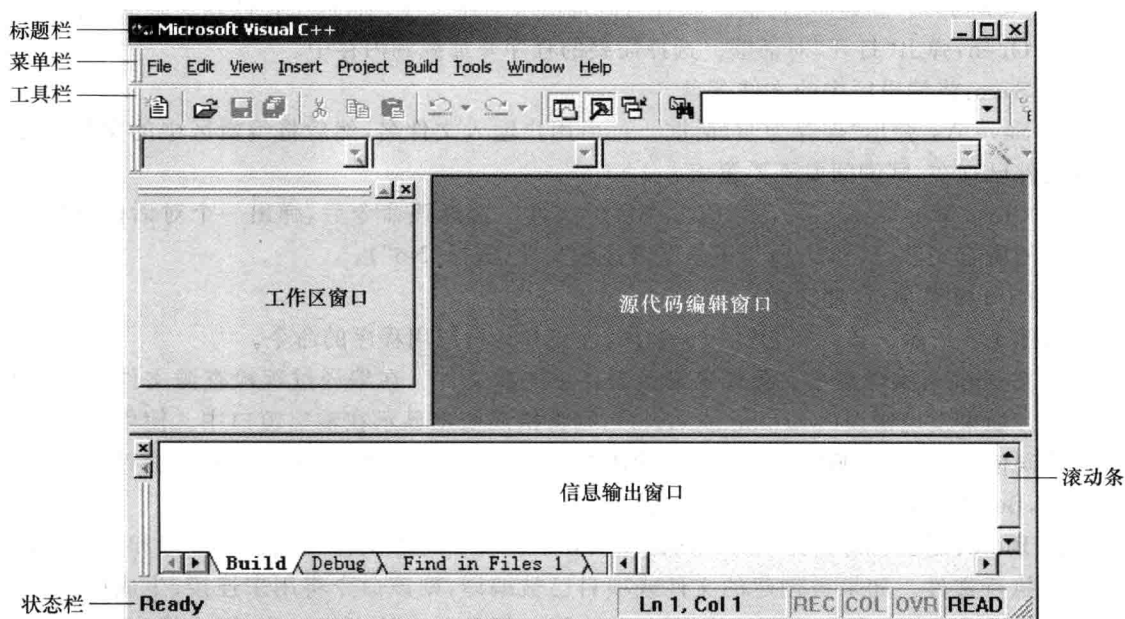


图 1.2 Microsoft Visual C++ 编译环境的工作界面

Visual C++ 6.0 集成开发环境的工作界面(即主窗口)由标题栏、菜单栏、工具栏、工作区窗口、源代码编辑窗口、信息输出窗口和状态栏等组成。

屏幕窗口中最上方是标题栏,显示所打开的应用程序名。标题栏左端是控制菜单图标,单击后弹出窗口控制菜单。标题栏右端从左至右有 3 个控制按钮,分别为最小化、最大化(还原)和关闭按钮,可用它们快速设置窗口的大小。

标题栏下方是菜单栏,由 9 个菜单项组成。单击菜单项弹出下拉式菜单,可使用这些菜单项实现集成环境的各种功能。

菜单栏下方是工具栏,它由若干个功能按钮组成,单击按钮可实现某种操作功能。

工具栏的下方有左、右两个窗口,左窗口是项目工作区窗口,右窗口是源代码编辑窗口。在项目工作窗口和源代码编辑窗口的下方有一个输出窗口,在创建项目(Build)时,用来显示项目创建过程中的错误信息。

屏幕最底部是状态栏,它可给出当前操作或所选命令的提示信息。

在上述窗口组成部分中,工作区窗口可通过单击工具栏中的 Workspace 按钮隐藏或显示;输出窗口可通过单击工具栏中的 Output 按钮隐藏或显示。隐藏这些窗口可以扩大源代码编辑区的大小。

3. Visual C++ 6.0 集成开发环境常用命令的功能

Visual C++ 6.0 主窗口的菜单栏中包含 9 个主菜单项:File(文件)、Edit(编辑)、View(查看)、Insert(插入)、Project(工程)、Build(编译)、Tools(工具)、Window(窗口)和 Help(帮助)。下面对一些常用和比较重要的菜单命令进行介绍。

(1) File 菜单(文件菜单)

如图 1.3 所示。该菜单的各个命令选项主要完成对文件进行创建、打开、关闭、保存和打印等操作。

① New:弹出“新建”对话框。该对话框是用来创建文件、项目、工作区以及其他文档的。

② Open:弹出“打开”对话框。选择需要的程序并加载到内存中。

③ Save:将编辑区中的文件存盘。

④ Save As:弹出“保存为”对话框。可由用户输入文件名,然后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。

⑤ Close Workspace:关闭当前工作区的文件。选择该命令后,弹出一个对话框,提示用户是关闭所有文件(选择“Yes”)还是保留这些文件(选择“No”)。

(2) Build 菜单(组建菜单)

如图 1.4 所示。该菜单包括用于编译、连接和运行应用程序的命令。

① Compile:编译显示在源代码编辑窗口中的源文件。在编译过程检查源文件中是否有语法错。如果发现错误(warning 或 error),则将错误信息显示在输出窗口中。使用鼠标双击某行错误信息时,将在源代码编辑窗口中用粗箭头指向出错的代码行,方便用户修改。当编译通过后,Build 菜单会发生变化。请读者自己观察。

② Build:创建当前文件项目。该命令实际上包含了对源文件或项目的编译和连接,最终生成可执行文件。如果被创建的文件或项目已被编译,则该命令将用来连接、生成可执行文件。在编译或连接中检查出语法错误时,将出错信息显示在输出窗口中,用户修改后,再进行创建,直到生成可执行文件为止。

③ Execute:运行已生成好的可执行文件,并将运行结果显示到相应的环境中(如 MS-DOS、Windows 98 或 Windows NT 等)。

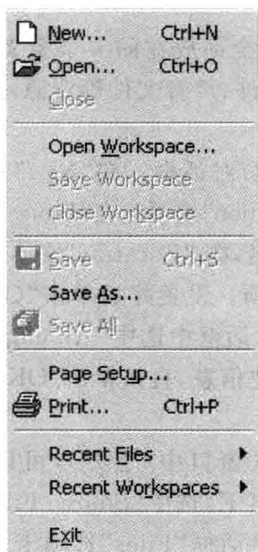


图 1.3 文件菜单

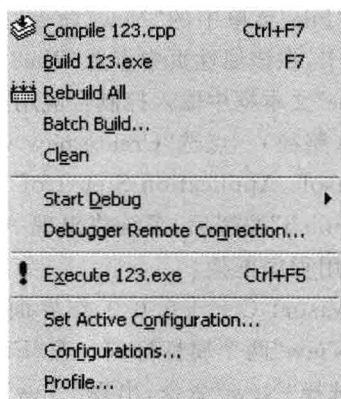


图 1.4 组建菜单

4. C语言源程序文件的创建、编辑、编译、连接和运行

单击选择“File”菜单中的“New”命令,弹出“New”(新建)对话框,如图 1.5 所示。在“New”对话框中点选“Files”选项卡。在“Files”选项卡左侧的列表框中点选“C++ Source File”选项。在“Files”选项卡右侧的“File”文本框中输入文件名(文件的扩展名. cpp 由编译系统自动生成);再在“Location”文本框中选择或输入文件保存的位置(路径)。最后,单击“Files”选项卡下方的“OK”(确定)按钮,进入源代码编辑窗口。

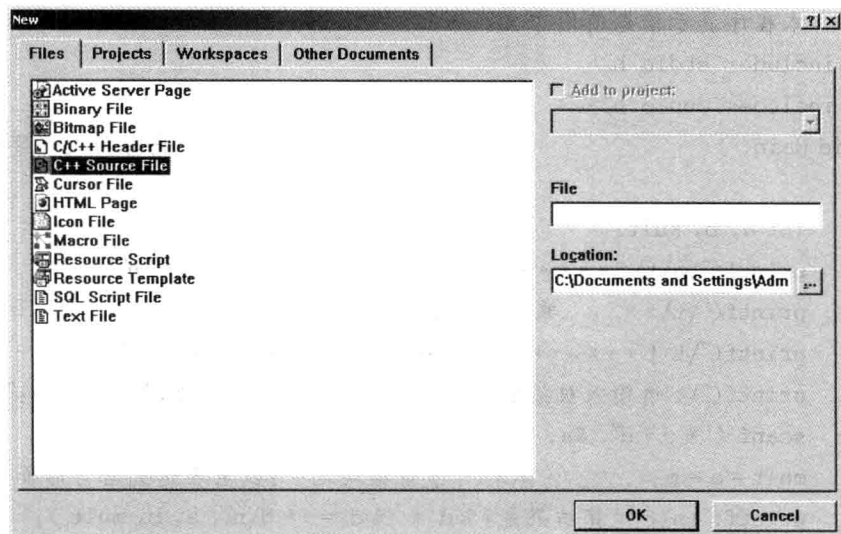


图 1.5 源代码编辑窗口

在源代码编辑窗口中完成程序代码的编辑后,使用“Build”菜单中的“Compile”、“Build”命令进行编译、调试和连接。成功生成可执行文件后,使用“Build”菜单中的“Execute”命令运行,并观察运行结果。

5. 在 Visual C++ 6.0 集成开发环境中建立工程

如果程序比较大,有成千上万行的代码,这时若将它们全部放在同一个源文件里,会显得杂乱无章,很难管理。此外,程序中可能还会用到如图像文件、声音文件等资源,这时需要建立工程(Project)。

单击选择“File”菜单中的“New”命令,弹出“New”(新建)对话框。在“New”对话框中点选“Projects”选项卡,在该属性页中点选“Win32 Console Application”选项。在“Projects”选项卡右侧的“Project name”文本框中输入控制台应用程序的工程文件名,在“Location”文本框中选择或输入保存的位置(路径)。点选“Create new workspace”单选项。设置好后单击“OK”(确定),进入“Win32 Console Application Step 1 of 1”对话框,在该对话框中选中“An empty project”单选项。单击“Finish”按钮后,系统将出现 AppWizard 的创建信息,直接单击“OK”按钮,系统会建立要求的应用程序框架。

这时,在 Visual C++ 6.0 工作界面中的项目工作区窗口中(下方),可以看到“Class-View”和“FileView”两个属性标签。在“FileView”属性标签下,选中“Source Files”,然后再在“File”菜单下选择“New”命令,出现“New”对话框。在该对话框“Files”选项卡属性列表中选择“C++ Source File”选项。在“File”文本框中输入文件名,在“Location”文本框中选择或输入文件保存的位置(路径),单击“OK”按钮,进入源代码编辑窗口。编辑源程序代码并调试无误后,编译,连接,运行程序。

【案例与程序】

【案例 1.3】以下程序计算任意两个整数的乘积。调试程序使之正确并运行。

/* 本程序的功能是计算任意两个整数的乘积。被乘数和乘数由用户从键盘输入。

注意:本程序故意预留了 3 个错误,供学习程序的调试方法用。

本程序提示信息使用中文。*/

```
#include<stdio.h>
#include<conio.h>
void Main() // 本行有预留错误之三:字母错写为大写字母。
{
    int a, b, mult;
    printf("\n\t\t*****\n");
    printf("\t\t* 乘积的计算 *\n");
    printf("\t\t*****\n\n");
    printf("\t请输入被乘数和乘数(用空格间隔两个数,然后回车:");
    scanf ("%d%d", &a, &b);
    mult = a * b // 本行有预留错误之一:赋值表达式语句后面缺少分号。
    printf("\n\t运算结果是:%d * %d = %d\n", a, b, mult);
    printf("\n\n\t\t***** 再 见 ! *****\n");
    getch();
    // 本行有预留错误之二:漏写函数结束的右花括号。
```

调试正确后,程序的运行结果应为(以计算 12×3 为例,空行省略):
