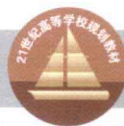


21世纪高等学校规划教材

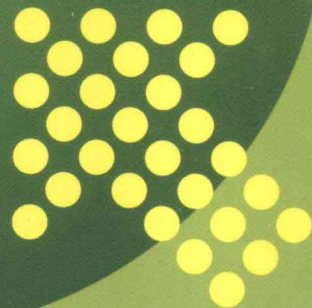


C YUYAN CHENGXU SHEJI

C语言程序设计

(第二版)

孙连科 许薇薇 编



中国电力出版社
CHINA ELECTRIC POWER PRESS

21世纪高等学校规划教材

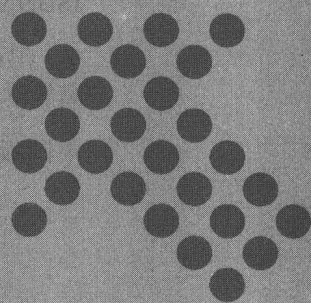


C YUYAN CHENGXU SHEJI

C语言程序设计

(第二版)

孙连科 许薇薇 编
李卓玲 主审



中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书为 21 世纪高等学校规划教材。本书是学习 C 语言程序设计的基础教程。全书共 12 章, 内容包括 C 语言概述, 数据类型、运算符与表达式, 顺序结构程序设计, 选择结构程序设计, 循环结构程序设计, 函数与编译预处理, 数组, 指针, 结构体和共用体, 位运算, 文件, 综合实例。

本书可作为普通高等院校 C 语言程序设计课程的本、专科教材(可以根据本科、专科教学要求的不同进行适当取舍), 也可作为备考等级考试和其他从事计算机编程人员的参考用书。

图书在版编目(CIP)数据

C 语言程序设计 / 孙连科, 许薇薇编. —2 版. —北京: 中国电力出版社, 2011.1

21 世纪高等学校规划教材

ISBN 978-7-5123-1301-9

I. ①C… II. ①孙… ②许… 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(2010)第 002912 号

中国电力出版社出版、发行

(北京市东城区北京站西街 19 号 100005 <http://www.cepp.sgcc.com.cn>)

北京市铁成印刷厂印刷

各地新华书店经售

*

2006 年 1 月第一版

2011 年 2 月第二版 2011 年 2 月北京第五次印刷

787 毫米×1092 毫米 16 开本 16.5 印张 401 千字

定价 28.00 元

敬告读者

本书封面贴有防伪标签, 加热后中心图案消失

本书如有印装质量问题, 我社发行部负责退换

版 权 专 有 翻 印 必 究

前 言

C 语言是国内外广泛使用的结构化程序设计语言,它功能丰富、表达能力强、使用方便灵活、目标程序效率高、可移植性好,既具有高级语言的优点,又具有低级语言的许多特点。因此,C 语言既可用于开发系统软件,也可用于开发应用软件,应用面很广。目前,多数高等院校不仅计算机专业开设 C 语言程序设计课程,非计算机专业也开设了这门课程。同时,许多学生都选择 C 语言作为参加全国计算机等级考试(二级)的考试科目。学习 C 语言已成为广大学生的迫切需要。

根据教材第一版的使用情况,这次对该教材的内容进行了大幅度的调整和增删。第二版与第一版相比,一是删除难度较高的、对数学知识要求较高的例题,增加了一些易理解而又典型的例题。二是考虑到高等院校的教学特点及实际需要,对原书的章节编排做了适当调整,使内容更加紧凑、结构更加合理。三是由于国家级计算机等级考试中,C 语言上机考试环境已经由 Turbo C 环境改为 Visual C++ 环境,根据形势的变化,本书采用 Visual C++ 6.0 作为开发环境。本书的具体特点如下:

(1) 教材内容经过精心组织,体系合理、结构严谨,全面讲授 C 语言程序设计的基本思想、方法和解决实际问题的技巧。

(2) C 语言的概念比较复杂,规则较多,使用灵活,容易出错,不少初学者感到困难。教材内容组织形式由浅入深、循序渐进,以便于学生学习并有利于提高学生的程序设计能力。

(3) 内容丰富,注重实践;突出重点,分散难点。本书的宗旨在于进一步巩固学生对基本知识的理解和掌握,提高学生的逻辑分析、抽象思维和程序设计能力,培养学生用计算机编程解决实际问题的能力。

(4) 对所介绍的内容都列举了典型的实例,所有实例均在 Visual C++ 6.0 环境下上机调试并通过,便于教师在上课时演示。同时,每章后都设有精心挑选的多种类型的习题,以帮助读者通过练习进一步理解和巩固所学的内容。

本书由沈阳工程学院的孙连科和许薇薇共同编写。其中,第 1 章~第 6 章由孙连科编写,第 7 章~第 12 章由许薇薇编写。

本书由沈阳工程学院李卓玲教授担任主审,她提出了许多宝贵的意见,在此表示衷心感谢。

在本书的编写过程中,编者广泛参阅、借鉴了国内外 C 语言程序设计方面的相关教材和资料,在此谨向这些教材和资料的作者致以诚挚的感谢。

限于编者水平,加之编写时间仓促,书中难免有疏漏和不足之处,恳请读者和专家批评指正。

编 者

2010 年 12 月

第一版前言

随着计算机技术的飞速发展,计算机已在各行各业得到了广泛应用。计算机技术的中心是程序设计,掌握一定的程序设计技术,是当今社会人们应当必备的技能与文化素质之一。掌握程序设计的前提是掌握程序设计语言,在众多的程序设计语言中,C 语言以其灵活性和实用性受到编程人员的广泛青睐。

C 语言概念简洁,数据类型丰富,表达能力强,运算符多而灵活,程序结构性和可读性好,易于体现结构化程序设计思想。它既具有高级语言的特点,又具有汇编语言的功能;既能有效地进行算法描述,又能对硬件直接进行操作;既适合编写应用程序,又适合开发系统软件,它是当今世界上应用广泛、最具影响的程序设计语言之一。此外,C 语言还具有整个语言紧凑整齐、设计精巧、编辑方便、编译与目标代码运行效率高、操作简便、使用灵活等许多鲜明的特点。

本书全面介绍了 C 语言的概念、特性和结构化程序设计方法。全书共有 12 章,第 1 章介绍了 C 语言程序设计的基本知识。第 2 章介绍了 C 语言的基本数据类型、常量、变量以及表达式。第 3、4、5 章介绍了 C 语言进行结构化程序设计的基本方法,包括数据输入/输出、顺序结构、选择结构、循环结构及其设计方法。第 6 章介绍了函数与预编译处理。第 7、8、9、10 章对 C 语言的数组、指针、结构体、共用体等作了较详细的介绍。第 11 章对 C 语言的文件操作做了详细的阐述。第 12 章介绍了 C 语言综合编程应用。本书在各章后都配有一定的上机操作题,意在加强实践环节,通过实际练习更好地掌握 C 语言。

在本书的编写过程中,得到了李卓玲教授的热情支持与指导并提出许多宝贵意见。本书由李珍教授担任主审,在审阅过程中提出了很多极有价值的意见和建议。在此对两位老师表示衷心的感谢。

由于编者水平有限,书中难免有错误和不当之处,敬请读者批评指正。

编 者

2005 年 9 月

目 录

前 言

第一版前言

第 1 章 C 语言概述	1
1.1 程序与程序设计语言	1
1.2 C 语言发展概述和主要特点	2
1.3 C 语言程序设计方法	3
1.4 Visual C++ 6.0 集成开发环境简介	5
本章小结	7
习题	8
第 2 章 数据类型、运算符与表达式	9
2.1 基本标识符	9
2.2 C 语言的数据类型	10
2.3 常量和变量	12
2.4 整型数据	14
2.5 实型数据	15
2.6 字符型数据	16
2.7 运算符及表达式	17
2.8 数据类型转换	20
本章小结	22
习题	24
第 3 章 顺序结构程序设计	26
3.1 C 语句分类概述	26
3.2 数据输入 / 输出	28
3.3 程序举例	37
本章小结	38
习题	39
第 4 章 选择结构程序设计	43
4.1 关系运算符和关系表达式	43
4.2 逻辑运算符与逻辑表达式	44
4.3 if 语句	45
4.4 switch 语句	50
4.5 程序举例	53
本章小结	56

习题	57
第5章 循环结构程序设计	60
5.1 while 语句	60
5.2 do-while 语句	61
5.3 for 语句	62
5.4 break 语句、continue 语句和 goto 语句	65
5.5 循环的嵌套	67
5.6 程序举例	69
本章小结	73
习题	73
第6章 函数与编译预处理	77
6.1 模块化程序设计与函数	77
6.2 函数的定义与调用	78
6.3 函数的递归调用	83
6.4 变量作用域与存储方式	85
6.5 内部函数和外部函数	93
6.6 编译预处理	94
6.7 程序举例	97
本章小结	99
习题	100
第7章 数组	105
7.1 一维数组的定义和引用	105
7.2 二维数组的定义和引用	108
7.3 字符数组与字符串	110
7.4 数组作为函数的参数	118
7.5 程序举例	120
本章小结	126
习题	126
第8章 指针	131
8.1 指针概述	131
8.2 指针变量的赋值与引用	135
8.3 指针变量的运算	137
8.4 指针与数组	138
8.5 指针与函数	147
8.6 带参数的 main() 函数及其使用	155
8.7 程序举例	156
本章小结	158
习题	159

第 9 章 结构体和共用体	167
9.1 结构体	167
9.2 结构体类型数组	171
9.3 指向结构体的指针	172
9.4 结构体和函数	174
9.5 共用体	177
9.6 枚举类型	180
9.7 用 typedef 进行类型定义	182
9.8 综合实例 简单链表	184
本章小结	196
习题	196
第 10 章 位运算	202
10.1 概述	202
10.2 位运算	202
10.3 位运算举例	204
10.4 位段	206
本章小结	207
习题	207
第 11 章 文件	209
11.1 文件概述	209
11.2 文件的读写	214
11.3 随机文件和定位操作	222
11.4 文件状态检测和错误处理	224
11.5 综合实例 学生信息文件的存取	225
本章小结	228
习题	228
第 12 章 C 语言综合应用程序示例	232
附录 A 常用字符与 ASCII 代码对照表	237
附录 B C 语言中的关键字	239
附录 C 运算符和结合性	239
附录 D C 语言编译、连接时常见的错误和警告信息	241
附录 E C 语言常用部分库函数	247
参考文献	255

第 1 章 C 语言概述

教学目标

了解程序与程序设计的基础知识,了解 C 语言程序的基本结构,掌握 C 语言程序的上机步骤。

本章要点

- C 语言的特点
- C 语言程序的基本结构
- C 程序设计步骤
- C 语言程序的上机步骤

1.1 程序与程序设计语言

1.1.1 程序

程序是指可以被计算机连续执行的一条条指令的集合,也可以说是人与机器进行“对话”的语言。

人们将需要计算机做的工作写成一定形式的指令,并把它们存储在计算机的内部存储器中,当人为地给出命令之后,这些指令就被计算机按指令操作顺序自动运行,这样程序就被执行了。

1.1.2 程序设计

程序设计就是用程序设计语言编写程序的过程。

广义上说,程序设计是用计算机解决一个实际应用问题时的整个处理过程,包括提出问题、确定数据结构、确定算法、编程、调试程序及书写文档等一系列的过程。

1.1.3 程序设计语言

人们利用计算机解决实际问题,一般要编写程序。程序设计语言就是用户用来编写程序的语言,它是人与计算机之间交换信息的工具。

程序设计语言一般分为机器语言、汇编语言和高级语言 3 类。

(1) 机器语言。机器语言是最底层的计算机语言。用机器语言编写的程序,计算机硬件可以直接识别和运行。在用机器语言编写的程序中,每一条指令都是二进制形式的指令代码。在指令代码中一般包括操作码和地址码,其中操作码告诉计算机作何种操作,地址码则指出被操作的对象。

例如,代码 10000000 表示加法操作,代码 10010000 表示减法操作。

对于不同的计算机硬件(主要是 CPU)而言,其机器语言是不同的。因此,针对一台计算机所编写的机器语言程序不一定在另一台计算机上运行。由于机器语言程序是直接针对计算机硬件的,因此它的执行效率比较高,能充分发挥计算机的速度性能。但是,用机器语言编写程序的难度比较大、容易出错,而且程序的直观性比较差,也不容易移植。

(2) 汇编语言。为了便于理解和记忆,人们采用能帮助记忆的英文缩写符号(称为指令助记符)来代替机器语言指令代码中的操作码,用地址符号来代替地址码。例如,ADD 表示加法运算操作码,SUB 表示减法运算操作码。用指令助记符及地址符号书写的指令称为汇编指令,用汇编指令编写的程序称为汇编语言源程序。汇编语言又称为符号语言。

汇编语言也是与具体使用的计算机相关的。由于汇编语言采用了助记符,因此它比机器语言直观,容易理解和记忆。用汇编语言编写的程序也比机器语言程序易读、易检查、易修改。但是,计算机不能直接识别源程序,必须由一种专门的翻译程序将汇编语言源程序翻译成机器语言程序后,计算机才能识别并执行。这种翻译的过程称为“汇编”,负责翻译的程序称为汇编程序。

(3) 高级语言。机器语言和汇编语言都是面向机器的语言,一般称为低级语言。低级语言对计算机的依赖性太大,用它们开发的程序通用性差,普通的计算机用户也很难胜任这一工作。

随着计算机技术的发展及计算机应用领域的不断扩大,20 世纪 50 年代中期开始逐步出现面向问题的程序设计语言,称为高级语言。高级语言与具体的计算机硬件无关,描述问题采用接近于数学语言或人的自然语言,人们易于接受和掌握。用高级语言编写程序要比用低级语言容易得多,并大大简化了程序的编制和调试过程,使编程效率得到了大幅提高。高级语言的显著特点是独立于具体的计算机硬件,通用性和可移植性好。

用高级语言编写的程序(称为源程序)计算机不能直接执行,它需要通过编译程序翻译成机器语言程序(称为目标程序)后才能执行,或者通过解释程序边解释边执行。

1.2 C 语言发展概述和主要特点

1.2.1 C 语言的发展历史

C 语言是国际上广泛流行的一种计算机高级语言。用 C 语言既可以编写系统软件,也可以编写应用软件。

C 语言是在 1972~1973 年间由美国贝尔实验室的 D.M.Ritchie 和 K.Thompson 及英国剑桥大学的 M.Riohards 等为描述和实现 UNIX 操作系统而设计的。UNIX 操作系统源代码的 90%以上是用 C 语言编写的。UNIX 操作系统的一些主要特点,如便于理解、易于修改、具有良好的可移植性等,在一定程度上都受益于 C 语言。所以,UNIX 操作系统的成功与 C 语言是密不可分的。

最初的 C 语言附属于 UNIX 的操作系统环境,而它的产生却可以更好地描述 UNIX 操作系统。时至今日,C 语言已独立于 UNIX 操作系统。它已成为微型、小型、中型、大型和超大型(巨型)计算机上通用的一种程序设计语言。M.D.Ritchie 和 K.Thompson 也以他们在 C 语言和 UNIX 系统方面的卓越贡献获得了很高的荣誉。1982 年,他们获得了《美国电子学杂志》颁发的成就奖,成为该奖自颁发以来首次因软件工程成就而获奖的获奖者。1983 年,他们又获得了计算机界的最高荣誉奖——图灵奖。1989 年,ANSI 发布了第一个完整的 C 语言标准——C89,人们习惯称其为“ANSI C”,C89 在 1990 年被国际标准组织 ISO 一字不改地采纳,所以也有“C90”的说法,1999 年,在做了一些必要的修改和完善之后,ISO 发布了新的 C 语言标准——C99。

随着计算机应用领域的不断扩展和深入,作为人与计算机进行信息交流工具之一的 C 语言同样得到了迅速的发展。C++是在 C 语言的基础上发展起来的程序设计语言,它是一种多范型程序设计语言,人们不仅可以为其编写面向对象的程序,也可以用其编写面向过程的程序。随后 Sun 公司和 Microsoft 公司又相继推出了 Java 语言和 C#语言。但可以说 C 语言是 C++、Java 和 C#语言的基础,还有很多专用语言也学习和借鉴了 C 语言,如进行 Web 开发的 PHP 语言、做仿真的 MATLAB 的内嵌语言等。所以,学好 C 语言对以后再学习其他语言大有帮助。

1.2.2 C 语言的主要特点

(1) 语言基本组成部分紧凑简洁。C 语言只有 32 个标准关键字、44 个标准运算符及 9 条控制语句,不但语言的组成精练、简洁,而且使用方便、灵活。

(2) C 语言运算符丰富,表达能力强。C 语言具有“高级语言”和“低级语言”的双重特点,其运算符包含的内容广泛,所生成的表达式简练、灵活,有利于提高编译效率和目标代码的质量。

(3) C 语言数据结构丰富,结构化强。C 语言提供了编写结构化程序所需要的各种数据结构和控制结构,这些丰富的数据结构、控制结构及以函数调用为主的程序设计风格,保证了利用 C 语言所编写的程序能够具有良好的结构化。

(4) 具有结构化的控制语句。如 if-else 语句、switch 语句、while 语句、do-while 语句、for 语句。用函数作为程序模块以实现程序的模块化,是结构化的理想语言,符合现代编程风格。

(5) C 语言提供了某些接近汇编语言的功能,有利于编写系统软件。C 语言提供的一些运算和操作,能够实现汇编语言的一些功能,如它可以直接访问物理地址,并能进行二进制位运算等,这为编写系统软件提供了方便条件。

(6) C 语言程序所生成的目标代码质量高。C 语言程序所生成的目标代码效率比用汇编语言描述同一个问题低 20%左右。

(7) C 语言程序可移植性好。C 语言所提供的语句中,没有直接依赖于硬件的语句,与硬件有关的操作,如数据的输入、输出等都是通过调用系统的库函数来实现的,而库函数本身不是 C 语言的组成部分。因此用 C 语言编写的程序可以很容易地从一种计算机环境移植到另一种计算机环境中。

C 语言的弱点:一是运算符的优先级较复杂,不容易记忆;二是由于 C 语言的语法限制不太严格,增强程序设计灵活性的同时,在一定程度上也降低了某些安全性,因此对程序设计人员提出了更高的要求。

1.3 C 语言程序设计方法

1.3.1 C 语言程序的基本结构

在使用 C 语言编写程序时必须按其规定的格式和提供的语句进行编写。我们通过简单的例子介绍 C 语言程序的基本结构。

【例 1.1】 从键盘上读入两个整数,计算这两个整数之和并显示出来。

```
#include "stdio.h"
main()
{   int a,b,sum;                      /*定义变量 a,b,sum */
    printf("Enter two numbers: ");
    scanf("%d%d",&a,&b);              /*调用函数输入 a,b 的值 */
    sum=a+b;
    printf("The sum is %d\n",sum);     /*调用函数输出 sum 的值 */
}
```

【例 1.2】 从键盘上读入两个整数，求出其中最大的数并显示。

```
#include "stdio.h"
main()
{   int a,b,c;
    printf("Enter two numbers: ");
    scanf("%d%d",&a,&b);
    c=max(a,b);
    printf("The max is %d\n",c);
}

int max(int x,int y)                  /*函数定义*/
{   int z;
    if(x>y) z=x;
    else z=y;
    return z;
}
```

从以上的程序例子中可以看出 C 语言程序的构成规则：

(1) C 语言程序由一个或多个函数构成，其中有且仅有一个主函数 `main()`。C 语言程序的执行总是从主函数开始，并在主函数中结束。

(2) 函数体是由花括号 `{}` 括起来的部分。

(3) C 语言程序中的每个语句都以 “;” 结束。

(4) C 语言程序书写格式自由，一行内可以写一个语句，也可以写多个语句。

(5) `#include` 是编译预处理命令，其作用是将由双引号或尖括号括起来的文件内容读入该命令位置处。对于输入/输出库函数一般需要使用 `#include` 命令将 “`stdio.h`” 文件包含到源文件中。`#include` 命令的使用方法将在第 6 章介绍。

(6) 可用 `/*...*/` 对 C 语言程序中的任何部分作注释。

(7) C 语言程序中所有变量都必须先定义类型，然后再使用。

(8) C 语言程序严格区分大写字母，程序代码习惯使用小写字母，所有的关键字都必须用小写字母。

(9) C 语言程序通过函数之间的相互调用来实现相应的功能。函数既可以是系统提供的库函数（标准函数），也可以是根据问题的需要自己定义的函数。

(10) 编写程序要规范，培养良好的程序设计风格，最好采用缩进对齐的书写格式。

1.3.2 C 语言程序设计步骤

C 语言程序设计的一般步骤如下：

(1) 设计算法。针对具体的问题，分析、建立解决问题的物理或数学模型，并将解决方法采用某种方式描述出来，为 C 语言程序设计打下良好基础。

(2) 编辑。使用一个文本编辑器编辑 C 语言源程序，并将其保存为文件扩展名为.C 的文件。

(3) 编译。编译就是将编辑好的 C 语言源程序翻译成二进制目标代码的过程。编译过程由 C 语言编译系统自动完成。编译时首先检查源程序的每一条语句是否有语言错误，当发现错误时，就在屏幕上显示错误的位置和错误类型信息，要再次调用编辑器进行查错并修改，然后再进行编译，直到排除所有的语法和语义错误。一旦正确的源程序文件经过编译后，就会在磁盘上生成同名的目标文件（扩展名为.obj）。

(4) 连接。就是将目标文件和库函数等连接在一起形成一个扩展名为.exe 的可执行文件。如果函数名称写错或漏写包含库函数的头文件，则可能出现提示错误的信息，从而得到程序错误提示信息。

(5) 执行。可执行文件可以脱离 C 语言编译系统，直接在操作系统下运行。若执行程序后达到预期的目的，则 C 语言程序的开发工作到此完成，否则要进一步修改源程序，重复编辑→编译→连接→运行的过程，直到取得正确结果为止。这一过程如图 1-1 所示。

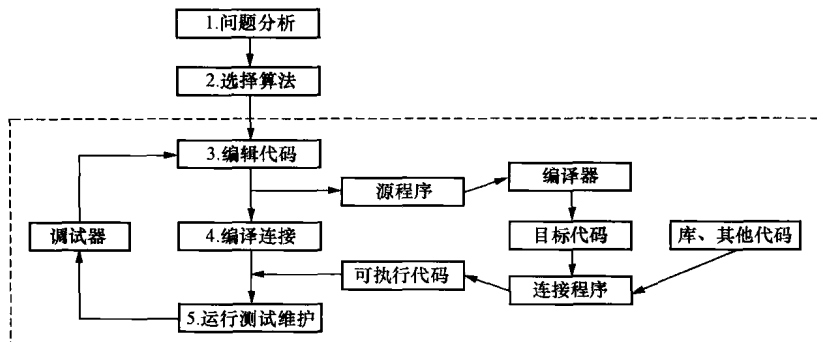


图 1-1 C 语言程序设计的一般步骤

1.4 Visual C++ 6.0 集成开发环境简介

对于 C 语言程序的源代码编写、编译、连接和运行测试维护等步骤（如图 1-1 所示虚框中的步骤），许多 C 语言工具软件商都提供了各自的集成开发环境（Integrated Development Environment, IDE）用于程序的一体化操作。集成开发环境又称编程环境，是每个 C 语言学习者都必须要掌握的开发工具。目前国内高等院校中用于在 DOS、Windows 平台上进行 C 语言教学的 IDE 中比较流行的有 Turbo C2.0、Visual C++ 6.0 及 WIN-TC1.9.1 等。本教材介绍全国计算机等级考试采用的集成开发环境 Visual C++ 6.0。

Visual C++ 是 Microsoft 公司推出的目前使用极为广泛的基于 Windows 平台的可视化编程环境。Visual C++ 6.0 是在以往版本不断更新的基础上形成的，由于其功能强大、灵活性好、完全可扩展及具有强有力的 Internet 支持，在各种 C/C++ 语言开发工具中脱颖而出，成为目前较为流行的 C/C++ 语言集成开发环境。

由于 Visual C++ 对 C/C++ 应用程序是采用文件夹的方式来管理的，即一个 C 程序的所有源代码、编译的中间代码、连接的可执行文件等内容均放置在与程序同名的文件夹及其 debug

（调试）或 release（发行）子文件夹中。因此，在用 Visual C++ 进行应用程序开发时，一般先要创建一个工作文件夹，以便于集中管理和查找。

下面以 [例 1.1] 为例来说明在 Visual C++ 6.0 中创建、编译、连接和运行的一般过程。

1. 创建工作文件夹

创建 Visual C++ 6.0 的工作文件夹，其路径可以设为“D:\C 程序”，以后所有创建的 C 程序都将保存在此文件夹下。

2. 启动 Visual C++ 6.0

选择“开始”→“程序”→Microsoft Visual Studio 6.0→ Microsoft Visual C++ 6.0 命令，运行 Visual C++ 6.0。第一次运行时，将显示“每日提示”对话框。单击“下一条”按钮，可看到有关各种操作的提示。如果在“启动时显示提示”复选框中单击鼠标，去除复选框中的选中标记“√”，下一次运行 Visual C++ 6.0 将不再出现此对话框。单击“关闭”按钮关闭此对话框，进入 Visual C++ 6.0 开发环境，如图 1-2 所示。

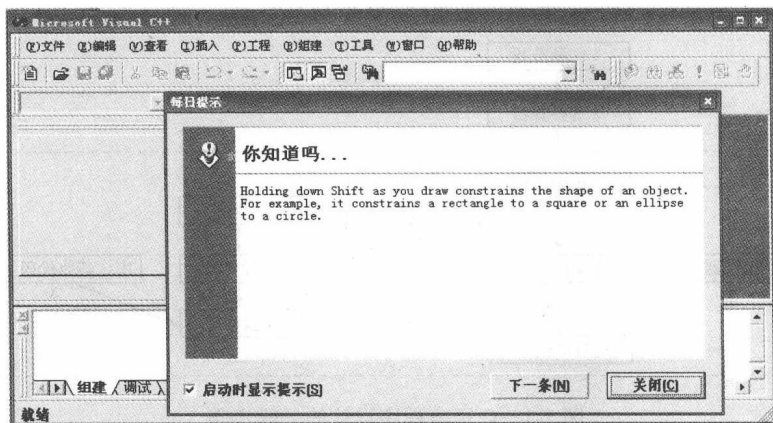


图 1-2 Visual C++ 6.0 启动窗口

3. 添加 C 语言源程序

（1）单击“标准”工具栏上的“新建”按钮，打开一个新的文档窗口，在这个窗口中输入 [例 1.1] 中的 C 程序代码。

（2）选择“文件”→“保存”菜单命令或按快捷键 Ctrl+S 或单击“标准”工具栏的按钮，弹出“保存为”文件对话框。将文件定位到“D:\C 程序”文件夹中并保存，文件名指定为 Ex_1.C（注意扩展名.C 不能省略）。

此时在文档窗口中所有代码的颜色都将发生改变，这是 Visual C++ 6.0 的文本编辑器所具有的语法颜色功能，绿色表示注释，蓝色表示关键字等，如图 1-3 所示。

4. 编译和运行

（1）单击“编译”工具条上的“组建”工具按钮或直接按快捷键 F7，系统会弹出一个对话框，询问是否为该程序创建默认的活动工作区间文件夹。单击“是”按钮，系统开始对源程序 Ex_1.C 进行编译、连接，同时在输出窗口中显示有关信息。如果出现错误提示，要根据错误提示信息修改源程序 Ex_1.C 的错误，再单击“组建”工具按钮，直到出现信息“ex_1.exe - 0 error(s), 0 warning(s)”，这就表示可执行文件 Ex_1.exe 已经正确无误地生成了，

如图 1-3 所示。

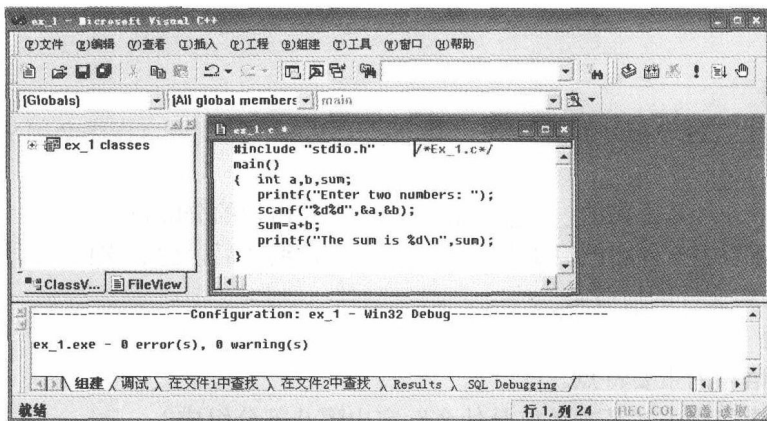


图 1-3 源程序输入、编译和运行窗口

(2) 单击“编译”工具条上的“执行程序”工具按钮或直接按快捷键 Ctrl+F5, 就可以运行刚刚生成的 Ex_1.exe 了, 弹出如图 1-4 所示的控制台窗口。

此时等待用户输入两个数。当输入“10 20”并按 Enter 键后, 控制台窗口情况如图 1-5 所示。

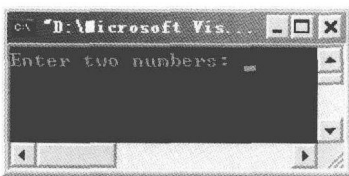


图 1-4 执行程序控制台窗口

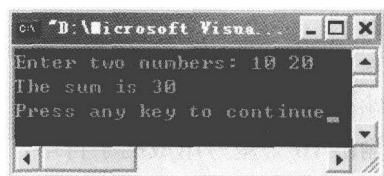


图 1-5 输出结果的控制台窗口

其中,“Press any key to continue”是 Visual C++自动加上去的,表示 Ex_1 运行后,按任意键将返回到 Visual C++开发环境。

对于一个已经存在的 C 语言源程序,可以在 Visual C++ 6.0 集成开发环境下,使用“文件”菜单下的“打开”命令,选择要打开的源程序文件,再单击“打开”按钮,进入到如图 1-3 所示的界面,然后再对源程序进行编译、连接和运行。

本章小结

本章首先介绍了程序、程序设计、程序设计语言的基本概念;明确了计算机不能直接读懂高级语言,必须由编译器或者解释器把高级语言翻译成计算机可以读懂的机器语言;介绍了 C 语言的发展历史及特性, C 语言程序的基本结构;还介绍了 Visual C++ 6.0 集成环境 and 在此环境下 C 语言编辑、连接和运行的基本方法和步骤。

习 题

一、填空题

1. 一个 C 语言程序必须有且只有一个_____, 一个 C 语言程序必须从_____开始执行。
2. C 语言中分号是语句的_____标志, 而不是语句的分隔符。
3. C 语言源程序文件名的后缀是_____; 经过编译后, 生成文件的后缀是_____; 经过连接后, 生成文件的后缀是_____。

二、简答题

1. 简述 C 语言的主要特点。
2. 构成 C 语言程序的基本单位是什么? 它由哪几部分组成?
3. 简述 C 语言程序调试、运行的步骤。

三、编程题

1. 参照本章示例, 编写一个 C 语言程序, 输出以下信息:

```
*****  
very good!  
*****
```

算法提示:

- (1) 输出 10 个星号组成的字符串, 换行。
 - (2) 输出 “very good!”, 换行。
 - (3) 输出 10 个星号组成的字符串, 换行。
2. 参照本章示例, 编写一个 C 语言程序: 输入圆的半径, 输出其周长。

算法提示:

- (1) 申请两个存储单元分别用 r 和 l 表示, 用来存放数据。
- (2) 读入圆的半径值, 存入 r 中。
- (3) 求圆的周长, 将值存入 l 中, 输出 l 的值。

第 2 章 数据类型、运算符与表达式

教学目标

掌握 C 语言基本数据类型及变量定义方法,掌握 C 语言运算符及由运算符构成的表达式。

本章要点

- 数据类型及变量定义
- 运算符及表达式

数据和操作是构成程序的两个要素,正如著名计算机科学家沃思(Nikiklaus Wirth)提出的:数据结构+算法=程序。数据是程序加工处理的对象,也是加工的结果。数据类型是计算机领域中一个非常重要的概念,它决定了数据的表示范围、数据在内存中的存储分配及数据能够进行的运算。在本章将介绍 C 语言的基本数据类型,各种运算符、表达式及运算时的相关规定。

2.1 基本标识符

简单地说,标识符就是一个名字。用来表示程序中用到的变量、函数、类型、数组、文件及符号常量等的名称。例如,每个人的姓名,就是每个人所对应的标识符。C 语言中的标识符可以分为 3 类:关键字、预定义标识符和用户定义标识符。

2.1.1 关键字

关键字又称保留字,是 C 语言规定的具有特定意义的标识符。它们在程序中都代表着固定的含义,不能另作他用。C 语言中共有 32 个关键字(见附录 B),分为以下 4 类:

- (1) 标识数据类型的关键字(14 个): int、long、short、char、float、double、signed、unsigned、struct、union、enum、void、volatile、const。
- (2) 标识存储类型的关键字(5 个): auto、static、register、extern、typedef。
- (3) 标识流程控制的关键字(12 个): goto、return、break、continue、if、else、while、do、for、switch、case、default。
- (4) 标识运算符的关键字(1 个),即 sizeof。

2.1.2 预定义标识符

预定义标识符是一类具有特殊含义的标识符,用于标识库函数名和编译预处理命令。系统允许用户把这些标识符另作他用,但这将使这些标识符失去系统规定的原意。为了避免误解,建议不要将这些预定义标识符另作他用。C 语言中常见的预定义标识符有以下几种:

- (1) 编译预处理命令,有 define、undef、include、ifdef、ifndef、endif、line、if、else 等。
- (2) 标准库函数。
数学函数: sqrt、fabs、sin、cos、pow 等。
输入/输出函数: scanf、printf、getchar、putchar、gets、puts 等。