工业和信息化普通高等教育“十二五”规划教材立项项目
21世纪高等教育计算机规划教材

COMPUTER

C语言程序设计教程（第2版）

C Programming Language Tutorials

杨有安 曹惠雅 鲁丽 陈维 编

以计算思维能力为教学指导
以学以致用实训教学为原则
以培养自主开发能力为目标

人民邮电出版社
POSTS & TELECOM PRESS

工业和信息化普通高等教育“十二五”规划教材立项项目
21世纪高等教育计算机规划教材

COMPUTER

C语言程序设计教程（第2版）

C Programming Language Tutorials

杨有安 曹惠雅 鲁丽 陈维 编

人民邮电出版社
北京

图书在版编目（CIP）数据

C语言程序设计教程 / 杨有安等编. -- 2版. -- 北京 : 人民邮电出版社, 2014.2
21世纪高等教育计算机规划教材
ISBN 978-7-115-33758-0

Ⅰ. ①C… Ⅱ. ①杨… 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313047号

内 容 提 要

本书根据全国高等学校计算机基础教育研究会发布的计算机基础教育的纲领性文件中有关“程序设计”课程的教学要求及人才培养的新要求编写而成。全书共 11 章，主要内容包括 C 语言的基本概念、变量、运算符、表达式、顺序结构、分支结构、循环结构、数组、函数、指针、结构体、联合体和枚举类型、预处理和标准函数、文件、数据结构和数据抽象。同时，还将介绍程序设计的基本方法和常用算法。

本书以计算思维模式进行计算机类课程教学的形式作为编写指导，内容全面，由浅入深，详略得当，注重实践，实例丰富，面向应用。各章附有适量的习题，便于自学。另外，针对书中各章内容和上机实验，本书还配有辅导教材《C 语言程序设计实践教程（第 2 版）》，引导读者学习和掌握各章节的知识。全书贯彻传授知识、培养能力、提高素质的教学理念。

本书为高等学校非计算机专业“C 语言程序设计”课程的教材，也可作为 C 语言初学者及计算机二级考试者和计算机工程技术人员的学习参考书。

◆ 编　　杨有安　曹惠雅　鲁　丽　陈　维
责任编辑　武恩玉
责任印制　彭志环　焦志炜

◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
大厂聚鑫印刷有限责任公司印刷

◆ 开本：787×1092　1/16
印张：18.75　　2014 年 2 月第 2 版
字数：491 千字　　2014 年 2 月河北第 1 次印刷

定价：42.00 元

读者服务热线：(010)81055256　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京崇工商广字第 0021 号

第 2 版前言

本书在第 1 版的基础上按“以计算思维能力的培养为目标”的教学模式进行修改。

“C 语言程序设计”是一门非常重要的计算机课程，其重要性不仅仅体现在一般意义上的程序编制，更体现在引导学生实现问题求解思维方式的转换，即培养学生的计算思维能力。本书从初学者的需求出发，体现“学生易学，教师易用，变应试为应用”的编写理念。

本书编写拟达到的目标：体现计算机基础教学是培养大学生综合素质和创新能力不可或缺的重要环节，是培养复合型创新人才的重要组成部分。在新形势下，计算机基础教学的内涵在快速提升和不断丰富，进一步推进计算机基础教学改革，适应计算机科学技术发展的新趋势，是国家创新工程战略对计算机教学提出的重大要求。彻底改变长期以来存在的“计算机只是工具”、“计算机就是程序设计”和“计算机基础课程主要讲解软件工具的应用”等片面认识。本书希望能在计算机基础教学过程中体现出以人为本、传授知识、培养能力、提高素质、协调发展的现代教育理念，在大力培养学生的计算思维能力等方面尽微薄之力，共托明日朝阳。

本书结合“以计算思维能力的培养为目标”的教学模式有针对性的做了如下几点修改：

（1）增加程序指令、语法及语言功能等内容作为程序设计概念的铺垫；

（2）增加数据存储的内容使学生更清楚程序执行的过程中计算机内数据处理的情况；

（3）增加数据输入的常见问题，让学习内容更贴近学生实践环节；

（4）增加内存与外存数据交流的内容使学生对程序执行过程中的信息交互情况一目了然；

（5）改正原教材中的个别错误，更加注意在概念表述上的措词准确和版面布局的规范。

全书分为 11 章，其中第 1 章、第 2 章和第 4 章由曹惠雅编写，第 3 章、第 8 章和第 10 章由鲁丽编写，第 7 章、第 9 章、第 11 章和附录 C 由陈维编写，第 5 章和第 6 章由杨有安编写。杨有安负责全书的统稿工作。本书在编写的过程中得到中科技大学文华学院各级领导的大力支持，在此表示衷心的感谢。

本书还同时出版了一本配套教材《C 语言程序设计实践教程（第 2 版）》（杨有安等编，人民邮电出版社出版），两者互为补充，相辅相成，对读者掌握程序设计的基本知识，提高程序设计的应用能力十分有益。配套教材按照主教材的章节顺序，对各章重点及难点进行总结，对重点难点题型进行分析，并附有大量的练习，以此帮助读者加深对 C 语言程序设计基础知识的理解，每章结束部分附有参考答案，方便读者进行自测。另外还配有相应的实验内容以增强实践性环节的学习。

由于编者水平有限，加之时间仓促，书中难免有不足之处，敬请读者批评指正。

编　者

2013 年 10 月

目 录

第 1 章 C 语言概述

C 语言是 Combined Language（组合语言）的简称，是一种被广泛重视且应用普遍的计算机程序设计语言。它既可以用来编写系统软件，也可以用来编写应用软件。

对于将 C 语言作为第一门编程语言（Programming Language）的读者来说，最关心的问题无疑是如何能尽快学会用 C 语言进行程序设计。要做到这一点，首先需要对程序设计语言有所了解，其次还要通过不断的编程实践，去领会和掌握程序设计的基本思想和方法。在此，建议读者先从看懂书中的程序做起，而后再模仿书中的程序试着改写或编写程序，循序渐进，直到能独立地编写程序并解决一些较复杂的问题。

为了使读者能逐步地从简单的模仿中体会到程序设计的基本思想和方法，本章将简要介绍程序设计语言的功能、语法要素、C 语言的特点、词汇以及 C 语言程序的结构和运行环境等知识。

1.1 程序与程序设计语言

计算机程序（Program）是人们为解决某种问题用计算机可以识别的代码编排的一系列加工步骤。计算机能够严格按照这些步骤去执行，包括计算机对数据的处理。程序的执行过程实际上是对程序所表示的数据进行处理的过程。一方面，程序设计语言提供了一种表示数据与处理数据的功能；另一方面，编程人员必须按照语言所要求的规范（即语法规则）进行编程。

1.1.1 程序与指令

程序最根本的功能是对数据的处理，计算机最基本的处理数据的单元就是计算机指令。单独的一条指令本身只能完成计算机的一个最基本的功能，那么，计算机能实现的指令的集合则成为计算机的指令系统，而一系列计算机指令的有序组合就构成了程序。

程序在计算机中是以 0、1 组成的指令代码来表示的，即程序实际上是 0、1 的有序组合，这个序列能够被计算机直接识别。程序和数据一样，共同存放在存储器中。当程序要运行时，当前准备运行的指令从内存被调入 CPU 中，由 CPU 处理这条指令。

如果程序设计者直接用 0、1 的代码来编写计算机指令，那将是一件非常令人难以忍受的事情，所以，程序设计语言就应时而生。用程序设计语言来描述程序，同时应用一种软件（如编译系统）将其描述的程序转换成计算机能够直接识别的指令序列。

总的来说，计算机程序是人们为了解决某种问题，用计算机可以识别的代码编排的一系列数据处理步骤，计算机将严格按照这些步骤去做。

1.1.2 程序设计语言的功能

程序设计语言，通常简称为编程语言，它是一种被标准化的交流技巧，是一组用来定义计算机程序的语法规则，用来向计算机发出指令，它可以让程序员准确地定义计算机所需要使用的数据，并精确地定义在不同情况下所应当采取的行动。

程序设计语言必须具有数据表达和数据处理（也称为流程控制）的能力。

1. 数据表达

数据种类多种多样，而语言本身的描述能力总是有限的。为了使程序设计语言能充分、有效地表达各种各样的数据，一般将数据抽象为若干种类型。数据类型（Data Type）就是对某些具有共同特点的数据集合的总称。比如大家常说的整数、实数等就是数据类型的例子。数据类型涉及两方面的内容：该数据类型代表的数据是什么（即数据类型的定义域）？能在这些数据上做什么（即操作，或称运算）？比如，整数类型所包含的数据有 2，–2,0,100，…，而+，–，*，/等就是作用在整数上的运算。

在程序设计语言中，一般都需事先定义几种基本的数据类型，以供程序员直接使用，如整型、实型、字符型等。同时，为了使程序员能更充分地表达各种复杂的数据，程序设计语言还提供了构造新的具有数据类型的手段，如数组（Array）、结构（Structure）、指针（Pointer）等。

程序设计语言提供的基本数据类型以及构造复杂类型的手段，为有限能力的程序设计语言表达客观世界中的多种多样的数据提供了良好的基础。

2. 流程控制

程序设计语言除了能表达各种各样的数据外，还必须提供一种手段来表达数据处理的过程，即程序的控制过程。程序的控制过程通过程序中的一系列语句来实现。

当要解决的问题比较复杂时，程序的控制过程同样会变得十分复杂。一种常用的程序设计方法是：将复杂程序划分成若干个相对独立的模块（Module），使完成每个模块的工作变得单纯而明确，在设计一个模块时不受其他模块的制约。同时，通过现有模块积木式的扩展就可以形成复杂的、更大的程序模块或程序。这种程序设计方法就是结构化的程序设计方法（Structured Programming）。C 语言就是支持这种设计方法的典型语言。

在结构化程序设计方法中，一个模块可以是一条语句（Statement）、一段程序或一个函数等。一般来说，从程序流程的角度看，模块只有一个入口和一个出口。这种单入单出的结构为程序的调试（Debug，又称查错）提供了良好的条件。

按照结构化程序设计的特点，任何程序都可以将模块通过三种基本的控制结构进行组合来实现。这三种基本的控制结构分别是：顺序结构、分支结构和循环结构。

顺序结构（Sequential Structure）：按照程序的书写先后顺序，执行完一个程序模块后，再顺序执行下一个模块。

分支结构（Branch Structure）：又称选择结构。在程序执行过程中，根据不同的条件来选择所要执行的模块，即判断某种条件，若条件满足就执行某个模块，否则就执行另一个模块。

循环结构（Loop Structure）：是指反复执行某个模块的过程。当然，重复执行这些模块通常是有条件的，只有条件满足时才会去重复执行相应的模块。

1.1.3 程序设计语言的语法

一般把用程序设计语言编写的未经编译的程序称为源程序（Source Code，又称源代码），而

源程序的编写必须符合相应语言的语法（Grammar）。那么，从语法的角度来说，源程序实际上是一个字符序列。这些字符序列按顺序分别组成了一系列的“单词”。这些“单词”是为了按照一定的语法规则构成语言的各种成分而规定的。下面分别介绍 C 语言中的一些常用词汇和语法单位。

1. C 语言的词汇

（1）C 语言字符集。组成 C 语言源程序代码的基本字符称为 C 语言字符集，它是构成 C 语言的基本元素。C 语言允许使用的基本字符有：

① 大小写英文字符：A ~ Z，a ~ z。

② 数字字符：0 ~ 9。

③ 特殊字符：+ = — _(下划线) ( ) * & ^ % # ! , . ; : ? ' " ~ \ | / < > { } [ ]。

④ 不可打印的字符：空格、换行符、制表符、响铃符。

一般的 C 语言源程序仅仅包含以上字符集合中的字符，在具体的 C 语言编译系统中可对上述字符集合加以扩充。

（2）关键字。关键字是具有特定含义的、专门用来说明 C 语言的特定成分的一类单词。例如，关键字 int 用来定义整型变量，而关键字 float 则用来定义实型变量。C 语言的关键字都用小写字母书写，不能用大写字母书写。例如，关键字 int 不能写成 Int。由于每个关键字都有特定的含义，所以不能作为用户程序中的变量名和函数名等，否则会产生编译错误。在 C89 标准中共有 32 个关键字：

```
auto        break    case        char     const       continue    default
do          double   else        enum     extern      float       for
goto        if       int         long     register    return      short
signed      sizeof   static      struct   switch      typedef     union
unsigned    void     volatile    while
```

在新的 C99 标准中，又增加了 5 个关键字：

```
_Bool      _Complex   _imaginary   inline    restrict
```

（3）标识符。计算机程序处理的对象是数据，程序用来描述数据处理的过程。在程序中，通过名字建立对象定义与使用的关系。为了满足这种需要，每种程序语言都规定了在程序中名字描述的规则。在 C 语言中用于标识名字的有效字符序列称为标识符，对标识符作了如下规定：

① 标识符的第一个字符必须是英文字母或下划线(_)。

② 如果第一个字符后面还有字符序列，则它应是英文字母、下划线符或数字组成的序列。标识符中的英文字母大小写是有区别的，如标识符 abc 与标识符 ABC 不相同。为了便于读者对标识符有进一步的认识，下面列举若干正确的标识符和不正确的标识符：

正确的标识符：

```
Abc     abc     _Abc     _4a5
```

不正确的标识符：

```
A?          （含有不合法字符“?”）
2abc        （第一个字符不允许为数字）
a b         （标识符中不允许有空格）
yes/no      （含有不合法字符“/”）
пr          （“п”为不合法字符）
```

标识符中有效字符个数（也称长度）视系统不同而不同。例如，Turbo C 规定前 32 个字符有效，超过的部分忽略。比如，对于 8 个字符有效的标识符而言，identifi 与 identifier 被视为同一标识符，因后者中的 er 已被忽略。

以后将会看到，标识符用来为变量、符号常量、数组、函数等取名。使用时，标识符的选择由程序员自定，但是不能与关键字相同。另外，为了增加程序的可读性，选择标识符时应遵循“见名知义”的原则，即选择描述性的标识符，标识符应尽量与所要命名的对象间有一定的联系，以助于识别和记忆。例如：

```
longth    （表示长度）
time      （表示时间）
pi        （表示圆周率 π）
```

（4）保留标识符。保留标识符是系统保留的一部分标识符，通常用于系统定义和标准库函数的名字。例如，以下划线开始的标识符通常用于定义系统变量。虽然它们也是合法的标识符，但用作一般标识符时可能会出现运行错误，因此不能使用这些标识符来定义自己的变量。

（5）注释。在 C 语言程序中，注释部分的格式是：

```
/*注释内容*/   或   //注释内容
```

注释不是程序代码，是对程序解释说明的标注，它可以是任何可显示的字符，不影响程序的编译和运行，程序编译时编译程序把注释作为空白符跳过而不予处理。另外，注释不允许嵌套。

例如：/*学生成绩管理程序*/、//My c program

在程序中插入适当的注释，可以使程序容易被人理解。

2. C 语言的主要语法单位

（1）变量定义。不同的变量有数据类型之分，在声明变量的时候一定要对其类型加以说明。变量类型的不同，说明其在计算机内存中所占的存储空间大小也不同。声明变量的一般格式为：

```
类型说明符  变量名;
例如: int i;   /*声明了一个整型变量 i*/
```

（2）表达式。由运算符及其运算对象可以组成形形色色的表达式。如：3.14*3*sin(x)。表达式中的运算符有运算优先级，如：表达式 2+3*4–4/4 中，应先执行运算符*和/，再执行运算符+和–。

（3）语句。语句是程序最基本的执行单位，程序的功能就是通过执行一系列的语句来实现的。C 语言提供了多种语句，大致可分为五类：表达式语句、函数调用语句、控制语句、复合语句、空语句。

① 表达式语句。表达式语句由表达式末尾加上分号“;”组成。其一般形式为：表达式;

执行表达式语句就是计算表达式的值。例如：

```
m=2;     赋值表达式语句
m+n;     算术表达式语句
m>n;     关系表达式语句
i++;     增量表达式语句
```

② 函数调用语句。由函数名、实际参数加上分号“;”组成。其一般形式为：函数名(实际参数表);

执行函数语句就是调用函数体并把实际参数赋予函数定义中的形式参数，然后执行被调函数体中的语句，求取函数值。

调用库函数，输出字符串。

例如：

```
printf("%d",m);     /*调用名为 printf 的标准库函数*/
```

③ 控制语句。控制语句用于控制程序的流程，以实现程序的各种结构方式。它们由特定的语

句定义符组成。C 语言有九种控制语句，分成以下 3 类：

- 条件判断语句。条件判断语句包括：if 语句、switch 语句。
- 循环执行语句。循环语句包括：for 语句、while 语句、do while 语句。
- 跳转语句。跳转语句包括：break 语句、continue 语句、return 语句和 goto 语句。其中，goto 语句应尽量少用，因为这不利结构化程序设计，滥用它会使程序流程无规律、可读性差。

④ 复合语句。把多个相关语句用一对花括号“{}”括起来，组成的一个语句就称为复合语句。在程序中，应把复合语句当作是单条语句，而不是多条语句。例如：

```
{
   temp=x;          /*将 x 的值赋予 temp*/
   x=y;             /*将 y 的值赋予 x*/
   y=temp;          /*将 temp 的值赋予 y*/
}
```

这就是一条复合语句。复合语句内的各条语句都必须以分号“;”结尾，但是在花括号“}”外却不能加分号。

⑤ 空语句。只有分号“;”组成的语句称为空语句。空语句是什么也不执行的语句，在程序中空语句可用来作空循环体。

例如：while(getchar()!='\n'); 本语句的功能是，只要从键盘输入的字符不是回车则重新输入。这里的循环体为空语句。

（4）函数定义。函数是完成特定任务的独立模块，是 C 语言唯一的一种子程序形式。函数的目的通常是接收 0 个或多个数据(称为函数的参数),并返回 0 个或 1 个结果(称为函数的返回值)。函数的使用主要涉及函数的定义与调用。

函数定义的主要内容是通过编写一系列语句来规定其所完成的功能。完整的函数定义涉及函数头和函数体。其中，函数头包括函数的返回值类型、函数名、参数类型；而函数体是一个程序模块，规定了该函数所具有的功能。函数调用则通过传递函数的参数并执行函数定义所规定的程序过程，以实现相应的功能。以下是函数定义的一个简单例子。

```
int max(int m,int n)          /*函数头：函数类型说明符  函数名(函数参数列表)*/
{      /*函数体的开始*/
   int x;                     /*声明一个整型变量 x*/
   if(m>n)
      x=m;                    /*判断 m、n 的大小，将其中的较大者值赋予变量 x*/
   else
      x=n;
   return x;                  /*结束函数调用，并返回变量 x 的值*/
}      /*函数体的结束*/
```

（5）输入与输出。C 语言没有输入输出语句，它通过调用系统库函数中的有关函数（如：printf()、scanf()函数等）实现数据的输入和输出，这种处理方式为 C 语言在不同硬件平台上的可移植性提供了良好的基础。相关的输入、输出函数的使用及其功能将会在后面陆续做讲解。

1.2　C 语言的发展与特点

C 语言作为计算机编程语言，具有功能强、语句表达简练、控制和数据结构丰富灵活、程序

时空开销小等特点。它既具有 Pascal、FORTRAN、COBOL 等通用程序设计语言的特点，又具有汇编语言（Assemble Language）中位（bit）、地址（Address）、寄存器（Register）等概念，拥有其他许多高级语言所没有的低层操作能力；既适合于编写系统软件，也可用来编写应用软件。C 语言的这些特点与其发展过程是密不可分的。

1.2.1 C 语言的发展

早期的系统软件（包括操作系统）主要用汇编语言编写，因而程序与计算机硬件的关系十分密切，使程序的编写难度大、可读性差、难于移植。为顺应时代的发展，需有一种与计算机硬件关系不紧密的高级语言（High-level Programming Language）用于编程。

19 世纪 70 年代初，贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计并实现了 C 语言，当时的 C 语言只是为描述和实现 UNIX 操作系统提供一种工作语言。后来，C 语言经过多次修改，逐渐形成了不依赖于具体机器的 C 语言编译软件，称为如今广泛应用的计算机语言之一。

目前，在各种类型的计算机和操作系统下，有不同版本的 C 语言编译程序，这些 C 编译程序各有特点。但一般来说，1978 年，B.W.Kernighan 和 D.M.Ritchit 合著的《THE C PROGRAMMING LANGUAGE》一书是各种 C 语言版本的基础，被称为旧标准 C 语言。由于没有统一的标准，使得不同版本的 C 语言之间出现了许多不一致的地方。为了改变这种状况，1983 年美国国家标准研究所（American National Standards Institute，ANSI）为 C 语言制定了第一个 ANSI 标准，称为 ANSI C。1987 年美国国家标准研究所又公布了新的 C 语言标准，称为 87 ANSI C。这个标准在 1989 年被国际标准化组织（ISO）采用，被称为 ANSI/ISO Standard C（即 C89）。B.W.Kernighian 和 D.M.Ritchie 根据这个标准，重写了他们的经典著作，并发表了《The C Programming Language, Second Edition》。

1995 年，C 语言中增添了一些新的函数，使之具有了 C++的一些特征，使 C89 成为 C++的子集。1999 年推出的 C99 在基本保留 C 语言特征的基础上，增加了一系列面向对象的新特征。C 语言也就从面向过程的语言发展成为面向对象的语言。

C 语言是 C++的基础，C++语言和 C 语言在很多方面是兼容的。因此，掌握了 C 语言，可为将来学习 C++打下坚实的基础。本教材使用 Visual C++6.0 作为 C 语言程序的运行环境。

1.2.2 C 语言的主要特点

C 语言之所以能存在和发展，并具有强大的生命力主要因为它具有强大的功能。

（1）C 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，语法控制不严格，表达式简练、灵活、实用。

（2）运算符丰富。C 语言中共有 34 个运算符，它们与丰富的数据类型相结合，构成了各种各样的表达式，实现了在其他高级语言中难以实现的各种复杂运算。

（3）数据结构丰富。C 语言的数据类型有：整型、实型、字符型、数组类型、指针类型、结构体类型、联合体类型等，能用来实现各种复杂的数据类型的运算。尤其是指针类型数据的引入，使程序运行效率更高。另外 C 语言具有强大的图形功能，支持多种显示器和驱动器，且计算功能、逻辑判断功能强大。

（4）C 语言是结构式语言。结构式语言的显著特点是程序代码模块化。C 语言的主要成分是函数，函数是 C 语言程序的基本结构单位，函数之间彼此独立，程序的许多操作可由不同功能的函数有机组装而成，从而容易达到结构化程序设计中模块的要求。另外，C 语言还提供了一套完

整的控制语句（如顺序、分支、循环）和构造数据类型（如结构、数组、指针），使程序流程与数据描述也具有了良好的结构性。C 语言的这种结构化方式使程序层次更清晰，使用、维护及调试更方便。

（5）C 语法限制不太严格、程序设计自由度大。一般的高级语言语法检查比较严，能够检查出几乎所有的语法错误。而 C 语言则放宽了语法检查，允许程序编写者有较大的自由度。例如：对数据类型检查不严格，表达式出现二义性；对数组下标越界不作检查等。因此，在程序设计中，程序员不要过分依赖编译器的语法检查。

（6）C 语言允许直接访问物理地址。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，能够像汇编语言一样对位、字节和地址进行操作，还可以用来编写系统软件。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语言称为“高级语言中的低级语言”。

（7）C 语言程序生成代码质量高。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%。

（8）C 语言适用范围大、可移植性好。C 语言编写的程序中没有依赖于硬件的输入输出语句，程序的输入输出功能是通过调用输入输出函数实现的，而这些函数是由系统提供的独立于 C 语言的程序模块，所以编写好的 C 源程序基本上不作修改就可以用于各种型号的计算机和各种操作系统，从而便于硬件结构不同的计算机之间实现程序的移植。

1.3　C 程序的结构

用 C 语言编写的程序称为 C 语言源程序，简称为 C 程序。为了说明 C 语言源程序的结构特点，先看以下几个程序。这几个程序由简单到复杂，虽然有关内容还未介绍，但可以从中了解到 C 语言源程序在基本组成结构上的特点及其书写风格。

1.3.1　简单C程序举例

【例 1-1】编写一个 C 语言程序，输出“good morning!”。

程序如下：

```
/*c1_1.c*/
#include <stdio.h>                    /*为文件包含，其扩展名为.h，称为头文件*/
void main()
{
   printf("good morning!\n");         /*通过显示器输出 good morning!*/
}
```

说明：

（1）C 语言程序中可以随时使用注释，但注释内容不参与编译。

（2）#include 称为文件包含命令或编译预处理命令，#include <stdio.h>是文件包含，其意义是把尖括号<>或引号“”内指定的文件包含到本程序来，成为本程序的一部分。被包含的文件通常是由系统提供的，其扩展名为.h，称为头文件或首部文件。C 语言的头文件中包括了各个标准库函数的函数原型。因此，凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。需注意：编译预处理命令的末尾不加分号。详细内容将在后面章节介绍。

（3）main 是主函数的函数名，表示这是一个主函数。每个完整的 C 语言源程序都必须有主函数，且只能有一个主函数(main 函数)，程序总是从 main 函数开始执行，并终止于 main 函数。函数体由一对大括弧“{ }”括起来，其间一般包括程序的说明部分和执行部分。

（4）printf 函数是一个由系统定义的标准函数，可在程序中直接调用。其功能是将输出的内容送到显示器显示。

该程序正确执行后，会在显示器上显示输出：

```
good morning!
```

【例 1-2】从键盘输入两个整数，输出平均数。

```
/*c1_2.c*/
#include<stdio.h>
void main()
{
   int yw,sx,sum;                          /*定义 3 个整型变量*/
   printf("Input two numbers:");           /*显示提示信息*/
   scanf("%d%d",&yw,&sx);                  /*输入 yw，sx 值*/
   sum=yw+sx;                              /*求出 yw 与 sx 之和，并把它赋予变量 sum*/
   printf("average=%d\n",sum/2);           /*输出语文和数学的平均成绩*/
}
```

程序分析：

（1）该程序中使用了 yw、sx 和 sum 3 个变量，所有变量在使用之前必须先定义。

（2）scanf 函数是一个由系统定义的标准函数，可在程序中直接调用。它的功能是输入变量 x 和 y 的值。&yw 和&sx 中“&”的含义是“取变量地址”，表示将从键盘输入的两个值分别存放到地址标志为 yw 和 sx 的存储单元中。

（3）“%d”是输入/输出数据的“格式说明”，用来指定输入/输出时的数据类型和格式，%d 表示“十进制整数类型”，在执行输出时，屏幕上显示一个十进制整数值。

（4）sum=yw+sx 为赋值表达式，表示将 x+y 之和赋值给 sum 变量所标识的存储单元。

该程序正确执行后，会在显示器上显示输出：

```
Input two numbers:89  95
average=92
```

【例 1-3】输入两个整数，进行比较后将较大数输出。

```
/*c1_3.c*/
#include<stdio.h>
void main()
{
   int x,y,z;                              /*定义 3 个整型变量*/
   int max(int a,int b);                   /*函数类型说明*/
   printf("Input two number:");            /*显示提示信息*/
   scanf("%d%d",&x,&y);                    /*输入 x，y 值*/
   z=max(x,y);                             /*调用 max 函数*/
   printf("max=%d\n",z);                   /*将较大数输出*/
}

int max(int a,int b)                       /*定义 max 函数*/
{
```

```
    int c;                                  /*定义一个整型变量*/
    c=a>b?a:b;                              /*求出变量 c 的值*/
    return c;                               /*将 c 的值返回到主调函数*/
}
```

程序分析：

（1）本程序包括两个函数：主函数 main 和自定义函数 max。max 函数的作用是将 a 和 b 中较大者的值赋于变量 c；return 语句将 c 的值返回主调函数。

（2）在调用 max 函数时，将实际参数 x 和 y 的值分别对应传给 max 函数中的形式参数 a 和 b。

（3）a>b?a:b 是一个条件表达式，当 a>b 成立时，a>b?a:b 表达式的值为 a 的值；反之则为 b 的值。详细内容将在第 2 章介绍。

该程序正确执行后，会在显示器上显示输出：

```
Input two numbers:8   2
max=8
```

本例中涉及函数调用、实际参数和形式参数等概念，如果读者对此不大理解，可先不予以深究，第 6 章中将会有详尽介绍。

1.3.2　C 语言程序的结构特点

通过上面 3 个 C 语言源程序，可以看出其基本结构具有以下几个特点：

（1）C 语言源程序的基本组成单位是函数。所有的 C 语言程序都由一个或多个函数构成，其中 main 函数必须有且只能有一个。

（2）main()函数可以出现在 C 源程序的任何位置，程序执行时总是从 main()函数开始，又在 main()函数结束。主函数可以调用标准库函数（如 printf()、scanf()等）和用户自定义函数，但标准库函数和用户自定义函数却不能调用主函数。

（3）源程序中的预处理命令通常放在源文件或源程序的最前面。

（4）分号“;”是 C 语句的必要组成部分。每个语句或每个变量说明都必须以分号结尾。但预处理命令、函数头和花括号“{”、“}”后面不能加分号。

（5）标识符、关键字之间必须至少加一个空格以示分隔。

（6）可以在程序的任何位置用/*注释内容*/或//注释内容的形式对程序或语句进行注释，以增加程序的可读性。

1.3.3　书写程序时应遵循的规则

C 语言程序的书写格式非常自由，但从书写清晰，便于阅读、理解、维护的角度出发，建议在书写 C 语言程序时遵循以下几个规则：

（1）一个说明或一条语句占一行。

（2）用{}括起来的部分，通常表示了程序的某一层次结构（如函数体、循环体、复合语句等）。{}一般与该结构语句的第一个字母对齐，并单独占一行。

（3）低一层次的语句或说明比高一层次的语句或说明向后缩进若干格后书写，同一层次的语句或说明左对齐，以增强程序编写的层次感，增加程序的可读性。

（4）函数块与函数块之间加一空行分隔，以便清楚地分出程序中有几个函数。

在编程时应力求遵循上述规则，以养成良好的编程习惯。

1.4 Visual C++ 6.0 上机简介

Visual C++（简称 VC++）是美国 Microsoft 公司开发的 Microsoft Visual Stutio 的一部分，是一个基于 Windows 操作系统的可视化、面向对象且使用广泛的 C/C++集成开发环境（Integrated Development Environment，IDE）。它成功地将面向对象和事件驱动编程概念联系起来，并得到了很好的配合，使得编写 Windows 应用程序的过程变得简单、方便且代码量小。VC++ 6.0 集程序的代码编辑、编译、连接、调试于一体，给编程人员提供了一个完整、方便的开发界面和许多有效的辅助开发工具。

VC++ 6.0 的编辑环境包含了许多独立的组件，它们包括：文本编辑器、资源编辑器、C/C++编译器、连接器、调试器、AppWizard、ClassWizard、源程序浏览器以及联机帮助。所有这些构件的功能都隐藏在 VC++ 6.0 的菜单和工具条中。通过该集成环境，程序员可以观察和控制整个开发过程。

1.4.1 Visual C++ 6.0 集成开发环境简介

在已安装 Visual C++的计算机上，可以直接从桌面双击 Microsoft Visual C++图标，进入 Visual C++ 集成开发环境，或者单击【开始】|【程序】菜单，选择 Microsoft Visual Studio 6.0 中的 Microsoft Visual C++ 6.0 菜单项，进入 Visual C++ 6.0 集成开发环境。

Visual C++集成开发环境主要由标题栏、菜单栏、工具栏、项目工作区、编辑区、输出区等组成，如图 1-1 所示。

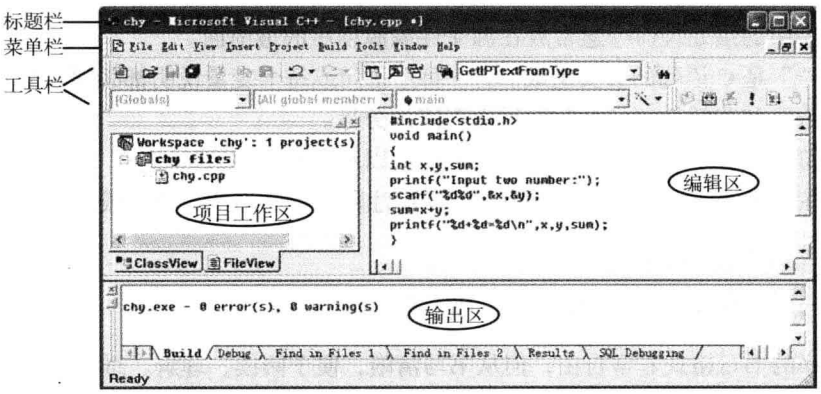

图 1-1 Microsoft Visual C++集成开发环境

1. 项目工作区

Visual C++集成开发环境以项目工作区来组织应用程序的工程，项目工作区文件扩展名为.dsw，这种类型的文件在 Visual C++中级别是最高的。项目工作区含有工作区的定义和工程中所包含文件的所有信息。所以，要打开一个工程，只需打开对应的项目工作区文件（*.dsw）即可。

项目工作区窗格位于屏幕左侧，包含 ClassView（类视图）、ResourceView（资源视图）和 FileView（文件视图）3 种视图。