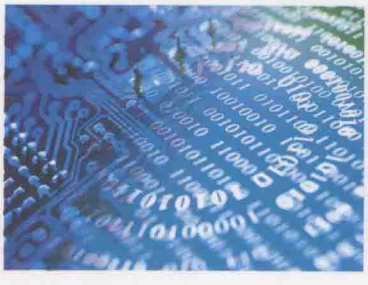
高等院校计算机专业教材

C语言程序设计


主　编：吴红庆　李春萍
副主编：周红明　沐士光

云南大学出版社
Yunnan University Press

C 语言程序设计

主编　吴红庆　李春萍

云南大学出版社

图书在版编目（CIP）数据

C 语言程序设计/吴红庆，李春萍主编．—昆明：
云南大学出版社，2010
ISBN 978－7－5482－0008－6
Ⅰ．①C…　Ⅱ．①吴…　②李…　Ⅲ．①C 语言—程序设
计　Ⅳ．①TP312
中国版本图书馆 CIP 数据核字（2010）第 018666 号

C 语 言 程 序 设 计

吴红庆　李春萍　主编

策划编辑：聂　滨
责任编辑：石　可
封面设计：刘　雨
出版发行：云南大学出版社
印　　装：昆明南方印业有限责任公司
开　　本：787×1092　1/16
印　　张：16
字　　数：399 千
版　　次：2010 年 2 月第 1 版
印　　次：2010 年 2 月第 1 次印刷
书　　号：ISBN 978－7－5482－0008－6
定　　价：30.00 元

社　　址：云南省昆明市一二·一大街 182 号云南大学英华园内（邮编：650091）
发行电话：0871－5033244　5031071
网　　址：http://www.ynup.com
E－mail：market@ynup.com

《C 语言程序设计》编写人员名单

主　编：吴红庆　李春萍

副主编：周红明　沐士光

参　编：（按姓名字母顺序排列）

贾纳豫　马　瑜　苏亚丽　杨增芳

前　言

C 语言是一种结构化、模块化的程序设计语言，数据类型丰富、使用灵活方便、可移植性好，运用领域广泛。它不仅具有高级语言的特点，还具有汇编语言的功能，既可以用于编写系统软件，又可以用于编写应用软件，具有极强的兼容性和良好的用户界面。20 世纪 90 年代以来，C 语言不仅被计算机专业人员所使用，而且也受到了广大计算机爱好者的青睐。目前在许多高校中，不仅计算机专业开设了 C 语言课程，而且通信工程、管理信息系统、应用数学等非计算机专业也开设了 C 语言课程。

由于 C 语言涉及的概念较复杂，规则繁多而使用灵活，容易出错，对于初学者来讲，学习起来比较困难。针对此现状，依据“以学生能力形成和发展为核心”的教育理念，结合多年的教学经验，我们对全书的内容作了精心安排，分解难点，由浅入深，采用通俗易懂的语言和丰富的例题帮助学习者理解复杂的概念，并在学习中逐步建立程序设计的理念。

本书内容共包括十一章，安排如下：

第一、二章是基础知识介绍，主要内容包括与计算机编程有关的基本概念、软件开发的基本方法、C 语言程序的构成要素及编程环境、基本数据类型及数据存储、基本运算符及表达式。第三至五章的主要内容是介绍结构化程序的三种基本结构，还包括算法及算法描述的基本方法。通过对这部分内容的学习，读者可以了解结构化程序设计的基本思想，能设计简单的算法并根据算法编写相应的 C 语言程序。其余章节主要介绍模块化程序设计的思想及实现方法，其内容包括函数、预处理命令、数组、指针、构造数据类型、位运算和文件，通过对相关章节的学习，学习者可以进一步理解结构化程序设计思想，掌握模块化程序设计的方法，提高程序设计的能力。

在 C 语言的教学过程中，函数是较难理解的一个内容，特别是函数调用过程中参数的传递方式，因而在内容的组织上，我们作了分减难点的处理：第六章引入函数的基本概念及实现后，只介绍函数参数的值传递方式；直到第七章引入数组后，才涉及数组名作为函数参数的地址传递方式。

C 语言程序设计是一门实践性非常强的课程。在教学过程中，学习者往往能够看懂书上的例子，但是一旦自己动手就觉得无从下手。为帮助学习者克服这个问题，增强学习信心，我们采用由易到难、循序渐进的方式，根据知识点的逐渐深入选择例题，并在每个章节的最后引入一个案例分析，选择与生活息息相关的问题，通过对问题的分析及编程解决，培养学习者建立软件开发的思想，提高分析问题和解决问题的能力，养成良好的程序设计风格，从而不断激发学习者的学习兴趣。

本书的每个章节均附有本章小结及习题，帮助学习者理解基本概念、掌握重点、消化

难点，可结合上机编程的训练，不断提高学习者程序设计的能力。

本书内容由浅入深，通俗易懂，文字精练，适合作为普通高等院校计算机程序设计语言的教学用书，也可作为其他人员自学的参考书。

本书第一章、第五章及附录由李春萍编写，第二章由沐士光编写，第四章由马瑜编写，第三章、第十一章由吴红庆编写，第六章由贾纳豫编写，第七章由杨增芳编写，第八章由苏亚丽编写，第九至十章由周红明编写，所有程序均在 VC + +6.0 环境下调试通过。

由于时间仓促及编者水平有限，书中难免有不妥及遗漏之处，敬请读者批评指正。

全书涉及的程序源代码、教学课件或习题答案，均可登录网站 site. yxnu. net，在“教学资源/C 语言教辅资料”栏目中下载。

编　者

2009 年 12 月

目　录

第一章　计算机编程及C语言概述

本章学习目标

☆ 熟悉与计算机编程有关的基础知识。
☆ 了解软件开发的基本方法，能给出解决问题的简要算法。
☆ 了解C语言出现的历史及特点。
☆ 熟悉Visual C++ 6.0环境下编写C程序的上机步骤。
☆ 了解C程序的基本要素，能读懂简单的C程序。

1.1　计算机与编程

20世纪70年代，随着微型计算机的出现，人类社会从工业社会迅速迈向信息社会。信息社会有两大支柱：计算机和网络。现代计算机接收、存储 、处理和输出信息，可处理包括数字、文本、图像和声音在内的各种类型的数据。计算机程序是计算机技术的根本，人们使用计算机能够“懂得”的语言和语法格式编写程序，实现与计算机的信息交流。编写程序所采用的语言就是程序设计语言。下面我们将选择最通用的编程语言——C语言来开始计算机编程的学习。

计算机系统由硬件和软件组成。软件是程序的集合，由一系列能解决问题的指令构成。硬件是用于执行所需计算的设备。从本质上看，现代计算机大都包含中央处理单元（Central Processing Unit，CPU）、主存储器（内存）、辅助存储器（硬盘、CD、U盘）、输入设备（键盘、鼠标、手写板、扫描仪）和输出设备（显示器、打印机和扬声器）。程序在执行前必须先从辅助存储器中传输到主存储器中；由中央处理器CPU访问并处理这些数据，处理结果暂时保存到主存储器中；计算机与外界的信息交流由输入/输出设备完成。本节主要介绍与C语言编程有关的几个概念。

1.1.1　内存及数据存取

内存是所有计算机的必需组件，可将内存想象为存储单元的有序序列，如图1-1所示。为了存储和访问信息，计算机必须能以某种方式确认某个具体的内存单元。每个内存单元都有一个唯一的地址表示其在内存中的相对位置，因此可用地址实现内存单元的标识。图1-1所示为一个包含1000个内存单元的计算机内存，单元地址是0~999。

存储在内存单元中的数据称为内存单元的内容，图中内存单元3的内容是-26，而单

元4中存储的是字母H。内存单元还可存储程序指令。计算机存储程序和数据的能力称为存储程序概念：程序的指令在执行前必须调入主存储器中。

内存单元初始情况下处于随机状态，可以是数据、指令或者代码，具体由程序执行时指派。内存单元中的内容永不为空，但其初始值可能对程序毫无意义。

0	-27.2
1	354
2	0.05
3	-26
4	H
…	……
999	75.62

图1-1　内存示例

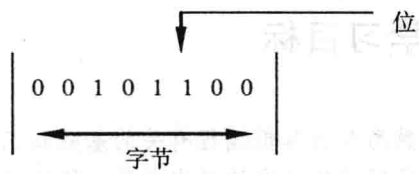


图1-2　字节与位的关系

内存单元的大小一般为一个字节（Byte），一个字节由八个位（bit）组成，如图1-2所示。一个字节是存储一个字符所需要的存储空间，例如图1-1单元4中存储的字母H。一个字包含多少个字节由计算机系统的地址结构决定，32位的系统地址结构中字长为32位，即4个字节；而64位的地址结构中字长为64位、8个字节。字节是存储空间的单位，例如：40G的硬盘、1G的内存。

内存中的数据都表示为某种模式下的0、1串。计算机对内存主要有两种操作：读（检索）和写（存储）。为存储一个值，被选定单元的每一位被重新置为0或1，此时该单元中以前存放的内容将被覆盖（overwrite）。读操作从指定单元读出其中存储的数据，这种操作对目标单元中的信息没有影响，是无损操作。无论内存中存储的是字符、数字，还是程序指令，读写过程都是相同的。

内存是易失性存储介质，系统电源关闭或计算机关机时，存储在内存中的信息将不能保留。为实现信息的永久存放，计算机还提供了其他存储方式，如硬盘（硬质磁盘）是最常用的辅助存储器。存储在磁盘上的数据以文件为单位来组织，所有文件通过磁盘的文件目录进行管理。文件目录中可以包含多个子目录，每个子目录存储与某个主题相关的文件。CPU不能直接读取辅助存储器中的数据，处理前必须将这些数据装载到内存。

1.1.2　计算机语言

开发软件需要编写供计算机执行的指令。机器语言由0、1串指令构成，具有非标准性，不同计算机系统的机器语言标准不同，因此很少用于程序编写。这种非标准性也表现在汇编语言上。汇编语言是在机器语言的基础上引入助记符形成的，使用便于记忆的代码而不是二进制数来表示计算机操作，通过变量名而不是二进制地址来访问存储单元。

高级语言用于编写与CPU类型无关且可由其执行的程序。这种语言结合了数学表达式和英语符号，接近自然语言，更容易表达问题的解决方案，但计算机识别不了，需要引入编程平台来弥补两者间理解上的鸿沟。高级语言的编写涉及以下几个概念，它们的关系如图1-3所示。

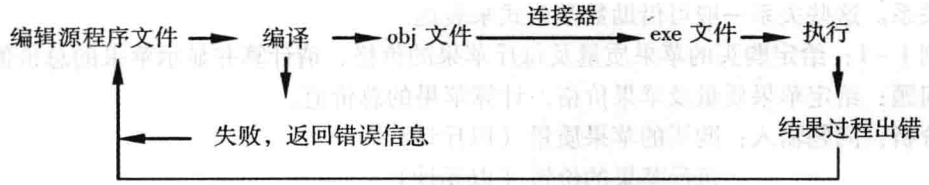


图1-3　高级语言的程序设计步骤

源程序文件：使用高级语言编写的程序代码文件，格式为文本文件（只包含标准字符集合中定义的标准字符），文件后缀为*.c。

编译器：将高级语言编写的源程序翻译成机器语言的软件或程序。

目标文件：经过编译器处理后输出的由机器语言指令构成的文件，文件后缀为*.obj。

连接器：目标文件尽管已是二进制文件，但并不完整，还不能由计算机直接执行。几乎所有程序都会用到编程平台中其他目标文件所包含的函数，连接器将这些函数与目标文件合并起来，创建一个二进制的可执行文件，文件后缀为*.exe。

程序的执行：可执行程序一般以文件形式存放在磁盘上，必须将其装入内存才能运行。这个过程由装载器完成，装载器还指导CPU从第一条指令开始执行。

为了更好地利用计算机，从第一台计算机诞生到现在，人们开发了几百种高级语言，其中仅有一小部分被广泛使用，例如Fortran、Cobol、Pascal、C、Ada等。利用人类认识世界的特性，面向对象的程序设计对现实世界的对象进行建模，提供了一种更自然直觉地看待程序设计过程的方式，例如C++、Java等。

1.2　软件开发的基本方法

编程是为了利用计算机解决问题。解决问题的方法覆盖了很多学科领域，程序员使用的是软件开发方法。

1.2.1　软件开发步骤

（1）说明程序需求。
（2）分析问题。
（3）设计问题解决的算法。
（4）实现算法。
（5）测试、检查已完成的程序。
（6）程序的维护与更新。

问题：说明程序需求，弄清楚程序的目的和需求，去除不重要的方面，找到最根本的问题所在。

分析：包括确定问题的输入（要处理的数据）、问题的输出（希望的结果）及解决方案附加的需求或约束条件。此外，还应确定结果应以怎样的格式显示，列出问题的变量及

相互关系。这些关系一般可借助数学公式来表达。

例 1-1：给定购买的苹果质量及每斤苹果的价格，请计算并显示苹果的总价值。

问题：给定苹果质量及苹果价格，计算苹果的总价值。

分析：问题输入：购买的苹果质量（以斤计）
每斤苹果的价钱（以元计）
问题输出：苹果的总价值（以元计）
问题的公式：总价 = 单价 * 数量

设计：设计算法解决问题，要求：①开发“算法”（一系列步骤）；②检查该算法是否按预期目标那样解决了问题。编写算法通常是问题解决过程中最难的部分，开始时不应试图解决问题的每个细节，而应注重于自顶向下设计方法的训练。

自顶向下的设计也叫做分治法，将问题分解成多个子问题，分别解决：首先列出需要解决的最主要步骤，即子问题，通过解决每个子问题使得初始问题最终得到解决。

大多数计算机算法都至少包含以下子问题：获取数据；执行计算（算法细化，分解成更详细的步骤）；显示结果；桌面检查。其中，桌面检查指逐步模拟算法的计算机执行，是算法设计中常被忽视的重要部分。模仿计算机认真执行算法的每一步，检查算法是否按预期工作，可以在问题解决过程的早期找出错误并修改，节省大量人力、时间。

实现：编写程序。用编程语言将算法的每个步骤转换成一个或多个语句。

测试：检查、测试程序目标是否按预期达到。这个环节需要使用不同数据集合来多次运行程序，确保程序算法在提供的所有情况下都能正常工作。

维护：一个程序的维护一般为 5 年或更长。如果想创建易于阅读、理解和维护的程序，编程过程中使用科学规范的方法是很关键的，必须遵循那些已被大家广泛接受的编程风格和原则，避免使用所谓的编程技巧或捷径。

按部就班地使用以上这些步骤来解决问题是有帮助的，但并不一定总能得到正确的解决方案。失败是编程过程的一部分，每次的失败和修正都有助于更好地理解和解决问题。

1.2.2 案例分析

问题 给定苹果的总重量及单价，求总价。

分析 问题输入：weightprice
问题输出：total_price
计　　算：total_price = price * weight

设计 初始算法

(1) 读取每斤苹果的单价及要计算的苹果的质量；
(2) 根据公式计算总价；
(3) 显示输出计算结果。

实现
```
/*程序 1-1.c，给定苹果的总重量及单价，求总价*/
#include <stdio.h>
void main( )
{   int weight, price, total_price;                    /*定义变量*/
```

```
    printf("输入苹果质量 > ");                    /*输入苹果的质量*/
    scanf("%d", &weight);
    price = 3;                                    /*指定苹果的单价*/
    total_price = price * weight;                 /*根据公式计算总价*/
    printf("苹果总价为 %d. \n", total_price);      /*显示结果*/
}
```

运行情况如下：

输入苹果质量 > 100↙

苹果总价为 300.

1.3　C语言出现的历史及特点

C语言诞生以前，系统软件主要是用汇编语言编写的。由于汇编语言依赖于计算机硬件，程序的可读性和可移植性都比较差，但一般的高级语言又难以对计算机硬件直接操作，人们希望开发一种兼具高级语言和低级语言特性的新的程序设计语言。

1.3.1　C语言的简要历史

早在1963年，英国剑桥大学基于ALGOL 60推出了CPL（Combined Programming Language）语言。ALGOL 60是一种面向问题的高级语言，与硬件的应用结合不好，不宜用来编写系统软件。CPL更接近硬件，但规模较大，难以实现。

1967年，剑桥大学的Martin Richards对CPL作了简化，推出了BCPL（Basic Combined Programming Language）语言。

1969年美国电话电报公司（AT&T）贝尔实验室的工程师K. Thompson与D. M. Ritchie用汇编语言初步完成了UNIX操作系统。为克服汇编语言的缺点，1970年K. Thompson对BCPL进一步简化，推出了B语言，并用B语言完成了UNIX。B语言过于简单，功能有限。

1972至1973年间，D. M. Ritchie在B语言的基础上加入丰富的数据类型和强有力的控制功能，设计了C语言。C语言既保持了BCPL和B语言精练、接近硬件的优点，又克服了它们过于简单的缺点，但这时的C语言只是描述和实现UNIX的一种工作语言。1973年，K. Thompson与D. M. Ritchie用C语言重写了UNIX系统90%以上的代码，扩充了UNIX系统的功能，形成UNIX第5版。C语言虽经多次改进，但主要在贝尔实验室内部使用，直到1975年UNIX第6版发布后，其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硬件的C语言版本，UNIX迅速在各种机器上得以实现。随着UNIX的广泛使用，C语言得到了推广。1978年以后，C语言先后移植到大、中、小、微型计算机上，成为应用最广泛的几种计算机语言之一。

1978年，贝尔实验室正式发表了C语言，B. W. Kernighan和D. M. Ritchie合著了影响深远的《*The C Programming Language*》，该书介绍的C语言成为后来广泛使用的C语言版本的基础，被称为标准C。

1983年美国国家标准化协会（ANSI）整理了各种版本对C语言的发展和扩充，制定了新的标准ANSI C。尽管后来又出现了新的标准，但流行的C编译系统大多仍以ANSI标准为基础开发，不同版本的C编译系统所实现的语言功能和语法规则又略有区别。

之后C语言不断得到改进和扩充，1993年引入了Simula67类概念的C语言正式被命名为C++。随着诸如运算符重载、引用、虚函数等面向对象设计思想的引入，C++功能日趋完善，并得到了ISO的认可，实现了标准化。在微型计算机上，有Microsoft C、Turbo C和Quick C等C语言版本，有MS C++、Visual C++、Turbo C++、Borland C++和C++ Builder等C++语言版本。目前较流行的Visual C++6.0既可以运行标准C程序，也可运行C++程序。本教材将以Visual C++6.0为实验平台，所有例题都在Visual C++或Turbo C 2.0环境下运行测试通过。

1.3.2 C语言的特点

C语言不仅具有高级语言的特点，还具有汇编语言的功能，同时又有极强的兼容性和良好的用户界面，主要特点如下：

（1）C语言是一种结构化、模块化的程序设计语言。C语言程序以函数为单位组成，便于模块化的程序设计，其程序结构完全由顺序结构、选择结构和循环结构构成。C语言通过9种结构控制语句来描述各种结构。

（2）C语言既具有高级语言的功能，又具有汇编语言的许多功能，可以编写系统软件，也可实现应用软件。

（3）C语言是一种数据类型丰富的高级语言，具有现代一般高级语言的各种数据结构，还具有特别的指针类型，能描述并实现许多复杂的数据结构。

（4）C语言是一种简洁的高级语言，许多I/O功能都由函数提供，因此C语言编译的实现比较简略，用C语言编写的程序可移植性好，语言简洁、紧凑，使用灵活方便，便于掌握。

（5）C语言是一种运算符丰富的语言，包含34种运算符，表达式类型多样，灵活使用各种运算符可以实现其他高级语言难以实现的运算。

（6）C语言是一种书写灵活的高级语言，一行可以写多个语句，语法限制不太严格，例如对数组下标不作检查，变量类型使用比较灵活，但这也容易导致程序错误。

（7）C语言是一种高效的高级语言，生成的目标代码质量高，程序执行效率高，用C语言程序生成的目标代码的效率能达到汇编语言目标代码效率的80%~90%。

虽然掌握C语言比其他语言难一些，但C语言良好的结构化和描述性使其特别适合教学要求；尽管C语言主要用于系统软件及硬件控制，但仍有广泛的应用领域，20世纪90年代以来C语言成为学习和使用人数最多的一种计算机语言。熟练掌握C语言成为计算机开发人员的一项基本功。

1.4　C语言的元素

C语言的一个优点是我们可以使用类似于日常英语的语言来编写程序，即使不懂得如何编写程序，也能阅读并理解基本的C语言程序。下面给出几个简单的C程序，让大家简要体会C语言的基本语言特性，具体介绍将贯穿到后继章节。

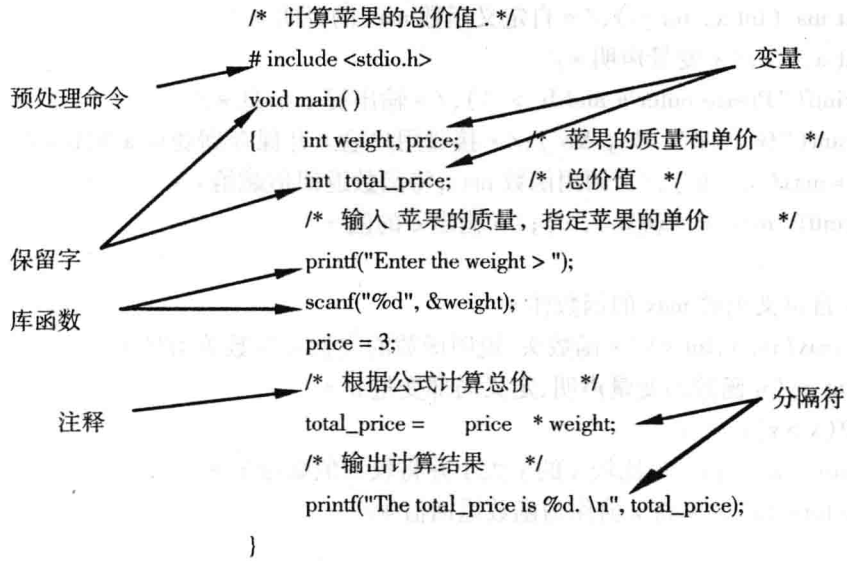


图1-4　计算苹果总价值程序中的语言元素

图1-4中的C程序包含两个部分：预处理命令和main函数。ANSI C允许源程序使用预处理命令来改进程序设计环境、提高编程效率。预处理命令不是C语言本身的组成部分，不能直接对它们进行编译，必须在程序编译前进行“预处理”。预处理命令以“#”打头。图中用到的#include是文件包含预处理命令，使系统编译源程序前将指定标准头文件的定义插入到程序中。

main函数是程序执行的起点。每个C程序都有一个main函数。图1-4中源程序的其余代码构成了函数的主体，用一对大括号{}包围起来。

注释是指以“/*”开头并以“*/”结束的部分，用来补充说明信息，提高程序的可阅读性。注释不是C程序的一部分，编译处理时将被忽略。

几乎每一行代码都包含保留字。保留字是C语言中具有特定含义的字，用于标识标准库函数、变量类型等。图中其他字可以分成两类：标准标识符和用户自定义标识符。标准标识符与保留字一样具有特殊含义，例如printf和scanf是标准I/O库中定义的输入函数和输出函数的函数名。自定义标识符，可用来标识保存数据或程序结果的变量，也可用来命名自定数函数。

程序运行时，语句printf(“Enter the weight > ”)；将显示输入提示信息Enter the weight >，输入苹果的质量100（回车表示输入结束）；语句scanf(“%d”, &weight)；接

收用户输入并将输入值100保存到变量weight；语句prince = 3；将苹果的单价price赋值为3，而计算总价值则由语句total_ price = price * weight；实现；程序最后通过printf（"The total_ price is %d. \ n", total_ price）；输出计算结果300。

例1-2：求两个数中的较大者。

```
/*程序1-2.c，求两个数中的较大者*/
1 #include <stdio.h>
2 void main( )/*主函数入口,程序执行起点*/
3 { int max(int x, int y );/*自定义函数max的说明*/
4   int a,b,c;/*变量声明*/
5   printf("Please enter a and b > ");/*输出提示信息*/
6   scanf("%d%d", &a, &b );/*接收用户输入并保存到变量a和b*/
7   c=max( a , b );/*调用函数max,将函数返回值赋给c*/
8   printf("max = %d\n", c);/*输出c的值*/
9 }
10 /*自定义函数max的函数体*/
11 int max(int x,int y)/*函数头,说明函数的类型及参数等信息*/
12 { int z;/*函数的变量声明,定义局部变量z */
13   if(x>y)z = x;
14   else  z = y; /*比较x的y大小并将较大值赋给z */
15   return (z);/*将z值作为函数返回值*/
16 }
```

运行情况如下：

```
Please enter a and b > 8  5↙
max=8
```

程序1-2.c包含两个函数：主函数main和自定义函数max。函数max的作用是比较x和y的大小并将较大值赋给变量z，返回值通过函数名带回到主调函数main的调用点处。为使自定义函数能被编译系统正确识别和调用，需要在主调函数中对max进行函数声明（第3行）。程序的第7行调用max函数，并将实际参数a和b传递给max的形式参数x和y，此时程序将转去执行12行开始的max函数体，得到一个返回值（15行）；函数max结束，程序的执行返回到调用点处（第7行"="右侧），赋值运算符"="将函数值赋给变量c。有关函数的内容详见第六章。

通过前面两个程序的分析，可以看出C程序的结构特点如下：

（1）C程序是由函数构成的，一个源程序至少包含一个main函数，还可包含若干其他函数。被调用的函数可以是系统提供的库函数（printf、scanf），也可以是用户的自定义函数（上例中的max）。函数是C程序的基本单位。

（2）函数由两部分构成：

* 函数的说明部分，包括函数类型（返回值类型）、函数名、函数参数列表，若无具体返回值，则用void标识函数类型。

* 函数体，以大括号{}为界限，包含声明部分和执行部分。声明部分定义函数用