高等学校计算机基础教育
改革与实践系列教材

# C程序设计

杨国林　主编

高等教育出版社
HIGHER EDUCATION PRESS

高等学校计算机基础教育改革与实践系列教材

# C 程序设计

C Chengxu Sheji

杨国林　主编

萨智海　陈秀燕　赵永红　秦俊平　编著




高等教育出版社·北京
HIGHER EDUCATION PRESS　BEIJING

内容提要

本书以培养学生结构化程序设计基本能力为主线，围绕相关知识点，用大量难易不等并具有代表性的实例，按照面向应用、重视实践的原则，由浅入深、循序渐进地讲解C语言程序设计的基本概念、语法，以及使用C语言进行程序设计的方法和技巧。

全书共10章，内容包括C语言概述，基本数据类型、运算符和表达式，数据的输入/输出，程序控制结构与结构化程序设计，数组，指针，函数，编译预处理，结构体、联合体及枚举类型，文件。

本书可作为高等学校计算机类专业程序设计基础课程教材，也可作为非计算机专业程序设计类公共基础课教材，还可作为参加全国计算机等级考试的考生、工程技术人员的参考书和程序设计爱好者的自学用书。

为了提高学生的实践能力，巩固各章节的知识点，作者还编写了与本书配套的《C程序设计实验指导与习题解答》，可供读者参考使用。

**图书在版编目(CIP)数据**

C程序设计/杨国林主编. --北京:高等教育出版社,2013.2

ISBN 978-7-04-036821-5

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009105号

策划编辑 李 林　　责任编辑 李 林　　封面设计 张雨薇　　版式设计 马敬茹

插图绘制 尹 莉　　责任校对 杨雪莲　　责任印制 韩 刚

出版发行 高等教育出版社
社　　址 北京市西城区德外大街4号
邮政编码 100120
印　　刷 高教社(天津)印务有限公司
开　　本 787mm × 1092mm 1/16
印　　张 21.5
字　　数 530千字
购书热线 010-58581118

咨询电话 400-810-0598
网　　址 http://www.hep.edu.cn
http://www.hep.com.cn
网上订购 http://www.landraco.com
http://www.landraco.com.cn
版　　次 2013年2月第1版
印　　次 2013年2月第1次印刷
定　　价 29.30元


物 料 号 36821-00

# 前　言

随着计算机技术的飞速发展，多种高级程序设计语言应运而生，其中C语言最具生命力。C语言是C++、Java、C#等语言的基础，这些语言完全或部分兼容了C语言的语法。因此，国内高校的计算机专业及许多非计算机专业都将C语言作为程序设计的入门课程。

近几年来，作者在主持建设“C语言程序设计精品课程”的同时，对课程的教学内容和教学方法进行了改革实践，重点对教学内容进行了优化和完善。在此基础上，结合作者多年从事C语言程序设计课程教学和利用C语言完成科研工作的经验，按照教育部高等学校计算机基础课程教学指导委员会发布的《高等学校计算机基础教学发展战略研究报告暨计算机基础课程教学基本要求》，编写了本书和与之配套的《C程序设计实验指导与习题解答》。经过多年教学实践证明，本书的内容组织循序渐进、通俗易懂，易于被学生接受。

本书以标准C语言(ANSI C)为依据，全面系统地阐述了C语言的基本概念及其程序设计方法，并对结构化程序设计技术作了较深入的讨论。全书共10章，内容包括C语言概述，基本数据类型、运算符和表达式，数据的输入/输出，程序控制结构与结构化程序设计，数组，指针，函数，编译预处理，结构体、联合体及枚举类型，文件。由于C语言涉及的概念比较复杂，规则较多，使用灵活，使初学者编程时很容易出错，所以书中还对指针概念、指针与数组的关系、函数间的数据传送以及结构体和联合体等难点内容，进行了深入的分析和解释，进一步突出了本书的实用性。

为了便于读者牢固掌握本书知识，并能尽快地把它们应用到实际开发中，书中给出了大量难易不等并具有代表性的实例，所有实例程序均在Turbo C 2.0上调试通过。配套的《C程序设计实验指导与习题解答》介绍了Visual C++ 6.0和Turbo C 2.0两个集成开发环境的使用，读者可以任选一个环境调试程序。每章末都附有一定数量的习题，可供不同层次的读者选作练习。

本书由杨国林主编。第一章、第二章、第五章、第六章和第八章由杨国林编写，第七章由萨智海编写、第九章由陈秀燕编写，第十章和附录由赵永红编写，第三章、第四章由秦俊平编写。全书最后由杨国林修改并统稿。

在本书的编写过程中，得到了校内外同行的大力支持和帮助，在此一并表示衷心的感谢。

限于编者的水平，书中难免存在错误和不当之处，敬请广大读者不吝赐教。

编　者

2012年8月

# 目　录

# 第一章
# C 语言概述

## 1.1 C语言的发展与特点

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)已成为目前世界上最广泛使用的高级程序设计语言之一。几乎在各种型号的大、中、小及微型计算机上都配有 C 语言编译系统。现在,C 语言已被人们普遍使用,它在系统软件(操作系统、语言处理、系统实用程序)、数据处理、科学工程数值计算等多个领域的软件开发中起着越来越重要的作用。

本章将简要介绍 C 语言的发展与特点、C 语言的基本程序结构、C 语言的基本语法单位、C 语言程序的编译与执行等,使读者对 C 语言有一个总体的印象,并为以后各章的学习打下良好的基础 。

### 1.1.1 C语言的发展

20 世纪 70 年代初期,为编写 UNIX 操作系统,种类繁多的计算机程序设计语言家族中又增添了一名新成员——C 语言。

1970 年,美国 AT&T 公司贝尔实验室的肯·汤普森(Ken Thompson)为实现 UNIX 操作系统而提出一种仅供自己使用的工作语言。该工作语言的前身是英国剑桥大学的马丁· 理查德(Martin Richards)在 1967 年开发的 BCPL(Basic Combined Programming Language)语言,被作者命名为 B 语言,B 取自 BCPL 的第一个字母。B 语言用于在美国 DEC 公司的 PDP-7 计算机上编写的第一个 UNIX 操作系统。此后,在美国贝尔实验室进行的更新型的小型机 PDP-11 的 UNIX 操作系统的开发中,戴尼斯·利奇(Dennis Ritchie)和布莱恩·卡尼汉(Brian Kernighan)又在 B 语言的基础上系统地引入了各种数据类型,从而使 B 语言的数据结构类型化,于 1972 年至 1973 年间推出了一种新型的程序设计语言,该语言被命名为 C 语言,C 取自 BCPL 的第二个字母。可见,C 语言名称的由来反映了该语言诞生所经历的两个过程。1973 年,肯·汤普森和戴尼斯·利奇用 C 语言重写了 UNIX 操作系统,推出了 UNIX V5,1975 年又推出了 UNIX V6。最初设计的 C 语言只是为描述和实现 UNIX 操作系统提供的一种工作语言,这时的 C 语言是附属于 UNIX 操作系统的。

在 UNIX V6 公布后,C 语言的优点逐渐引起人们的关注。为了使 UNIX 操作系统能够在所有的计算机上推广,C 语言又经过了多次改进。1977 年 C 语言的作者发表了不依赖于具体计算机系统的 C 语言编译文本《可移植 C 语言编译程序》,使 C 语言移植到其它计算机时所需做的工

作大大简化，从而推动了 UNIX 操作系统在各种计算机上的实现以及 UNIX 操作系统的发展。1978 年推出了 UNIX V7 之后，UNIX 操作系统的巨大成功和广泛使用使人们普遍注意到 C 语言的突出优点，从而又促进了 C 语言的迅速推广。C 语言和 UNIX 可以说是一对孪生兄弟，在发展过程中相辅相成。1978 年，布莱恩·卡尼汉和戴尼斯·利奇以 UNIX V7 中的 C 编译程序为基础合著了具有深远影响的著作《The C Programming Language》，被称为标准 C。这本书上介绍的 C 语言是以后各种 C 语言版本的基础。1978 年以后，C 语言先后被移植到各种大、中、小及微型计算机上。C 语言与大多数高级语言不同，它是在长期的实践中不断变化，产生了很多变种，最终才成为今天的形式。C 语言注重硬件直接支持的低级操作，从而带来了高速度。随着 UNIX 操作系统的不断传播，C 语言也普及开来，C 语言的可移植性又反过来推动了 UNIX 的传播。目前，C 语言已成为世界上使用最广泛的高级程序设计语言之一，且不依赖于 UNIX 操作系统而独立存在。

1983 年，美国国家标准化协会（American National Standards Institute，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 又公布了新标准——87 ANSI C。1988 年，戴尼斯·利奇按照 ANSI C 标准又重写了《The C Programming Language》一书。目前人们常将 1978 年的标准 C 称为旧标准，将 ANSI C 称为新标准，1990 年国际标准化组织（International Standard Organization，ISO）公布的 C 语言标准是以 87 ANSI C 为基础制定的。当前，国内最流行的 IBM PC 系列计算机上使用的 C 版本有 Turbo C、Microsoft C、Quick C 等。它们的不同版本又略有差异，因此，读者可查阅有关手册来了解所用计算机系统的 C 编译的特点和规定。

### 1.1.2 C语言的特点

C 语言之所以成为目前世界上广泛使用的程序设计语言，总是有些优于其它语言的特点。概括地说 C 语言有如下特点。

（1）C 语言兼容了其它计算机语言的一些优点，其程序结构紧凑、简洁、规整，表达式简练、灵活、实用。用 C 语言编写的程序书写格式自由，可读性强。C 语言压缩了一切不必要的成分，相对其它语言则源程序短，因此输入程序时的工作量少，编译效率高。

（2）C 语言是介于高级语言和汇编语言之间的一类语言，它比其它高级语言更接近硬件系统。它既具有像汇编语言那样直接访问硬件的功能，又具有高级语言面向用户，容易记忆和便于阅读等特点。它把高级语言的基本结构和汇编语言的高效率结合起来，其运行效率可以与汇编语言媲美。

（3）C 语言是一种结构化程序设计语言，即程序的逻辑结构可以用顺序、选择和循环 3 种基本结构组成。C 语言具有编写结构化程序所必需的基本流程控制语句（如 if ~ else、for、do ~ while、while、switch 等语句），十分便于采用自顶向下、逐步求精的结构化程序设计方法。因此，用 C 语言编制的程序具有容易理解，便于维护的优点。

（4）C 语言程序是由函数集合构成的，函数各自独立，它特别适合于大型程序的模块化设计。C 语言程序又可分割成多个源程序文件分别进行编译，然后连接起来构成一个可执行的目标文件，这一特点为开发大型软件提供了极大的方便，也为多人同时进行大型软件的集体性开发

提供了强有力的支持。

(5) C语言的运算符多达44种。除可使用常见的四则运算(+、-、*、/)及与(&&)、或(||)、非(!)等逻辑运算功能外,还可以实现以二进制位(bit)为单位的位与(&)、位或(|)、位反(~)、位异或(^)以及移位(>>、<<)等位运算,并且具有如i++、i--等单目运算符和+=、-=、*=、/=等复合运算符。丰富的数据类型与丰富的运算符相结合,使C语言具有表达灵活和效率高等特点。

(6) C语言具有丰富的数据类型,它有整型、字符型、实型等基本数据类型,还有数组类型、结构体类型、联合体类型、枚举类型及指针类型等构造数据类型,利用这些类型可实现各种复杂的数据结构。因此,C语言具有较强的数据处理能力。

(7) C语言程序可以通过#define、#include等编译预处理命令来定义"宏"和实现外部文本文件的读取和合并;还可使用#if、#else等来实现条件预编译等。总之,可通过使用预编译功能来提高软件的开发效率。

(8) C语言代码质量高(指C源程序经编译后生成的目标程序其运行速度快,占用的存储空间少),一般高级语言相对于汇编语言而言其代码质量要低得多,而C语言在代码质量上只比汇编语言低10%~20%。

(9) C语言具有较高的可移植性(可移植性是指将一个程序不作改动或稍加改动就能从一个计算机系统移到另一个计算机系统上运行)。在C语言中,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,而这些函数是系统提供的独立于C语言的程序模块。从而便于硬件结构不同的计算机间实现程序的移植。

由于C语言的上述特点,使它成为一种既可用于编写系统软件,又可用于编写应用软件的通用程序设计语言,它特别适用于编写各种与硬件环境相关的系统软件,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。因此,C语言又被称为"高级汇编语言"。

C语言的优点很多,但和其它程序设计语言一样,它也有弱点,如运算符的优先级较多,有些还与常规约定不同,不便记忆;各种C语言版本间略有差异;C语言中的变量及其值在数据类型上不要求具有严格的对应关系,类型检验很弱,转换比较随便。如char类型数据可以作为int类型数据参加运算,表达式可以作为语句使用,数组元素和结构体成员均可用指针来表示等。C语言强调灵活、高效的同时,在一定程度上也存在某些不安全因素。因此,C语言对程序设计人员提出了较高的要求,对于有经验的C程序员来说,就要能够避免不安全因素可能造成的影响。但是,C语言的优点远远超过了它的弱点,这些优点使C语言具有强大的吸引力。

## 1.2 C语言的基本程序结构

任何一种计算机程序设计语言都具有特定的语法规则和表现形式,程序的构成规则和书写格式则是程序设计语言表现形式的重要方面。熟悉C程序的构成规则和书写格式是程序设计人员编制一个易读并能被计算机正确执行的程序的前提。

学习一种新的语言,最好先读懂几个由这种语言编写的程序,弄清楚该语言的语法规则,然后自己仿写几个程序并上机运行,才能加深对其了解,这是学习计算机语言最行之有效的方法。

本节以几个简单的例子说明 C 程序的基本结构。

【例 1.1】 建立一个简单的 C 源程序(简称 C 程序)。

程序如下:

```
/* 该程序的文件名为 example1.c */
#include <stdio.h>
main()
{
 printf("The C Programming Language.\n");
}
```

这是一个简单而又完整的 C 程序。这个程序的功能是在显示屏上输出下面一行信息:

The C Programming Language.

该程序仅由一个主函数 main()构成,main 是主函数的函数名,任何一个 C 程序都必须有一个名为 main 的主函数,main 后的圆括号( )中可以有参数,也可以像该例那样没有参数,但无论有参数或无参数,括号均不能省略。函数名和参数表构成函数的头部,用花括号{ }括起来的部分称为函数体,花括号是函数体开始和结束的标志符号,程序中的花括号必须成对出现。本例中主函数的函数体仅由一个可执行语句组成。

printf()是 C 函数库中的标准输出函数,它的作用是输出 printf 后面( )中括在双引号" "中的文字(称为字符串常量或字符串),其中的"\n"是换行字符,它是由"\"和"n"两个字符构成,表示输出"\n"左边的文字之后要换行。分号";"是 C 语句的结束符,一个 C 语句必须以分号结束。加上分号的"printf();"称为输出语句。

#include 是编译程序的预编译命令,它不是 C 语句。"stdio.h"是 Standard Input&Output 的缩写,文件后缀"h"是 head 的缩写,#include 命令都是放在程序的开头,因此这类文件被称为包含文件或头文件。"stdio.h"是 C 编译程序提供的许多头文件之一,该文件中定义了 I/O 库所用到的某些宏和变量。#include <stdio.h>的作用是把头文件"stdio.h"的内容展开在#include 命令所在位置处。程序中凡是调用了标准输入与标准输出函数则均需在调用之前写上#include <stdio.h>预编译命令,并独占一行。但是,考虑到 printf()和 scanf()函数(例 1.2 中介绍)使用频繁,系统允许在使用这两个函数时可不加#include 命令。

【例 1.2】 计算三个数之和。

程序如下:

```
/* 计算三个数的和 */
#include <stdio.h>
main()
{int x,y,z;                          /* 说明 x,y,z 为整型变量 */
 float sum;                          /* 说明 sum 为实型变量 */
 printf("input x,y,z: ");
 scanf("%d%d%d",&x,&y,&z);           /* 输入 x,y,z 三个数 */
 sum=x+y+z;                          /* 计算三个数的和并赋值给 sum */
 printf("\nsum=%f\n",sum);           /* 输出 sum 的值 */
```

```
}
```

运行该程序时，首先给出提示信息"input x,y,z:"，提示用户输入3个数，然后计算这三个数的和值，并把计算出的和值以如下形式显示在屏幕上：

sum=…

在此程序中，/*……*/表示注释部分，以"/*"开头到"*/"结尾之间的内容是一个注释，它可在一行书写或分多行书写，也可写在一个语句之后。为了便于理解，本书用汉字表示注释，当然也可以用英语或汉语拼音作注释。注释的作用是帮助阅读和理解程序。编译时，注释行被忽略掉，即不产生代码行。注释在语法上仅起一个空白符的作用，它可以出现在程序中空白符能出现的任何地方，但不能嵌套出现。例如：注释/*…/*…*/…*/，这种嵌套形式非法。

程序的第四、五行是变量说明语句，x、y、z被说明为整型(int)变量，sum被说明为实型(float)变量，以便使超出整数范围的数值也能存放在该变量中。

程序的第六行是一个输出语句，用于输出一行提示信息"input x,y,z:"。第七行的"scanf();"是输入语句，scanf()是一个由系统提供的标准格式化输入函数，其后的圆括号内为参数表，"%d%d%d"为格式串，%d表示十进制整数格式。该输入语句的作用是把来自标准输入设备的三个整数均按十进制整数格式分别存入变量x、y、z所对应的存储单元(以后简称存入变量x、y、z)，&x、&y、&z分别表示变量x、y、z所对应的存储单元的地址(以后简称&a为变量a的地址)，执行该语句时，操作员要从键盘输入3个整数，数之间以空格隔开。

程序第八行是赋值语句，等号=是赋值运算符，表示把赋值号右边的表达式x+y+z的运算结果赋给sum。

程序的第九行为输出语句，它首先在新的一行上输出字符串"sum=",然后按实数格式(%f)输出变量sum的值，并使光标移到下一行。

【例1.3】 求三个数中的最大值。

程序如下：

```
/*找出三个数中的最大值*/
main()                                  /*主函数*/
{int a,b,c,maxi;                        /*变量说明*/
 printf("please to input a,b,c:");
 scanf("%d%d%d",&a,&b,&c);              /*输入变量a、b、c的值*/
 maxi=max(a,b,c);                       /*调用max函数,将得到的最大值赋给maxi*/
 printf("\nmaximum is %d",maxi);        /*输出最大值maxi*/
}
int max(int x,int y,int z)              /*在三个数中找最大值的函数*/
{int m;
 if (x>y)
   m=x;
 else
   m=y;
 if (m<z)
```

```
    m=z;
  return(m);                          /*将最大值 m 通过 max 函数返回调用处*/
}
```

该程序由两个函数组成,主函数和一个名为 max 的用户自定义函数。用户自定义函数中的"int max(int x,int y,int z)"说明函数返回值的类型为整数类型,函数名为 max,形式参数为 x、y、z,它们均为整数类型。max 函数的功能是找出 x、y、z 中较大的一个值并存入变量 m,"return(m);"将结果返回给主函数 main。

C 程序的执行总是从 main 函数开始的,main 函数中的所有语句执行完毕,则程序执行结束。main 函数中 scanf 语句的作用是将三个整数值分别输入到变量 a、b、c 中,当执行到"maxi=max(a,b,c);"语句时,即调用 max 函数,在调用时将实际参数 a、b、c 的值分别传递给 max 函数中的形式参数 x、y、z 后,控制被传递给 max 函数,经过执行 max 函数得到了一个返回值 m,当执行"return(m);"语句后,结束 max 函数的执行,控制又被传递给 main 函数,并将这个值赋给了变量 maxi。然后通过 printf 语句输出 maxi 的值。

程序运行情况如下:

```
please to input a,b,c:4  6  2↵        (给变量 a、b、c 分别输入 4、6、2)
maximum is 6                          (输出 maxi 的值)
```

该例的程序中只包含了两个函数,实际上,一个 C 程序可以包含多个像 max()这样的被调用函数。

上面三个引例,源程序较短,各函数都放在同一个文件中。如果程序很长,将会增加程序的编译时间。C 语言允许把一个程序分成几块,放在不同的文件中分别进行编译。分别编译的优点是,当只修改了一个文件中的代码时,不必重新编译整个程序,而只要编译做过修改的那个文件即可。另外,在编写程序的过程中,人们可能积累了一些有用的函数,此时可以把它们分别放在不同的文件中,以便在其它程序中随时调用。

下面是一个分别编译的例子。

【例 1.4】 例 1.3 源程序被分别编辑在两个文件中。

文件名为 file1.c 的第一个文件内容如下:

```
/*找出三个数中的最大值*/
main()                                /*主函数*/
{int a,b,c,maxi;                      /*变量说明*/
 printf("please to input a,b,c:");
 scanf("%d%d%d",&a,&b,&c);            /*输入变量 a、b、c 的值*/
 maxi=max(a,b,c);                     /*调用 max 函数,将得到的最大值赋给 maxi*/
 printf("\nmaximum is %d",maxi);      /*输出最大值 maxi*/
}
```

文件名为 file2.c 的第二个文件内容如下:

```
int max(int x,int y,int z)            /*在三个数中找最大值的函数*/
{int m;
 if (x>y)
```

```
    m=x;
  else
    m=y;
  if (m<z)
    m=z;
  return(m);                          /*将最大值 m 通过 max 函数返回调用处*/
}
```

这实际上是把例 1.3C 程序中的两个函数分别放在了 file1.c 和 file2.c 两个源文件中,每个源文件是一个编译单位,编译时应分别进行。当组成一个 C 程序的所有源文件都被编译成二进制代码形式的目标文件之后,由连接程序将各目标文件中的目标函数和系统标准函数库中的函数装配成一个可执行的 C 程序,运行连接生成的可执行程序其结果同例 1.3。

下面是在 Turbo C 环境下进行多文件编译、连接的操作过程。

(1) 启动 Turbo C。

(2) 利用编辑程序 Edit 建立一个项目文件,该文件的内容是:

file1.c

file2.c

即项目文件中只含有 C 源程序的两个文件名,在项目文件中还可指定这些 C 源程序文件的路径。

将该项目文件存盘时的文件扩展名命名为".prj",该例将项目文件命名为 filec.prj。

(3) 选择主菜单项 Project 下拉菜单中的 Project Name 项,按 Enter 键后,输入项目文件名,如:

filec.prj ↵

该项目文件是将 C 程序的多个源文件和目标文件连接成一个完整的程序。

(4) 按功能键<F9>进行编译、连接后可得到一个可执行文件,如 filec.exe,该文件是将项目文件中的各个 C 源程序文件依次编译成目标文件,并将它们连接起来后生成的。

(5) 选择运行,则可执行所生成的可执行文件 filec.exe。

看了以上几个程序实例,现将 C 程序基本结构说明如下。

1. C 程序的组成

一个 C 程序可由一个函数或多个函数组成,其中必须有且只能有一个以 main 命名的主函数。其余函数的名字由程序设计者自定。每个函数完成一定的功能,函数与函数之间可以通过参数传递信息。各函数只能并列定义,相互不能嵌套。主函数 main 可以位于源程序文件的任何位置,但程序的执行总是从 main 函数的第一个可执行语句开始,随 main 函数的执行结束而结束。其余函数都是在 main 函数开始执行以后,通过函数调用或嵌套调用而得以执行的。因此,主函数实际上是整个程序的控制部分。其余的函数可以是用户自定义的函数,也可以是系统提供的库函数。C 语言的库函数十分丰富,ANSI C 提供 100 多个库函数,Turbo C 和 Microsoft C 提供 300 多个库函数。可以说 C 语言是函数式语言。

2. 函数的组成

一个函数由两部分组成:函数首部和函数体。函数首部包括函数的类型、函数名和形式参数

表。如例1.3中max函数的函数首部为：

```
int max(int x,int y,int z)
```

形式参数表可以为空，这样的函数为无参函数。函数体是花括号括起的部分，它包括局部变量说明部分和语句部分。可以没有局部变量说明部分，也可以没有语句部分，如

```
void dummy( )
{       }
```

是一个不执行任何操作的空函数。综上所述，一般函数的结构如下：

```
[数据类型标识符] <函数名>(形式参数说明表)
{
  [局部变量说明部分]
  [语句部分]
}
```

3. C库函数

C函数分为两类：标准函数(即库函数)和用户自定义函数。标准函数是由C编译程序提供的，标准函数的定义以编译后的目标代码形式存放在系统的标准函数库中。用户在程序中需要使用标准函数(如scanf、printf函数)完成某个功能时，只需使用#include预编译命令指出所使用标准函数的系统头文件即可。

用户自定义函数(如例1.3中的max函数)是用户根据自己的需要而编制的函数，有关函数的详细内容将在第七章介绍。

4. 外部说明

在函数定义之外还可以包含一个外部说明部分，它可包含#include预编译命令、外部量的说明等。

C程序书写格式说明如下：

(1) C程序习惯使用小写英文字母书写，大写英文字母一般用作符号常量名和其它特殊用途。

(2) C程序每个语句可从任意列开始，不存在程序行的概念。一行中可以有多个语句，一个语句也可以占用多行。但是，为了使程序结构层次清楚，应当以锯齿形格式书写程序，即在语句之前加上适当数量的空格字符，使处于同一结构层次的语句从同一列开始，从而形成层层缩进对齐的书写格式。

(3) 标识符的命名应当“见其名知其意”。

(4) 为了增强C程序的可读性，可以使用适量的空格和空行，但变量名、函数名以及C语言的关键字(如for、switch等)中间不能加入空格。除此之外的空格和空行可以任意设置，C语言编译系统不理会这些空格和空行。

(5) 为了帮助读者了解函数、程序块或某几个语句的功能，可以适当地使用注释。

下面是按上述书写格式编写的计算给定的10个数中奇数与偶数之和的程序实例。