C语言程序设计

主　编　高　禹
副主编　王　丹　苏荣聪　林玉梅
参　编　张梅娇　胡小琴　郭新华
　　黄丽凤　宋　伟　芦　欣
　　张　琼　杨雨薇

北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

前　言

C 语言是一种优秀的计算机语言。C 语言具有语言简洁、功能丰富、灵活性强、可移植性好等特点。C 语言深受广大用户的喜爱，为人们广泛使用。C 语言具有较强的实用性，它既可以用于编写系统软件，也可以用于编写各种应用软件。

C 语言程序设计既是计算机类专业的必修课程，也是许多高校为非计算机类专业学生开设的一门程序设计语言课程。对于从未接触过程序设计语言的学生来说，在有限的学时内掌握好 C 语言具有一定难度。笔者根据多年从事 C 语言教学的经验，在编写本书时充分地考虑了这些实际情况。

本书的编写具有以下主要特点：

（1）在内容的编排上，充分考虑高等院校培养应用型本科专业人才的要求。

（2）尊重学生的学习规律，按照由浅入深、循序渐进的原则安排各章的知识点。

（3）从初学者的角度出发，重点考虑如何使用 C 语言编程来解决实际问题，选择读者容易理解的问题作为实例，并结合知识点来讲解程序设计的方法和技巧。

（4）例题类型丰富，包含了多种常见类型，且对于例题中出现的各种算法都有较详细的分析。

（5）每章都设计了上机实验项目，并详细说明了各实验的目的和内容。

（6）每章都提供了相关习题，并提供了习题参考答案。

全书共分 10 章：第 1 章，介绍 C 语言的发展历史、特点及源程序结构等知识；第 2 章，介绍 C 语言程序设计的基本知识，如数据类型、运算符和表达式、基本的输入与输出操作和顺序结构程序设计等；第 3 章，介绍 C 语言的选择结构程序设计知识；第 4 章，介绍 C 语言的循环结构程序设计知识；第 5 章，介绍 C 语言使用数组的知识；第 6 章，介绍 C 语言使用函数的知识及变量的属性；第 7 章，介绍 C 语言的编译预处理知识；第 8 章，介绍 C 语言指针的使用知识；第 9 章，介绍 C 语言的结构体、共用体和枚举类型知识；第 10 章，介绍 C 语言使用文件的知识。全书覆盖了计算机等级考试（二级 C）的全部内容。

本书条理清晰、语言流畅、通俗易懂、实用性强，既可以作为高等院校应用型本科专业学生的教材，也可供自学者以及参加 C 语言计算机等级考试者阅读参考。

本书由泉州信息工程学院的高禹担任主编，由黄河交通学院的王丹、泉州信息工程学院的苏荣聪和林玉梅担任副主编，参加本书编写工作的还有泉州信息工程学院的张梅娇、胡小琴、郭新华、黄丽凤，以及南通理工学院的宋伟、芦欣、张琼、杨雨薇等，在此向所有关心及帮助此书编写的人士致谢！

由于编者水平有限，书中难免存在不足之处，恳请读者批评指正。

编　者

CONTENTS 目录

第1章

C语言概述

C语言是一种应用普遍的程序设计语言。C语言数据类型丰富、语句简洁紧凑、灵活性强、具有结构化的控制语句、编程功能强大，深受广大编程人员喜爱。

本章主要介绍C语言的发展简史和特点，举例说明C语言源程序的结构特点，以及C程序的编辑、编译、连接和运行的过程。

1.1 C语言简介

C语言由早期的B语言发展演变而来。1970年，贝尔实验室的Ken Thompson根据BCPL(Basic Combined Programming Language)设计出了较简单且接近硬件的B语言，但B语言过于简单，功能有限，无法满足人们的需要。1972年，Dennis Ritchie在此基础上开发出C语言，并首次在UNIX操作系统的DEC PDP-11计算机上使用。C语言继承了B语言的优点，且克服了它的缺点。

C语言最初只能在大型计算机上执行，随着UNIX操作系统的日益普及，它被移植到微机上，并且出现了许多不同版本的C语言。由于没有统一的标准，这些C语言之间出现了一些不一致的地方。1983年，美国国家标准协会ANSI(American National Standards Institute)为C语言制定了标准，即ANSI C。1987年，ANSI公布了C语言的新标准；1989年，ANSI又公布了一个新的C语言标准，即C 89，现在流行的各种C语言版本都以C 89为标准。

1990年，国际标准化组织(International Standards Organization，ISO)接受C 89作为国际标准，后通常称之为C 90。1999年，ISO对C语言标准进行修订，在基本保留原来的C语言特征的基础上增加了一些面向对象的特征，简称C 99。

微机上常用的C语言编译系统有Visual C++、C-Free、Turbo C、WIN TC(Turbo C的Windows版本)等。

C 语言编程功能强大，主要优点如下：

（1）与其他高级语言相比较，C 语言简洁、紧凑、灵活，使用方便。

（2）C 语言具有丰富的运算符和数据类型，使用这些运算符和数据类型可以实现各种复杂的运算。

（3）C 语言可以直接访问物理地址，能进行位操作，能实现汇编语言的大部分功能，可直接对硬件进行操作，兼有高级语言和汇编语言的特点。

（4）C 语言具有结构化的控制语句（如 if…else 语句、while 语句、do…while 语句、switch 语句、for 语句），以函数作为程序的基本模块，是结构化的理想语言。

（5）C 语言对语法的限制不太严格，程序设计自由度大。例如，对数组下标越界不进行检查，对变量的类型使用比较灵活，整型数据与字符型数据以及逻辑型数据可以通用。编写程序时，应当仔细检查，防止程序出错，但不要过分依赖 C 编译程序（它可以检查错误，但有的错误检查不出来，如程序的逻辑错误）。

（6）用 C 语言编写的程序可移植性好（与汇编语言相比）。在某一操作系统下编写的程序，基本上无须作任何修改就可以在其他类型的计算机和操作系统上运行。

由于具有以上优点，因此 C 语言应用广泛。

与学习其他高级语言相比，C 语言对编程人员的要求比较高，编程人员在学习 C 语言的语法上必须耗费更多精力，尤其是在指针的应用方面。但是，待熟悉 C 语言的语法之后，便可以感受到 C 语言编程功能的强大和使用的方便。

1.2 简单的 C 程序

下面通过几个简单的 C 程序，进一步了解 C 程序的结构特点。

例 1.1 在屏幕上显示“Welcome!”和“Let's learn about C programming.”两行信息。

程序代码如下：

```
#include <stdio.h>
int main()
{ printf("Welcome!\n");
   printf("Let's learn about C programming.");
  return 0;
}
```

程序运行后，输出以下两行信息：

```
Welcome!
Let's learn about C programming.
```

上面程序中的“#include <stdio. h >”表示把尖括号 <> 内的 stdio. h 文件包含到本程序中。stdio 为 standard input/ output（标准输入/输出）的缩写。C 语言中有关输入/输出函数的格式均定义在 stdio. h 文件里。

C 程序是由许多函数组合而成的，在上面的程序中只包含一个“主函数”，main 是主函数名（每一个 C 程序都必须有且只有一个 main 函数），主函数 main 是 C 程序执行的入口。main 前面的 int 表示函数的返回类型，即 main 函数的类型为 int 型。

在例 1.1 的程序中，放在一对大括号 {} 里面的部分称为函数体。函数体内的 printf 是 C 语言中的输出函数，双引号内的字符串是被输出的信息。“\n”是换行符，表示在输出“Welcome!”后回车换行，然后输出“Let's learn about C programming.”。

每个语句用一个分号结尾。函数体内的“return”语句为主函数结束时的返回值。由于 main 函数的类型为 int，因此返回值必须为一个整型值。一般而言，返回值为 0 表示正常返回。

例 1.2 计算两个 int 型变量之和，并在屏幕上显示计算结果。

程序代码如下：

```
#include <stdio.h>
int main()                              /* main 是主函数 */
{ int n,m,sum;                          /* 定义 int 型变量 n,m,sum */
  n=23;  m=345;                         /* 为变量赋值 */
  sum=n+m;                              /* 求两个变量之和 */
  printf("两个变量的和为 %d",sum);      /* 输出两个变量的和 */
  return 0;
}
```

程序运行后，输出结果如下：

```
两个变量的和为 368
```

在该程序中，使用/* … */表示注释。注释只是用于解释程序，对编译和运行不起任何作用。

在该程序的函数体内（即一对大括号之间）：第 1 行，定义变量，定义了 3 个 int 型变量；第 2 行，两个赋值语句，将 23 赋值给变量 n，将 345 赋值给变量 m；第 3 行，将 n 和 m 之和赋值给 sum；第 4 行，printf 是输出函数，其中的%d 表示按照“十进制整数类型”的格式输出 sum 的值，执行输出时，系统将在%d 位置上以一个十进制整数值代替%d；函数 printf 中括弧内最右端的 sum 是要输出的变量，它的值是 368，因此输出“两个变量的和为 368”。

例 1.3 调用自定义函数计算变量 a 与 b 的和。要求在主函数中输入变量 a 与 b 的值，并将计算结果输出。

程序代码如下：

```
#include <stdio.h>
int sumtwo(int x,int y);                /* 自定义函数声明 */
int main()                              /* 主函数 */
{ int a,b,sum;                          /* 定义变量 */
  printf("请输入变量 a 与 b 的值:");    /* 显示提示信息 */
```

```
    scanf("%d%d",&a,&b);                    /* 输入变量 a 和 b 的值 */
    sum = sumtwo(a,b);                      /* 调用 sumtwo 函数 */
    printf("a 与 b 的和等于%d",sum);        /* 输出计算结果,即 sum 的值 */
    return 0;
}
int sumtwo(int m,int n)                     /* 自定义函数 sumtwo,包含参数 m、n */
{ int k;
   k = m + n;
   return k;
}
```

上面的程序由两个函数（即主函数 main 和自定义函数 sumtwo）组成。

函数 sumtwo 是一个用户自定义函数，它的功能是求两个整数之和并返回给主函数。它有两个 int 型的形参 m 和 n，函数 sumtwo 的返回值是 int 型的。

main 函数前面的函数声明语句“int sumtwo(int m,int n);”表明，sumtwo 是一个有两个 int 型的形参并返回一个 int 型值的函数。这样的函数声明叫作函数原型，它应与函数的定义和调用一致。

该程序的执行过程：首先，在屏幕上显示提示字符串“请输入变量 a 与 b 的值:”，等待用户输入两个数。用户输入两个数（要用空格间隔）并按回车键后，由 scanf 函数语句接收这两个数并存入变量 a、b。然后，调用 sumtwo 函数，把 a 和 b 的值传递给 sumtwo 函数的参数 m 和 n。在 sumtwo 函数中，计算 m 与 n 之和，并赋给变量 k，由 return 语句把变量 k 的值返回给主函数 main，并赋值给变量 sum。最后，由 printf 函数输出 sum 的值。

从以上例题可以看出，C 语言源程序有以下结构特点：

（1）一个 C 语言源程序由一个或多个源文件组成，每个源文件由一个或多个函数构成，其中有且仅有一个主函数（main 函数）。

（2）一个函数由函数首部（即函数的第一行）和函数体（即函数首部下面的大括号内的各行代码）组成。

（3）函数首部包括函数类型、函数名和放在圆括号内的若干参数。函数体由声明部分和执行部分组成。

（4）C 源程序的每个语句以分号结尾。一行内可以写多条语句，一条语句也可以分写在多行中。

（5）放在“/*”与“*/”之间的是注释内容，注释部分允许出现在程序中的任何位置。

1.3 C 程序的编辑、编译、连接和运行

1. 编辑程序

用编辑软件将 C 源程序输入计算机，经修改无误后，保存为一个文件，C 源程序文件的

扩展名为“.c”。可用于编写C源程序的编辑软件有很多，在Windows环境下，可以使用Visual C++、C-Free、WIN TC。

2. 编译程序

使用Visual C++（或C-Free、WIN TC）软件，将扩展名为“.c”的源程序编辑保存之后，通过快捷键或者选择菜单的方式进行编译。编译的过程是把C源程序代码转换为计算机可识别的代码。如果在编译过程中发现源程序有语法错误，系统会显示出错信息，然后用户重新修改源程序，再进行编译，如此反复直至编译通过。编译通过后，生成目标程序，目标程序的文件名与源程序相同，其扩展名为“.obj”。

3. 连接程序

将目标程序和库函数或其他目标程序连接，即可生成可执行程序。在Visual C++（或C-Free、WIN TC）中，可通过快捷键或选择菜单的方式进行连接。可执行程序的文件名与源程序相同，其扩展名为“.exe”。

4. 运行程序

输入可执行文件的文件名即可运行程序。在Visual C++（或C-Free、WIN TC）中通过快捷方式或选择菜单的方式即可运行程序。

编辑、编译、连接、运行程序的过程如图1.1所示。

图1.1 C程序的执行过程示意图

1.4 习 题

1. 说明C程序具有哪些结构特点。

2. 分析构成例1.3的源程序中的每个函数的结构，指出每个函数体的声明部分和执行部分各包括哪些内容。

3. 分别编写完成以下任务的程序，然后上机编辑、编译、连接、运行。

（1）输出两行字符，分别是“We learn C language.”和“We use the Internet.”。

（2）从键盘输入int型变量x、y的值，分别计算x+y、x-y的值，将计算结果分别存放在int型变量s1、s2中，并输出s1、s2的值。

扫描二维码获取习题参考答案

第2章

C程序设计基础

使用C语言编写程序时，需要一些基础知识。例如：常量、变量；数据类型（整型、实型、字符型等）；几种运算符（算术运算符、赋值运算符、强制类型转换运算符、自增/自减运算符、逗号运算符、求字节数运算符、位运算符等）；几种表达式（算术表达式、赋值表达式、逗号表达式等）；顺序结构设计方法；输入/输出函数等内容。熟悉了这些基本知识之后，才能编写程序。本章将介绍这些基础知识。

2.1 C语言的数据类型

C语言提供了如图2.1所示的数据类型。

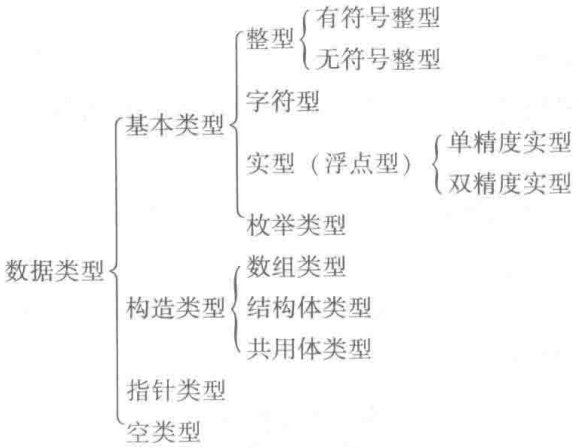

图2.1 C语言的数据类型

2.1.1 常量和变量

1. 常量

常量是在程序运行过程中其值不能被改变的量。常量分为以下几种：

（1）整型常量，如 -247、0、368。

（2）实型常量，如 -3.14159、0.618、2.71828。

（3）字符常量，如 'a'、'b'、'A'、'B'、'#'、'*'、'3'、'6'。

（4）符号常量，例如：

```
#define PI 3.14159
```

这里，用#define 指令指定用符号 PI 代表 3.14159。程序中出现的 PI 将用 3.14159 代替。

2. 变量

变量是在程序运行过程中其值可以被改变的量。

任何一个变量在使用前，必须对其进行定义。定义就是为该变量命名并声明其数据类型。定义后，编译系统将为该变量在内存中分配存储单元，在该存储单元中存放该变量的值。

用于标识变量名（或符号常量名、函数名、数组名、类型名、文件名）的有效字符序列称为标识符。C 语言规定，标识符只能由 3 种字符组成——英文字母、数字、下划线，并且第一个字符必须是字母或下划线。

编译系统认为，大写英文字母和小写英文字母是不同的字符。例如，book 和 Book 是两个不同的变量名。为变量命名时，一般用小写英文字母。

定义变量的一般格式如下：

```
[存储类型] 数据类型 变量名1[,变量名2,…];
```

例如：

```
int n1,n2,n3,sum;
```

在定义变量的同时，可以对变量赋初值，这种操作称为变量初始化。变量初始化的一般格式如下：

```
[存储类型] 数据类型 变量名1[=初值1][,变量名2[=初值2]…];
```

例如：

```
int math=85,phys=86,chemi=87;
```

在这些定义格式中，放在中括号内的内容可以省略，本书后面文中都是如此。

关于定义格式中的“存储类型”，将在后面章节中介绍。

2.1.2 整型数据

1. 整型常量

在 C 语言中，整型常量可以用以下 3 种形式表示。

（1）十进制形式，如 2345、0、-3169。

（2）八进制形式（以数字 0 开头），如 0135，即 $(135)_8$，对应于十进制的 93。

（3）十六进制形式（以数字 0 和小写字母 x 开头），如 0x23，即 $(23)_{16}$，对应于十进制的 35。

2. 整型变量

整型变量可分为有符号整型变量和无符号整型变量两大类，根据变量的取值范围，每类

可分为基本整型、短整型、长整型。因此，共有以下6种整型变量：

```
有符号基本整型        [signed] int
有符号短整型          [signed] short [int]
有符号长整型          [signed] long [int]
无符号基本整型        unsigned [int]
无符号短整型          unsigned short [int]
无符号长整型          unsigned long [int]
```

使用时，方括号内的部分可以省略，如“unsigned [int]”与“unsigned”等价。

例如，下面两种定义方式相同，分别定义了有符号基本整型变量m和n。

```
signed int m,n;
int m,n;
```

数据在内存中是以二进制形式存放的。若不指定是无符号型unsigned，或者指定是有符号型signed，则存储单元的最高位是符号位（用0代表正数，用1代表负数）。若指定是无符号型unsigned，则存储单元的全部二进制位（bit）都用于存放数本身，而不包括符号。

整型数以二进制补码形式存放于内存中。

二进制正数的原码、反码和补码都相同。例如，若定义“short n=6;”，则n的原码、反码和补码都是00000000 00000110。

二进制负数的原码：符号位是1，数值部分用二进制的绝对值表示。它的反码：将其原码除符号位外，其余各位按位取反，即将1都换成0、将0都换成1。它的补码：在其反码的最低位加1。例如，若定义“short n=-13;”，那么n的原码是10000000 00001101，n的反码是11111111 11110010，n的补码是11111111 11110011。

关于以上各类数据所占内存的大小，C标准要求long型数据不短于int型、short型不长于int型即可，具体怎样实现，由计算机系统自行决定。例如，在使用Visual C++或C-Free软件时，short型占2字节，int型和long型各占4字节。

对于有符号整型变量，2字节的取值范围为$-2^{15}\sim2^{15}-1$，即-32768~32767；4字节的取值范围为$-2^{31}\sim2^{31}-1$，即-2147483648~2147483647。

对于无符号整型变量，2字节的取值范围为$0\sim2^{16}-1$，即0~65535；4字节的取值范围为$0\sim2^{32}-1$，即0~4294967295。

根据上面规定的取值范围，在为整型变量赋值时，应注意不要超出变量的取值范围，否则会发生“溢出”，而出现“溢出”时程序并不报错。因此，编写程序时要根据实际情况，准确选择变量的类型，避免“溢出”。

3. 整型数据的输入输出

函数scanf可以实现输入。函数scanf的功能是按照指定格式、将从标准输入设备输入的内容送入变量。

函数printf可以实现输出。函数printf的功能是按照指定格式、将数据显示在标准输出设备上。

这里的“指定格式”需要使用格式说明符%和格式字符。用于输入/输出整型数据的格式字符有英文字母d、o、x、u等。具体含义如下：

%d——表示输入/输出十进制整型数据。

%o——表示输入/输出八进制整型数据。

%x——表示输入/输出十六进制整型数据。

%u——表示输入/输出无符号整型数据。

除了%d格式之外，上面的其他几种格式都将数据作为无符号数据进行输入/输出。也就是说，若要输入/输出带符号的整数（正整数或负整数），就必须使用%d格式。

如果输入/输出的是长整型数，就一定要在%的后面加上字符l（字符L的小写），否则可能显示不正确。例如，使用%ld输入/输出十进制长整型。

例2.1 举例说明整型数据的输出。

程序代码如下：

```
#include <stdio.h>
int main()
{ int n1 =80,n2 =20,s;
   s =n1 +n2 +68;
   printf("%d,%d\n",s,n1 +n2);
   printf("%o,%o\n",s,n1 +n2);
   printf("%x,%x\n",s,n1 +n2);
   return 0;
}
```

程序运行结果如下：

```
168,100
250,144
a8,64
```

例2.2 举例说明整型数据的输入。

程序代码如下：

```
#include <stdio.h>
int main()
{ int a,b,c;  unsigned d;long e;
   scanf("%d,%o,%x",&a,&b,&c);
   printf("%d,%d,%d\n",a,b,c);
   scanf("%u,%ld",&d,&e);
   printf("%u,%ld\n",d,e);
   return 0;
}
```

运行程序，若输入：100,100,100↙（回车符）

则输出：100,64,256

若输入：54321,765432↙

则输出：54321,765432