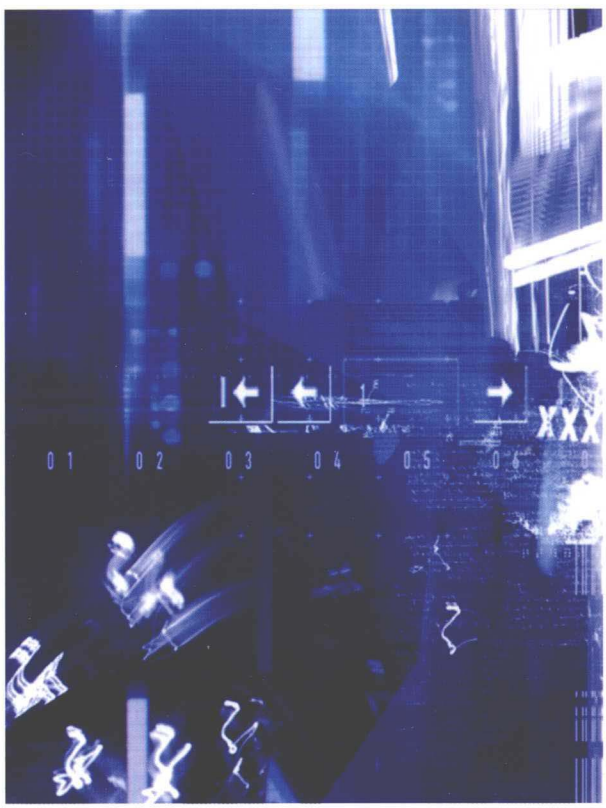
高等学校计算机应用规划教材

C语言程序设计

- C语言基础知识
- 顺序结构程序的设计
- 选择结构程序的设计
- 循环结构程序的设计
- 数组的使用
- 函数的使用
- 预处理命令
- 指针的使用
- 结构体、链表与共用体
- 位运算
- 文件的概念和操作



高禹　主编

马相忠　主审

顾沈明　刘军　亓常松　副主编

清华大学出版社

内 容 简 介

C 语言是国内许多高校为学生开设的第一门程序设计语言课程。C 语言具有很强的实用性，它既可用来编写系统软件，也可用来编写各种应用软件。

本书主要内容包括：C 语言概述，数据类型、运算符与表达式，程序设计初步，选择结构程序的设计，循环结构程序的设计，数组，函数，预处理命令，指针，结构体与其他数据类型，位运算，文件等。书中安排了大量程序设计实例，通过实例使读者能够更好地掌握运用 C 语言进行程序设计的方法和技巧。

本书既可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。

为了使读者更好地掌握 C 语言，清华大学出版社还出版了与本教材配套的学习指导与实验辅导教材：《C 语言程序设计学习指导与实验教程》，书号：978-7-302-24344-1。

本书对应的电子教案和实例源文件可以到 http://www.tupwk.com.cn/downpage 网站下载。

图书在版编目(CIP)数据

C 语言程序设计/高禹　主编. —北京：清华大学出版社，2011.1

(高等学校计算机应用规划教材)

ISBN 978-7-302-24345-8

I. ①C… II. ①高… 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 248687 号

责任编辑：胡辰浩(huchenhao@263.net)　袁建华

装帧设计：孔祥丰

责任校对：成凤进

责任印制：王秀菊

出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座

http://www.tup.com.cn　　**邮　　编**：100084

社　总　机：010-62770175　　**邮　　购**：010-62786544

投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn

质　量　反　馈：010-62772015，zhiliang@tup.tsinghua.edu.cn

印 刷 者：北京密云胶印厂

装 订 者：北京鑫海金澳胶印有限公司

经　　销：全国新华书店

开　　本：185×260　**印　张**：15　**字　数**：346 千字

版　　次：2011 年 1 月第 1 版　　**印　　次**：2011 年 1 月第 1 次印刷

印　　数：1～5000

定　　价：26.00 元

产品编号：040519-01

前　言

C 语言是广泛使用的一种计算机语言，由于它功能丰富，灵活性强，可移植性好，语言简洁，应用面广，因此深受广大用户的喜爱。C 语言具有较强的实用性，它既可以用来编写系统软件，也可以用来编写各种应用软件。

C 语言程序设计既是计算机专业的必修课程，也是国内许多高校为非计算机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，在规定的有限学时内，掌握好 C 语言具有一定的难度。作者在编写本书时，根据多年从事 C 语言教学的经验，充分地考虑到了以上实际情况。

本书的编写具有如下主要特点。

1. 本书在编写过程中，充分考虑到了高等院校培养应用型本科专业学生的要求，在内容的编排上充分考虑了初学者的要求。

2. 本书内容的组织遵循深入浅出、通俗易懂的原则，首先采用精练的语言介绍每章的知识点，然后选择学生容易理解的问题作为实例，结合该章知识讲解程序设计的方法和技巧。

3. 本书编写本着实用的原则，重点放在如何使用 C 语言来解决实际问题，在丰富的例题中包含了各种常见问题，对于例题中出现的解决每个问题的算法都有较详细的解释。

4. 与本书相配套，我们还编写了《C 语言程序设计学习指导和实验教程》，对各章知识的要点和难点进行了整理归纳和深入分析，为读者准备了各种类型的习题，并且给出了习题的参考答案，为读者设计了各种上机实验项目并详细说明了每个实验的目的和内容。

5. 本书内容覆盖了“C 语言计算机等级考试”的内容。

全书共分 12 章：第 1 章介绍了 C 语言的发展历史、特点及源程序结构；第 2 章介绍了 C 语言的基本数据类型、运算符和表达式；第 3 章介绍了 C 语言基本的输入输出操作和顺序结构程序设计；第 4 章介绍了 C 语言的选择结构程序设计；第 5 章介绍了 C 语言的循环结构程序设计；第 6 章介绍了 C 语言的数组；第 7 章介绍了 C 语言函数的使用、变量的存储类别；第 8 章介绍了 C 语言的预处理命令；第 9 章介绍了 C 语言的指针的使用；第 10 章介绍了 C 语言的结构体和其他数据类型；第 11 章介绍了 C 语言的位运算；第 12 章介绍了 C 语言的文件的概念及操作。

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

除主编和副主编外，参加本书编写工作的还有章毓凤、谭小球、张建科、陈荣品、徐妙君、江有福、李鑫、朱顺乐、王广伟、管林挺、乐天等。

由于编者水平有限，书中难免存在错误与不足，诚恳欢迎读者批评指正。我们的联系方式为信箱：huchenhao@263.net，电话：010-62796045。

编　者

目　录

第1章　C语言概述

C 语言是一门非常优秀的结构化程序设计语言，它具有简洁、紧凑、灵活和可移植性强等优点，深受广大编程人员的喜爱，并得到广泛的应用。

本章主要介绍了 C 语言的发展历史、C 语言的特点以及 C 语言是如何编译、连接和运行的。

1.1　C 语言的发展历史简介

C 语言是美国贝尔实验室的 Dennis Ritchie 于 1972 年开发出来的，并首次在 UNIX 操作系统的 DEC PDP-11 计算机上使用，C 语言是由早期的 B 语言发展演变而来的。在 1970 年，贝尔实验室的 Ken Thompson 根据 BCPL(Basic Combined Programming Language)语言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，所以 Dennis Ritchie 在此基础上开发出了 C 语言，C 语言既保持了 B 语言的优点，又克服了它的缺点。

最初的 C 语言只能在大型计算机上执行，随着微型计算机的日益普及，它被移植到微机上来，并且出现了许多不同版本的 C 语言。由于没有统一的标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，1983 年美国国家标准化协会(ANSI)为 C 语言制定了标准，即 ANSI C，1987 年，ANSI 又公布了新的标准，即 87 ANSI C。现在流行的各种 C 语言版本都是以它为标准的。微机上正在使用的 C 语言有 Turbo C、Borland C、Microsoft C、Quick C 等。

1.2　C 语言的特点

C 语言由于其功能强大，早已成为最受欢迎的语言之一。许多著名的软件都是用 C 语言编写的。C 语言具有如下一些特点。

(1) 语言简洁、紧凑，使用方便、灵活，具有丰富的运算符和数据结构。C 语言一共只有 32 个关键字、9 种控制语句、34 种运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理，从而使得 C 语言的运算类型极其丰富，表达式类型多样化。C 语言的数据类型有：整型、实型、字符型、枚举类型、数组类型、指针类型、结构体类型、共用体类型等，使用这些数据类型可以实现各种复杂的数据结构运算。

(2) C 语言允许直接访问物理地址，能进行位操作，能实现汇编语言的大部分功能，可以直接对硬件进行操作。因此，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，可用来编写系统软件。C 语言既是成功的系统描述语言，又是通用的程序设计语言，人们通常称之为“中级语言”，即它兼有高级和低级语言的特点。

(3) 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)，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，符合现代编程语言风格的要求。

(4) 语法限制不太严格，程序设计自由度大。例如，对数组下标越界不作检查，由程序编写者自己来保证程序的正确性。对变量的类型使用比较灵活，例如，整型数据与字符型数据以及逻辑型数据可以通用。一般的高级语言语法检查比较严格，能检查出几乎所有的语法错误。而 C 语言允许程序编写者有较大的自由度，因此放宽了语法的检查。程序员应当仔细检查程序，来保证其正确，而不要过分依赖 C 编译程序来检查错误。

(5) 用 C 语言编写的程序可移植性好(与汇编语言相比)。在某一系统编写的程序，基本上不作任何修改就能用于其他类型的计算机和操作系统上运行。

(6) 生成目标代码质量高，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%。

C 语言的以上特点，使得 C 语言功能强大、应用广泛，用 C 语言可以编写出任何类型的程序，它既可用来编写系统软件，也可以用来编写各种应用软件。但同时 C 语言对编程人员也提出了更高的要求，编程人员学习 C 语言和学习其他的高级语言相比，必须花费更多的心思在学习 C 语言的语法上，尤其是指针的应用，常让初学者摸不着边际。但是，一旦熟悉了 C 语言的语法，便可以享受到 C 语言所带来的便利性与快捷性。

1.3　C 语言源程序举例

通过上一节的介绍，我们了解了一些 C 程序的特点，下面通过几个简单的 C 程序实例，让我们进一步来分析 C 程序的结构特点。

例 1.1　编写一个 C 语言程序，在屏幕上显示两行信息，分别是“How are you！”和“Welcome you!”。

程序代码如下：

```
# include <stdio.h>
int main( )
  {
      printf("How are you!\n");
      printf("Welcome you!");
      return 0;
      }
```

程序运行的结果是输出两行文本信息，如下：

```
How are you!
Welcome you!
```

C 程序是由许多函数组合而成的，而在函数里面又可以再调用其他函数。上面的程序中，main 表示“主函数”，每一个 C 程序都必须有一个 main 函数，它是程序执行的入口，main 前面的 int 表示函数的返回类型，即 main 函数为整型类型。程序中一对大括弧 { } 括起来的部分称为函数体。函数体内的 printf 是 C 语言中的输出函数，双引号内的字符串按原样输出，“\n”是换行符，即在输出“How are you!”之后回车换行，然后在屏幕的下一行输出“Welcome you!”，每个语句结尾为一个分号。函数体内的 return 语句为主函数结束时的返回值，由于 main 函数的类型为整型(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。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来，stdio 为 standard input/output 的缩写，即标准输入输出，C 语言中有关输入输出函数的格式均定义在这个文件里。

例 1.2　计算两个整数 a,b 之和，并在屏幕上显示结果。

程序代码如下：

```
#include <stdio.h>
int main (  )                          /*主函数*/
{
  int a,b,sum;                         /*定义变量*/
  a=111;b=222;                         /*为变量赋值*/
  sum=a+b;                             /*求两数之和*/
  printf ("sum is: %d", sum);          /*输出 sum 的值*/
  return 0;
  }
```

程序运行的结果是输出两个整数 a 和 b 的和 sum，显示结果如下：

```
sum is: 333
```

在程序中，/*……*/表示注释部分，为了便于理解，我们用汉字表示注释，当然也可以用英语或汉语拼音作注释。注释只是用于解释程序，对编译和运行不起任何作用。本程序中，在函数体内(即一对大括号之间)的第一行是变量定义部分，定义了 3 个整型变量；第二行是两个赋值语句，使 a 和 b 的值分别为 111 和 222；第三行使 sum 的值为 a 和 b 之和，即为 333；第四行 printf 是输出函数，其中的“%d”表示输出 sum 时的数据类型和格式为“十进制整数类型”，在执行输出时，此位置上代以一个十进制整数值，printf 函数中括弧内最右端的 sum 是要输出的变量，现在它的值是 333，因此输出的信息为“sum is: 333”。

例 1.3　输入两个整数，调用自定义函数来计算 a、b 之和，并在屏幕上输出结果。

程序代码如下：

```
#include <stdio.h>
int sumab (int x, int y);                  /*函数声明*/
int main (  )                              /*主函数*/
{
  int a,b,sum;                             /*定义变量*/
  printf("input a and b:");                /*提示字符串*/
  scanf ("%d %d", &a, &b);                 /*输入变量 a 和 b 的值*/
```

```
    sum=sumab(a,b);                    /*调用 sumab 函数*/
    printf("sum=%d", sum);             /*输出 sum 的值*/
    return 0;
}
int sumab (int x, int y)               /*定义 sumab 函数，并定义形参 x、y */
{
    int z;
    z=x+y;
    return z;
}
```

程序由两个函数组成，即由主函数 main 和函数 sumab 组成。函数 sumab 的功能是求两个整数之和并返回给主函数。sumab 函数是一个用户自定义函数，有两个整型的形参 x 和 y，它是一个具有整型类型返回值的函数。main 函数前面的函数声明语句“int sumab (int x, int y);”表明 sumab 是一个有两个整型的形参并返回一个整型类型值的函数。这样的函数声明叫做函数原型，它要与函数的定义和调用相一致。

本程序的执行过程如下：首先在屏幕上显示提示字符串，请用户输入两个数，回车后由 scanf 函数语句接收这两个数并送入变量 a、b 中，然后调用 sumab 函数，并把 a 和 b 的值传递给 sumab 函数的参数 x 和 y，在 sumab 函数中，计算 x 和 y 二者之和赋给变量 z，并由 return 语句把变量 z 的值返回给主函数 main，然后赋值给变量 sum，最后由 printf 函数在屏幕上输出 sum 的值。

从以上 3 个例子可以看出，C 源程序的结构特点如下。

(1) 一个 C 语言源程序由若干个函数构成，其中有且仅有一个主函数(main 函数)。

(2) 一个函数由函数首部(即函数第一行)和函数体(即函数首部下面的大括号内的部分)组成。函数首部包括函数类型、函数名和放在圆括号内的若干个参数。函数体由声明部分和执行部分组成。

(3) C 程序书写格式自由，一行内可以写多条语句，一个语句也可以分写在多行中，且语句中的空格和回车符均可忽略不计。

(4) 程序的注释内容放在/*和*/之间，/和*之间不允许有空格；注释部分允许出现在程序中的任何位置。

1.4　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，经修改认为无误后，保存为一个文件。C 源程序文件的后缀为“.C”。可用于编写 C 源程序的编辑软件有很多，而在本书中，DOS 环境下，使用 Turbo C；Windows 环境下，使用 WIN TC(Turbo C 的 Windows 版本)。

2. 编译程序

程序编辑完之后，在 Turbo C 或 WIN TC 下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，则系统会输出出错信息，告诉用户第几行有错误，用户重新修改源程序再进行编译，如此反复直至编译通过为止。编译通过后生成目标程序，目标程序的文件名与相应的源程序同名，但后缀为“.obj”。

3. 连接程序

将目标程序和库函数或其他目标程序连接，即可以生成可执行程序，可执行程序的文件名与相应的源程序同名，但后缀为“.exe”。在 Turbo C 或 WIN TC 下是通过快捷键或选择菜单的方式进行连接的。

4. 运行程序

只要输入可执行文件的文件名即可运行程序。在 Turbo C 或 WIN TC 下是通过快捷键或选择菜单的方式运行程序的。

上述的编辑、编译、连接、运行程序的过程如图 1.1 所示。

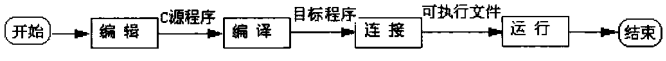


图 1.1　C 程序的执行过程示意图

1.5 习　题

1. 简述 C 程序的结构特点。

2. 写出一个 C 程序的构成。

3. 分别编写完成如下任务的程序，然后上机编译、连接并运行。

(1) 输出两行字符，第 1 行是“The computer is our good friends!”，第 2 行是“We learn C language.”。

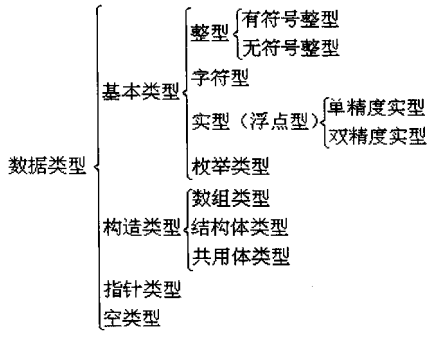
(2) 从键盘输入变量 a、b 的值，分别计算 a+b、a-b 的值，将计算结果分别存放在变量 c、d 中，最后输出计算结果。

第2章　数据类型、运算符与表达式

本章主要介绍C程序中经常用到的常量、变量、基本数据类型(整型、实型、字符型)、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、求字节运算符)和表达式(算术表达式、赋值表达式、逗号表达式)等内容。

2.1　C语言的数据类型

在程序中要使用数据，对每一个数据都要指定其数据类型，C语言提供了如下数据类型：



本章将介绍整型、实型和字符型的用法。

2.2　常量和变量

2.2.1　常量

在程序运行过程中，其值不能被改变的量称为常量。

常量分为以下几种：

(1) 整型常量(如369、0、-547)；

(2) 实型常量(如2.71828、-9.8、3.14159)；

(3) 字符常量(如‘A’、‘a’、‘#’、‘3’)；

(4) 符号常量——用一个标识符代表一个常量。例如，如果在程序开始有这样的预处理命令：“#define　N　10”，那么C预处理程序会将程序中所有的N都用10代替。

2.2.2　变量

在程序运行过程中，其值可以被改变的量称为变量。

在使用一个变量之前，必须先定义该变量，就是为该变量起个名字并声明其数据类型。根据定义，编译系统在内存中为该变量分配存储单元，在该存储单元中存放该变量的值。

用来标识变量名(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，标识符只能由英文字母、数字、下划线三种字符组成，并且第一个字符必须是字母或下划线。

注意，大写英文字母和小写英文字母是不同的字符，例如 aver 和 Aver 是两个不同的标识符。为变量起名字时一般用小写英文字母。

变量定义的一般格式如下：

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

例如：　int　a, b, number, sum;

在定义变量的同时对变量进行赋初值的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：float　radius=2.5, length=3.6, area;

2.3　整 型 数 据

2.3.1　整型常量

整型常量即整常数，在C语言中，整型常量可以用如下 3 种形式表示：

(1) 十进制，例如 456、0、-789。

(2) 八进制(以数字 0 开头)，例如 0123，即$(123)_8$，等于十进制的 83。

(3) 十六进制(以数字 0 +小写字母 x 开头)，例如 0x23，即$(23)_{16}$，等于十进制的 35。

2.3.2　整型变量

根据变量的取值范围，整型变量可分为：基本整型(类型关键字为 int)、短整型(类型关键字为 short　[int])、长整型(类型关键字为 long　[int])。

对于以上 3 种都可以加上修饰符 unsigned，以指定是“无符号数”。不加修饰符 unsigned 的，隐含是有符号(signed)。即有符号的，signed 可以省略不写。

归纳起来，整型变量有以下 6 种：

有符号基本整型　　[signed] int

无符号基本整型　　unsigned [int]

有符号短整型　　[signed] short [int]

无符号短整型　　unsigned short [int]
有符号长整型　　[signed] long [int]
无符号长整型　　unsigned long [int]

其中，方括弧内的部分可以省略，如 unsigned long [int]与 unsigned long 等价。

例如，下面分别定义了有符号基本整型变量 a 和 b、无符号长整型变量 c 和 d：

```
int    a, b;
unsigned long    c, d;
```

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(0 为正，1 为负)。若指定是无符号型 unsigned，则存储单元的全部二进制位(bit)都用来存放数本身，而不包括符号。

C 标准没有具体规定以上各类数据所占内存大小，只要求 long 型数据不短于 int 型，short 型不长于 int 型，怎样实现由计算机系统自行决定。例如，在微机上，short 型和 int 型占 2 个字节，long 型占 4 个字节。

表 2.1 列出了 ANSI 标准定义的各种整数类型和有关数据，其中“最小取值范围”是指不能低于此值，但可高于此值，如有的 C 编译系统规定一个 int 型数据占 4 个字节。

注意，一个基本整型变量只能取-32768~32767 范围内的整数，超出范围就会发生“溢出”现象，但程序运行并不报错。其他类型也有取值范围，超出范围也会“溢出”。所以要根据实际情况，准确选择变量的类型，避免超出能取值范围。

表 2.1　ANSI 标准定义的各种整数类型和有关数据

类　型	字 节 数	最小取值范围
[signed] int	2	-32768~32767　即　-2^{15}~$(2^{15}-1)$
unsigned [int]	2	0~65535　即　0~$(2^{16}-1)$
[signed] short [int]	2	-32768~32767　即　-2^{15}~$(2^{15}-1)$
unsigned short [int]	2	0~65535　即　0~$(2^{16}-1)$
[signed] long [int]	4	-2147483648~2147483647　即　-2^{31}~$(2^{31}-1)$
unsigned long [int]	4	0~4294967295　即　0~$(2^{32}-1)$

2.3.3　整型数据的输入输出

可以使用 scanf 函数和 printf 函数进行数据的输入与输出。

scanf 函数的功能是按照指定格式将标准输入设备输入的内容送入变量中，printf 函数的功能是按照指定格式在标准输出设备上显示数据。“指定格式”需要使用格式说明符%和格式字符，显示整型数的格式字符有英文字母 d、o、x、u 等。

具体含义如下：

%d——表示把数据按十进制整型输入(输出)；

%o——表示把数据按八进制整型输入(输出)；

%x——表示把数据按十六进制整型输入(输出)；

%u——表示把数据按无符号整型输入(输出)。

除了%d 格式之外，上面的其余几种格式都将数据作为无符号数进行输入(输出)。

如果输入(输出)的是长整型数，一定要在转换字符的前面加上字符 l(字符 L 的小写)，否则显示可能不对。

例 2.1　整型数据的输出。

```
#include <stdio.h>
int main()
{ int a=200,b=100,c;
  c=a+b+15;
  printf("%d,%d,%d,%d\n", a,b,c,a-b-70);
  printf("%o,%o,%o,%o\n", a,b,c,a-b-70);
  printf("%x,%x,%x,%x\n", a,b,c,a-b-70);
  getch();
  return 0;
}
```

输出结果如下：

```
200, 100, 315, 30
310, 144, 473, 36
C8, 64, 13b, 1e
```

例 2.2　整型数据的输入。

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

若输入为：

```
10, 10, 10, 65533, 654321 ↙(回车符)
```

则输出结果为：

```
10, 8, 16, 65533, 654321
```

2.4　实 型 数 据

2.4.1　实型常量

实数又称浮点数，有两种表示形式：

(1) 十进制小数形式：由数字和小数点组成(必须有小数点)，例如 3.14159、0.0、

9.0、.12345、-9.8 等。

(2) 指数形式：<尾数>E(或 e)<指数>。例如，1.23E3、2.71e-5(分别代表 1.23×10^{3}、2.71×10^{-5})等。注意：E(或 e)的两边必须有数字，且后面的指数必须是整数。

一个实数有多种指数表示形式。例如，314.159 可以表示为 3141.59E-1、314.159E0、3.14159E2、0.314159E3 等，把其中的 3.14159E2 称为“规范化的指数形式”，即小数点左边有且只有一位非零数字。

2.4.2 实型变量

实型变量分为单精度型和双精度型，有的 C 版本还支持长双精度型(long double)。

(1) 单精度型：类型说明符为 float，在内存中占 4 个字节(32 位)，有效数字的个数是 7 位十进制数字，数值范围为-3.4×10^{-38}~3.4×10^{38}。

(2) 双精度型：类型说明符为 double，在内存中占 8 个字节(64 位)，有效数字的个数是 15 位十进制数字，数值范围为-1.7×10^{-308}~1.7×10^{308}。

2.4.3 实型数据的输入输出

可以使用%f 和%e 来控制输入(输出)float 类型的数据，使用%lf 和%le 控制输入(输出) double 类型的数据。

例 2.3 实型数据的输入输出。

```
#include <stdio.h>
int main()
{ float a,b; double x,y;
   scanf("%f,%e,%lf,%le", &a,&b,&x,&y);
   printf("%f,%e,%lf,%le \n", a,b,x,y);
   return 0;
}
```

若输入为：

3.1415, 314.15, 123.456, 12345.6 ↙(回车符)

则输出结果为：

3.141500, 3.14150e+02, 123.456000, 1.23456e+04

若输入为：

3.1415926, 666.666666, 123456789.123456789, 123456.7898765 ↙

则输出结果为：

3.141593, 6.66667e+02, 123456789.123457, 1.23457e+05

从结果可知：对于十进制小数形式，单精度型和双精度型的有效数字分别是 7 位和 15 位。对于十进制指数形式，都是 6 位有效数字。