计算机程序设计科普读物
C 程序设计教学配套教参
计算机等级考试与程序设计竞赛参考书

趣味C程序设计集锦

杨克昌 刘志辉 编著

爱上程序设计

本书特色

- 突出求解问题生动有趣
- 注重求解程序简单易懂
- 力求问题结果直观明了
- 重视程序变通与问题引申

中国水利水电出版社
www.waterpub.com.cn

趣味 C 程序设计集锦

杨克昌　刘志辉　编著

中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书作为计算机 C 程序设计的科普读物与学习 C 语言程序设计的教学参考书，着眼于应用 C 程序设计求解问题的基本方法与技巧，提高通过 C 程序设计解决实际问题的能力。

本书以各类趣题的 C 程序设计求解为主线，取材注重典型性与趣味性，程序注重结构化与可读性。所精选的趣解包括典型的数值求解、常见的数据处理、有趣的智力游戏、巧妙的模拟探索、新颖的图表创建，大多是引导入门的基础题、常规题，也适当设计少量难度较大的综合题与经典名题，难度适宜，深入浅出。

为适应计算机基础不同的读者学习与欣赏，对有些趣题采用多种思路设计与多个程序实现。所有程序均可在 TC、WIN-TC 及 VC++（除涉及作图的程序外）中运行通过。其中少量难度较大、要求较高的问题在目录中用“*”标注，可供在校学习“C 程序设计”课程的同学们进行课程设计时选用。

本书适合普通高校本专科学生、职业技术学院学生与程序设计爱好者学习 C 程序设计参考，还可供各级程序设计选拔赛、计算机等级考试与计算机程序员水平考试复习参考，也可供中学信息学（计算机）奥林匹克指导与 IOI、NOI 培训选用。

图书在版编目（CIP）数据

趣味C程序设计集锦 / 杨克昌，刘志辉编著. -- 北京 : 中国水利水电出版社，2010.1
ISBN 978-7-5084-7068-9

Ⅰ. ①趣… Ⅱ. ①杨… ②刘… Ⅲ. ①C语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2009)第228423号

策划编辑：杨庆川　　责任编辑：张玉玲　　封面设计：李　佳

书　　名	趣味 C 程序设计集锦
作　　者	杨克昌　刘志辉　编著
出版发行	中国水利水电出版社 （北京市海淀区玉渊潭南路 1 号 D 座　100038） 网址：www.waterpub.com.cn E-mail：mchannel@263.net（万水） sales@waterpub.com.cn 电话：（010）68367658（营销中心）、82562819（万水）
经　　售	全国各地新华书店和相关出版物销售网点
排　　版	北京万水电子信息有限公司
印　　刷	北京市天竺颖华印刷厂
规　　格	170mm×237mm　16 开本　23 印张　470 千字
版　　次	2010 年 1 月第 1 版　2010 年 1 月第 1 次印刷
印　　数	0001—3000 册
定　　价	35.00 元

前　　言

从当前高校各专业计算机语言课程的系统开设，到中小学信息技术（计算机）课的相继开出，推动了计算机文化在神州大地的广泛普及。顺应信息技术迅猛发展与计算机教育不断深入的潮流，帮助包括各大中专在校学生在内的广大青少年逐步掌握计算机程序设计的设计思路与基本技能，在程序设计中开拓求解思路，解决实际问题，培养创新意识，不断激发提出问题与应用程序设计解决问题的欲望与兴趣，不断提高程序设计水平与应用求解能力，是我们计算机教育工作者义不容辞的职责。

当前，大专院校理工科专业一般学习 C（C++）语言程序设计，而文科专业一般学习 VFP 程序设计。各种面向大专院校的程序设计教材繁杂而重复，而配合程序设计教学的参考书与课外读物却是凤毛麟角。对此，新推出计算机程序设计科普读物《趣味 C 程序设计集锦》和《趣味 Visual FoxPro 程序设计集锦》，可望推动广大在校本专科学生与程序设计爱好者学习程序设计的不断深入，促进程序设计水平与应用求解能力的逐步提高。

学习计算机语言的目的是什么？当然是程序设计！那么，程序设计的目的又是什么？毫无疑义，程序设计的目的是求解问题，求解一般靠人工计算或推理一时无从下手的各类实际应用问题。你应用程序设计解决的问题越多、越普遍、越深刻，你的程序设计成绩就越大，你的程序设计能力就越强。而应用程序设计求解各类实际应用问题，恰恰是现有大多程序设计教材所忽略的环节，也是造成广大在校学生学习程序设计语言缺乏兴趣的直接原因。

本书中以各类中外趣味问题的程序设计求解为主线，取材注重趣味性与典型性，题型丰富，内容新颖。精选的程序设计趣味问题包括各类整数求解、数据处理、智力游戏、模拟探索、图表创建，大多是引导入门的基础题与常规题，也适当设计少量难度较大的综合题与经典名题，难度适宜，深入浅出。书中部分选题取自国际国内信息学（计算机）奥林匹克与各类程序设计竞赛，同时参考了网上读者集中探讨的程序设计热点问题，有利于高校学生与程序设计爱好者在计算机实例求解上开阔视野，在程序设计思路开拓与应用技巧上有一个深层次的练习与提高。部分难度较大、要求较高的问题在目录中用“*”标注，可供在校学生进行课程设计时选用。

本编著力求突出以下 4 个特色：

（1）突出**求解问题生动有趣**。选取的程序设计趣题应为程序设计爱好者所喜爱。这些趣题通常是著名的中外名题或应用常规的推理与人工计算难以解决的问题，以充分体现程序设计求解的优势。为了避免一味拔高而降低趣味性，删除了若干枯燥难懂、学术性较强的问题。

前　言

（2）注重**求解程序简单易懂**。题解程序选用广大青少年与在校本专科学生正在学习的、使用率最高的计算机语言编写程序。同时在算法上尽量避免使用学术性强的专业算法。对其中有些典型问题采用不同设计思路与表现形式设计出不同的求解程序，以适应程序设计基础不同的读者学习与欣赏。

（3）力求**问题结果直观明了**。尽可能给出求解程序的输出结果与运行示例，并作必要的讨论与分析，使读者对问题求解结果一目了然，以帮助读者对所求解问题的清晰理解与对设计程序的深入掌握。

（4）重视**程序变通与问题引伸**。对求解程序进行必要的改进、变通与优化，是促进应用程序设计解决实际问题能力的培养，切实提高程序设计水平的必要手段与有效途径。同时，对有些求解趣题作适当的引伸与拓广，引导有兴趣的爱好者对相关问题作进一步的探索与研究。

本书作为计算机程序设计的教学参考书与科普读物，适合在校本专科学生与广大程序设计爱好者学习参考，还可供各级程序设计选拔赛与国际大学生程序设计竞赛（ACM）、计算机等级考试与计算机程序员水平考试复习使用，也可供中学信息学（计算机）奥林匹克指导与 IOI、NOI 及各省程序设计竞赛培训参考。

在书稿的编写过程中，湖南理工学院计算机学院院长王岳斌教授、周持中教授以及严权锋、郭华等老师提出了很好的修改意见，笔者在此深表感谢。

尽管每一道题解都经过反复检查，每一个程序都经过多次运行调试，但因涉及内容较广，难免存在各种差错，恳请广大读者批评指正。

编　者

2009 年 10 月

目　录

一、舍罕王的失算——不可忽视的和与积

1　舍罕王的失算

1. 问题提出

相传现在流行的国际象棋是古印度舍罕王（Shirham）的宰相达依尔（Dahir）发明的。舍罕王十分喜爱象棋，决定让宰相自己要求得到什么赏赐。这位聪明的宰相指着 8×8 共 64 格的象棋盘说：陛下，请您赏给我一些麦子吧，就在棋盘的第 1 个格中放 1 粒，第 2 格放 2 粒，第 3 格放 4 粒，以后每一格都比前一格增加一倍，依此放完棋盘上的 64 格，我就感恩不尽了。

舍罕王让人扛来一袋麦子，他要兑现他的许诺。

请问，国王能兑现他的许诺吗？共要多少麦子赏赐他的宰相？合多少立方米（1 立方米麦子约 1.42e8 粒）？如果把这些麦子堆成一个正圆锥形的麦堆，这麦堆约有多高？

2. 求解要点

这是一个典型的等比数列求和问题。

第 1 格 1 粒，第 2 格 2 粒，第 3 格 $4=2^2$ 粒，……，第 i 格为 2^{i-1} 粒，于是总粒数为

$$s = 1 + 2 + 2^2 + 2^3 + \cdots + 2^{63}$$

为一般计，设共有 n 个格。

设置求和 i（2～n）循环，在循环中通过 t=t*2 计算第 i 格的麦粒数，体现每一格为其前一格的 2 倍。再通过 s=s+t 把每一格的麦粒数累加到和变量 s，即可实现该等比数列求和。

求出的总粒数为 s，折合为 v 立方米，则有：v=s/1.42e8。

设正圆锥形的麦堆的底半径为 r，高为 h，则有：

$$v = \frac{1}{3}\pi r^2 h,\ h = \sqrt{3}\,r \quad \Leftrightarrow \quad h = \sqrt[3]{\frac{9}{\pi}v}$$

由和 s 算出体积 v，再算出高 h。

3. 程序实现

```
/* 舍罕王的失算 c011 */
#include<stdio.h>
```

```
#include<math.h>
void main()
{
  double  t,s,v,h; int i,n;
  printf(" 请输入格数 n: ");
  scanf("%d",&n);              /* 输入格数 n */
  t=1;s=1;
  for(i=2;i<=n;i++)
  {
    t=t*2;                     /* t 为第 i 格的麦粒数 */
    s=s+t;                     /* s 求所有格的麦粒和 */
  }
  v=s/1.42e8;                  /* 求出所有麦子合多少立方米 */
  h=v*9/3.1415926;
  h=pow(h,1.0/3);              /* 求出麦堆的高 */
  printf(" 总麦粒数约为: %.3e\n",s);
  printf(" 折合体积约为: %.0f 立方米\n",v);
  printf(" 正圆锥麦堆高约为: %.0f 米\n",h);
}
```

4. 程序运行结果与说明

运行程序，请输入格数 n：64

总麦粒数约为：1.845e+019

折合体积约为：129906648406 立方米

正圆锥麦堆高约为：7193 米

这是一个非常宠大的数值，相当于全世界若干年产的全部小麦。如果把这些麦子堆成一个正圆锥形的麦堆，这麦堆的高超过 7 千米，可与世界最高峰珠穆朗玛峰一比高低了。看来舍罕王失算了，他无法兑现他的诺言。

程序中设置通项 t，从 t=1 开始，应用 t=t*2 累乘计算通项。显然，当 i=2 时，t=2；当 i=3 时，t=4；……，这一处理技巧是常用的。尤其是在要求通项与和必须是准确值时，采用上述累乘而不用通项公式 2^(i-1)。

一般对有通项公式 f(i)的求和，通常在循环中应用 s=s+f(i)实现求和。

5. 失算的另一名题——买马钉

（1）问题提出

类似棋盘格放麦粒问题，俄罗斯学者马哥尼茨基的《算术》中有这样一道趣题：

某人以 156 卢布卖出一匹马。成交后，买主后悔并向卖主说："我上当了，你的马不值这个价钱。"

这时卖主提出另一笔交易："你既然嫌马太贵，那么你买马掌钉好了，这匹马就白送给你。每个马掌要钉 6 个钉，共需 24 个钉。钉的价格按如下方法计算：第 1 个钉 1 个包卢斯卡（俄罗斯货币单位，相当于 1/4 个卢比），第 2 个钉 2 个包卢斯卡，第 3 个钉 4 个包卢斯卡，……，以后每一个钉的价格为前一个钉的 2 倍，直到第 24 个。"

买主听后暗想，钉子如此便宜，马还可白得，欣然同意。

问：买主买马钉需要花多少卢布（1 卢布等于 100 卢比，即 400 包卢斯卡）？

（2）买马钉程序设计

```
/* 买马钉 c012*/
#include<stdio.h>
void main()
{
  double  t,s; int i,n;
  printf("  请输入 n: ");
  scanf("%d",&n);             /* 输入买马钉总数 n */
  t=1;s=1;
  for(i=2;i<=n;i++)
     {
       t=t*2;                 /* t 为第 i 个马钉的价格 */
       s=s+t;                 /* s 统计总价（包卢斯卡个数）*/
     }
  printf("  共需花费约 %.4f 卢布。",s/400); /* 换算成卢布输出 */
}
```

（3）程序运行结果与点评

请输入 n: 24

共需花费约 41943.0375 卢布。

以上两例历史上广为流传的名题求解实际上是求一个等比数列之和。这两题的计算结果给我们以启示：不可低估等比数列的和，不可忽略求和与求积等计算基础。

2　分数不等式

1. 问题提出

试解下列两个关于正整数 m 的分数不等式：

$$10 < 1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{m} < 11 \qquad (1)$$

$$4 < 1+\frac{1}{2}-\frac{1}{3}+\frac{1}{4}+\frac{1}{5}-\frac{1}{6}+\cdots\pm\frac{1}{m}$$ （符号为两个"+"后一个"-"号）

(2)

2. 分数不等式（1）求解

（1）设计要点

为一般计，解不等式

$$n < 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{m} < n + 1 \quad (3)$$

这里正整数 n 从键盘输入。

设和变量为 s，在 s<n 的条件循环中，累加求和 s=s+1/i，直至出现 s>n 退出循环，赋值 c=i，所得 c 为 m 解区间的下限。

继续在 s<n+1 的条件循环中累加求和 s=s+1/i ，直至出现 s>n+1 退出循环，通过赋值 d=i-1，所得 d 为 m 解区间的上限。

然后打印输出不等式的解区间[c, d]。

（2）程序实现

```
/* 求解调和级数不等式 c021 */
#include <stdio.h>
void main()
{
  long c,d,i,n;
  double s;
  printf("求 n<1+1/2+1/3+1/4+…+1/m<n+1 的整数 m。");
  printf("\n 请输入 n:");
  scanf("%ld",&n);
  i=0;s=0;
  while(s<n)
    {i=i+1;s=s+1.0/i;}
  c=i;
  while(s<n+1)
    {i=i+1;s=s+1.0/i;}
  d=i-1;
  printf("\n 满足不等式的 m 为: %ld≤m≤%ld \n",c,d);
}
```

（3）程序运行结果

```
求 n<1+1/2+1/3+1/4+…+1/m<n+1 的整数 m。
  请输入 n: 10
  满足不等式的 m 为: 12367≤m≤33616
  请输入 n: 15
  满足不等式的 m 为: 1835421≤m≤4989190
```

3. 分数不等式（2）求解

（1）设计要点

一般地，解不等式

$$d < 1+\frac{1}{2}-\frac{1}{3}+\frac{1}{4}+\frac{1}{5}-\frac{1}{6}+\cdots\pm\frac{1}{m} \qquad (4)$$

其中 d 为从键盘输入的正数，式中符号为两个“+”号后一个“-”号，即分母能被 3 整除时为“-”。

式中出现减运算，导致不等式的解可能分段。

设置条件循环，每三项（包含二正一负）一起求和，得一个区间解。

然后回过头来一项项求和，得个别离散解。

（2）程序设计

```
/* 解不等式：d<1+1/2-1/3+1/4+1/5-1/6+…+-1/m  c022 */
#include <stdio.h>
void main()
{
  long d,m,k;
  double s;
  printf("\n  请输入 d: ");
  scanf("%d",&d);
  printf("  %d<1+1/2-1/3+1/4+1/5-1/6+…+-1/m 的解：",d);
  m=1;s=0;
  while(1)
     {
       s=s+1.0/m+1.0/(m+1)-1.0/(m+2);
       if(s>d) break;
       m=m+3;
     }
  printf("\n  m>=%ld",m);              /* 得一个区间解 */
  k=1;s=0;
  while(k<m)
    {
      if(k%3>0) s=s+1.0/k;
      else s=s-1.0/k;
      if(s>d)                          /* 得一个离散解 */
           printf("\n  m=%ld",k);
      k++;
    }
}
```

（3）程序运行示例

运行程序，请输入 d：4

    4<1+1/2-1/3+1/4+1/5-1/6+…+-1/m 的解：

    m>=10153

    m=10151

注意：前一个是区间解，后一个是离散解。要特别注意，不要把后一个解遗失。

3　阶乘与阶乘和数

1．问题提出

计算 n 的阶乘：n ! = 1 · 2 · 3 · … · n（正整数 n 从键盘输入）。

2．常规求解

设置 k（k=1，2，…，n）循环，在循环中用 t=t*k 实现阶乘计算。

求 n!的常规设计：

```
/* 求 n 的阶乘 n! c031 */
#include <stdio.h>
void main()
{
  int k,n; long t;
  printf("  请输入 n: ",&n);
  scanf("%d",&n);
  t=1;
  for(k=1;k<=n;k++)
     t=t*k;          /* 循环变量 k 累乘到 t，体现阶乘运算规律  */
  printf("%d!=%ld\n",n,t);
}
```

运行程序，输入 n=12，得

    12!=479001600

3．递归设计

递归定义：n!=n*(n-1)!

边界条件：1!=1

设置 fac(n)=n!，当 n=3 时，fac(3)的递归求解可表示为：

    fac(3)=3*fac(2)=3*2*fac(1)=3*2*1*1=6

求 n!的递归设计：

```
/* 递归求n的阶乘n! c032 */
long  fac(int n)
{
  long f;
  if(n==1) f=1;                    /*  初始条件  */
  else f=n*fac(n-1);               /*  递归关系  */
  return (f);
}
#include <stdio.h>
void main()
{
  int n; long y;
  printf(" 请输入 n: ");
  scanf("%d",&n);
  y=fac(n);
  printf("%d!=%ld\n",n,y);
}
```

运行程序，输入 n=10，得

  10!=3628800

4．变通为求排列数 A(n,m)与组合数 C(n,m)

（1）设计要点

注意到从 n 个元素中取 m 个的排列数为

  A(n,m)=(n-m+1)×(n-m+2)×…×n

而从 n 个元素中取 m 个的组合数为

  C(n,m)=A(n,m)/m!=(n-m+1)/1×(n-m+2)/2×…×n/m

则只要把求阶乘的程序 k 从 1 开始到 n 循环改变为从 n-m+1 开始到 n 循环。

求排列数：t=t*k

求组合数：t=t*k/(k-n+m)

（2）求排列数 A(n,m)与组合数 C(n,m)程序设计

```
/* 选择计算排列数 A(n,m) 与组合数 C(n,m) c033 */
#include <stdio.h>
void main()
{
  int k,m,n,z; long t;
  printf(" 请选择：1 计算排列数 , 2 计算组合数： ");
  scanf("%d",&z);
  printf(" 请输入 n,m:(n>=m) ");
  scanf("%d,%d",&n,&m);
```

```
  t=1;
  for(k=n-m+1;k<=n;k++)
     if(z==1)
        t=t*k;                /* 变量k累乘到t，计算排列数 */
     else
        t=t*k/(k-n+m);        /* k/(k-n+m)累乘到t，计算组合数 */
  if(z==1)
        printf(" A(%d,%d)=%ld\n",n,m,t);
  else
        printf(" C(%d,%d)=%ld\n",n,m,t);
}
```

（3）程序运行示例

请选择：1 计算排列数，2 计算组合数：1

请输入 n,m:(n>=m) 10,8

A(10,8)=1814400

请选择：1 计算排列数，2 计算组合数：2

请输入 n,m:(n>=m) 30,20

C(30,20)=30045015

5. 阶乘和数

一个正整数如果等于组成它的各位数字的阶乘之和，该正整数称为阶乘和数。试求三位阶乘和数，进而求出所有阶乘和数。

（1）试求出所有三位阶乘和数：abc=a!+b!+c!

（其中 a 为百位数字，b 为十位数字，c 为个位数字。约定 0!=1）。

1）设计思路

通过循环累乘设计一个求阶乘的函数（子程序）：jc(x)=x!。

对任意一个三位数 m，分解其百位数字 a，十位数字 b，个位数字 c。

条件判别：若 m 等于 jc(a)+jc(b)+jc(c)，则作打印输出。

也可通过 a、b、c 三重循环组合为三位数 m=a*100+b*10+c，然后作条件判别。

由于 7!=5040>999，因此 a、b、c 都必定小于 7。

2）求三位阶乘和数的程序设计

```
/* 求三位阶乘和数 c034 */
long jc(int x)              /* 定义阶乘函数 jc(x)=x! */
{
  int i;long p=1;
  for(i=1;i<=x;i++)
  p*=i;
  return (p);
```