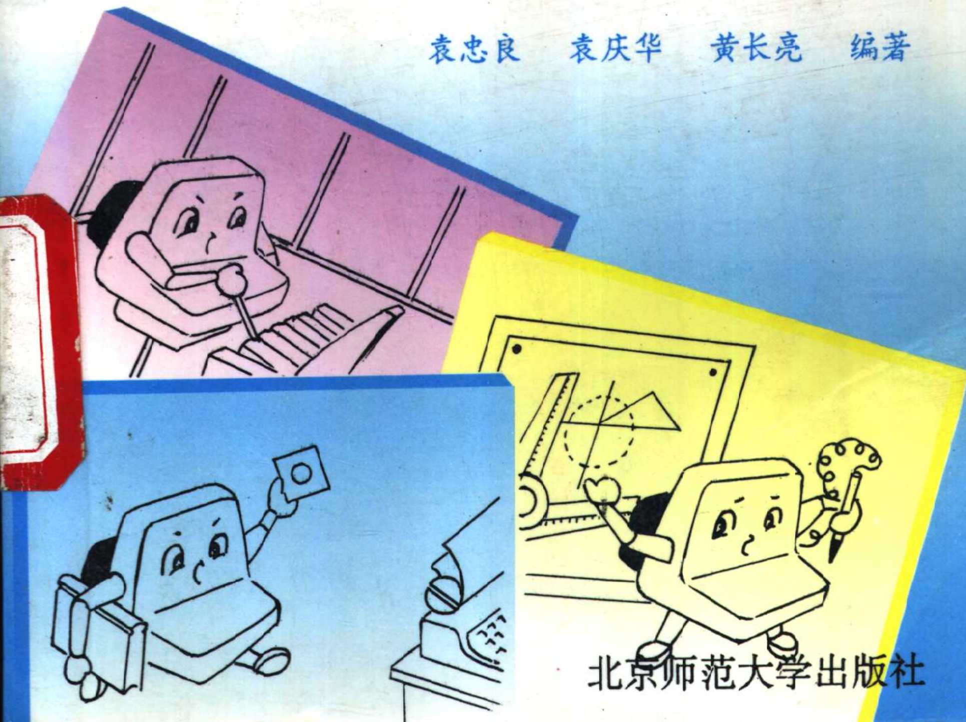
计算机趣味程序设计丛书

趣味BASIC程序

袁忠良　袁庆华　黄长亮　编著



北京师范大学出版社

计算机趣味程序设计丛书

趣味BASIC程序

袁忠良 袁庆华 黄长亮 编著

北京师范大学出版社

(京)新登字 160 号

责任编辑 倪 花
封面设计 何 欣

图书在编目(CIP)数据

趣味 BASIC 程序/袁忠良等编著.—北京:师范大学出版社,1994.5
(计算机趣味程序设计丛书)
ISBN 7-303-03619-9
Ⅰ.趣… Ⅱ.袁… Ⅲ.BASIC 语言-程序设计语言-程序系统 Ⅳ.TP312BA

中国版本图书馆 CIP 数据核字(94)第 05218 号

趣味 BASIC 程序
袁忠良 袁庆华 黄长亮 编著

北京师范大学出版社出版发行 (邮编 100088)
北京通县达明印刷厂 全国新华书店经销
开本:787×1092 1/32 印张:7 字数:150 千字
1994 年 6 月第 1 版 1994 年 6 月第 1 次印刷
印数:1—10000

ISBN7-303-03619-9/TP·20 定价:6.00 元

前　　言

随着我国建设事业的发展，需要有一批高级的电脑技术人材，也需要大批初、中级的电脑人材。因此，及早地在青少年中进行电脑知识普及教育，开展多种形式的电脑科技活动，对于培养大量的电脑人材，加快我国现代化建设具有重大的战略意义。

青少年利用微电脑开展课外学习，解算一些趣味数学题，把乏味的数字计算趣味化，既有利于开阔思路，发展智力，也有助于激发学习科学文化知识的热情和积极性。信息时代的青少年应该借助电脑来“思考”和解算各种数学问题，并且编制简易程序，上机调试，初步掌握微电脑技术，打破对微电脑的神秘感，培养进一步学习电脑技术的浓厚兴趣，以适应信息时代对人材素质的要求。

鉴于以上想法，我们编著了这本具有较强的实用性和趣味性的微电脑普及读物，奉献给广大青少年读者。

全书共分八个部分，选编了 58 个趣味数学程序，拷在一张软盘上，读者需要可。购买该书的复合版。如只显示不打印请删去程序中所有 LPRINT 语句。

书中全部应用程序都在 IBM－PS/2 微型机上运行通过，也适用于 IBM－PC、IBM－PC/XT、IBM－PC/AT 及其兼容机。通过屏幕提示，使程序操作简单、使用方便、易于阅读。突出了汉字系统、趣味数学与使用操作。本书内容选自国内外趣味程序设计，数学竞赛题，许多题出自著名数学家之手，也有些题是作者亲自编写，在 BASIC 语言的基础上引入了汉字系统，

使屏幕画面更加鲜艳直观，开拓了应用范围。

书中题目趣味横生，引人入胜，使乏味的数字计算趣味化，寓教于乐，集趣味与知识于一体。选材有简有繁，实用性强，适用面广，对开发不同层次学生的智力、提高青少年的学习兴趣及培养学习电脑技术的浓厚兴趣，定会起到不可多得的效果。

本书可供青少年电脑爱好者和电脑技术初学者阅读，也可供大、中专、中学师生，小学教师及各地少年宫、科技站、青少年电脑活动中心的辅导员参考，又可作为学习数学的补充资料，视为启发智力的绝妙习题集。

本书由袁忠良、袁庆华、黄长亮合作编著，由袁忠良负责全书的组织和定稿。由于编著者水平有限，对书中疏漏和不妥之处，欢迎广大电脑爱好者和读者给予指正。

编著者

1994年4月

目　　录

一、数的趣谈

1. 数的趣谈

(一) 问题提出

任意给出一个十进制的自然数(如 7485),然后求这个数字每位数的平方和($7^2+4^2+8^2+5^2$),对得出来的数(154),再用此法进行计算($1^2+5^2+4^2=42$),仿照此法继续进行下去($4^2+2^2=20,2^2+0^2=4,4^2=16,1^2+6^2=37,3^2+7^2=58,5^2+8^2=89,8^2+9^2=145,...$)。

证明这个过程若不把原来数变化成 1 (因为 1 的平方还是 1,所以以后的数永远是 1),就必然变成 145,然后便出现 42、20、4、16、37、58、89 这样一个周而复始的循环数。

(二) 算法分析

设

$$L=10^{n-1}a_n+10^{n-2}a_{n-1}+...+10^2a_3+10a_2+a_1$$

表示任意一个 n 位自然数,用

$$L_1、L_2、L_3、... \quad ... \qquad (1)$$

表示出由 L 产生的每位数字平方和的数列,其中 L_1 是 L 的各位数的平方和,其余各项都是各自的前一项数字的每位的平方和

$$L_1=a_n^2+a_{n-1}^2+...+a_3^2+a_2^2+a_1^2$$

于是有

$$L-L_1=(10^{n-1}-a_n)a_n+(10^{n-2}-a_{n-1})a_{n-1}+...+(10^3-a_4)a_4+(10^2-a_3)a_3+(10^1-a_2)a_2+(a_1-1)a_1$$

因为 a_1, $a_2,a_3,...$,a_n 都是由 0 到 9 的自然数,所以不

难看出

$(a_1-1)a_1 \leqslant 72$

当 $n \geqslant 3$ ($a_n \neq 0$ 时)

$(10^{n-1}-a_n)a_n \geqslant 99$

同时 $(10^{i-1}-a_i)a_i \geqslant 0$,$i=2,3,...,n-1$

所以 $L > L_1$

由这个不等式可知，当数列(1)各项还是大于或等于三位数之前，后面的项恒小于前面的项，即数列这一部分是递减数列，所以任一个大于三位数的 L，在经过若干步各位数平方和以后必定都变化成一个不多于三位的数，因此本题的结论是否正确？只要研究一下三位数就够了。

因此，我们完全可以认为 L 就是一个三位的数，即 $n=3$，这时 $a_3 \neq 0$ 并得：

$L-L_1=(100-a_3)a_3+(10-a_2)a_2-(a_1-1)a_1 \geqslant 99-72=$
27

或 $L_1 \leqslant L-27$

由这个不定方程可知，数列(1)的某个项是一个两位数，设这个项是：$L_g=10j+k$。由于把 L_g 改为 $10k+j$ 数列 L_g 1、L_g 2、L_g 3、...的各项不变，所以用检查 L_g 在 $j \geqslant k \geqslant 0, j \geqslant 1$ 时的各数字平方和的方法，可以完全证明本题的结论。

当 $L_g=10j+k, j \geqslant k \geqslant 0, j \geqslant 1$ 时，L_g 1 必然是表二中某个数字。表一中各列各数是 j^2+k^2 的值。

我们可以把表中的 1,10,100 三个数以及题中指出的 145,42,20,4,16,37,58,89 八个数舍掉，因为这些数都是循环中规定的数，此外还可以舍掉：2,40,50,52,61,73,80,81,85,90,98,130 这些数，因为这些数和前边那些数或表一中的

另外一些数的区别只是数码的排列不同，或只多一个数码 0 而已。这样，用来检验这个结论是否正确的数就只剩下 28 个数了，它们是：

表一　　平方和表

j \ K	0	1	2	3	4	5	6	7	8	9
1	1	2								
2	4	5	8							
3	9	10	13	18						
4	16	17	20	25	32					
5	25	26	29	34	41	50				
6	36	37	40	45	52	61	72			
7	49	50	53	58	65	74	85	98		
8	64	65	68	73	80	89	100	113	128	
9	81	82	85	90	97	106	117	130	145	162

5,8,9,13,17,18,25,26,29,32,34,36,41,45,49,53,64,65,68,72,74,82,97,106,113,117,128,162。

表二　　检验数表

检验数	数列(1)的项	检验数	数列(1)的项
5	25、29、85	72	53、34、25
8	64、52	74	65
9	81	82	68、100
18	65、61	106	37
32	13、10	113	11、2
36	45、41、17、50	128	69、117、51、26、40
49	97、130	162	41

我们把检验的结果放入表二中，表的头一栏所列的是应该检验的数，第二栏内依次写的是由这个数产生的数列(1)的几个项，填写时，要边写边查，一直写到出现了能肯定这个结论是

否正确的数为止。

由于用来检验的各数，最终都能变化成为 1 或 145，42，20，4，16，37，58，89 当中的一个数，而这些数又都周期地重复，因此本题所说的结论也就得到了证明。

（三）操作步骤及运行结果

开机

进入 CCDOS 系统

进入 BASIC 系统

将数的趣谈程序“QWSX1”调入内存，在 OK 提示符状态下键入命令：

RUN ↙

屏幕显示：

请输入自然数 N=？2583 ↙

N=2538——→145 共需步数=7

请输入自然数 N=？1324 ↙

N=1324——→145 共需步数=9

请输入自然数 N=？7695 ↙

N=7695——→145 共需步数=6

请输入自然数 N=？53764 ↙

N=53764——→145 共需步数=8

请输入自然数 N=？524368 ↙

OK

注：屏幕显示中，有下横线的自然数和回车为键盘输入操作，其它均为屏幕显示。（以下类同）。

在操作中，键入的自然数，按上述法则运算的结果非 145（定为 1）的，退回 BASIC 系统，例如：

OK

RUN ↙

请输入自然数 N=？32 ↙

OK

RUN ↙

请输入自然数 N=？1 ↙

OK

SYSTEM ↙（退出 BASIC 系统）

关机。

(四) 程序清单

```
10 REM "数的趣谈 QWSX1"
30 INPUT "请输入自然数 N=";N
40 J=0
50 IF N<=1 GOTO 210
60 M=0
70 L=N
80 FOR I=1 TO 5
90 K=L-INT(L/10)*10
100 M=M+K*K
110 L=(L-K)/10
120 IF L=0 GOTO 140
130 NEXT I
140 IF M=1 GOTO 210
145 J=J+1
150 L=M
160 M=0
170 IF L<>145 GOTO 80
180 PRINT "N=";N;"---->145 共需步数=";J
190 IF L<>145 GOTO 210
```

```
200 GOTO 30
210 END
```

2. 素数的求法

（一）问题提出

在自然数中，可分为两类数，一类的自然数，它的约数除了 1 和它本身外，尚有其它约数，如 4，6，8，9，10，12，... 这一类数叫做合数。另一类的自然数，它的约数除了 1 和它本身外，没有其它约数，如 2，3，5，7，11，... 这类数叫做素数（或质数）。如何求自然数中的素数？例如求自然数 1 至 500 间的所有素数。

（二）算法分析

先分析自然数的约数。如 11 的约数只有 1 和 11 本身，即只有两个约数，而 12 的约数除 1 和 12 外，尚有 2，3，4，6，共有 6 个约数。

1 是一个特殊的自然数，它只有一个约数，即它本身。

由以上分析，可知素数只有两个约数，合数多于两个约数。

除了 2 外，其它所有偶数都是合数；同时除了 2 外其它所有的素数都是奇数。

如果我们先把 2 除掉，从 3 开始至 500 间的所有素数，可用筛选法算出为：

3，5，7，11，13，17，19，... ，89，97，101，103 。

根据筛选法定理：若求 N 以内的所有自然数中的素数，可用 N 的平方根以内的素数倍数逐个筛掉，余下的便是所求的素数。

（三）操作步骤及运行结果

开机

进入 CCDOS 系统

进入 BASIC 系统

将素数的求法程序"QWSX2"调入内存，在 OK 提示符状态下键入命令：

RUN ↙

屏幕显示：

请输入自然数 N=? 500 ↙

从 1 至 500 间的所有素数

========================

3	5	7	11	13	17
19	23	29	31	37	41
43	47	53	59	61	67
71	73	79	83	89	97
101	103	107	109	113	127
131	137	139	149	151	157
163	167	173	179	181	191
193	197	199	211	223	227
229	233	239	241	251	257
263	269	271	277	281	283
293	307	311	313	317	331
337	347	349	353	359	367
373	379	383	389	397	401
409	419	421	431	433	439
443	449	457	461	463	467
479	487	491	499		

OK

SYSTEM ↙（退出 BASIC 系统）

关机。

（四）程序清单

```
1 REM "素数的求法 QWSX2"
2 INPUT "请输入自然数 N=";N
3 LPRINT "请输入自然数 N=";N
4 LPRINT
5 PRINT "            从 1 至";N;"间的所有素数"
6 LPRINT "            从 1 至";N;"间的所有素数"
10 PRINT "=============================="
12 LPRINT "=============================="
15 FOR I=3 TO N STEP 2
20 LET J=INT(SQR(I))
25 FOR K=3 TO J
30 L=I-INT(I/K)*K
35 IF L=0 GOTO 50
40 NEXT K
45 PRINT I,
48 LPRINT I,
50 NEXT I
52 PRINT
54 LPRINT
60 END
```

3. 哥德巴赫猜想

（一）问题提出

十七世纪两位著名数学家哥德巴赫和欧拉是多年的老朋友。1742 年德国数学家哥德巴赫给当时住在德国的数学家欧拉的一封信中，提出了一个命题。哥德巴赫在信中写到：我的问题如此，随便取某一个奇数，比如 77，它可以写成三个素数

之和：77＝53＋17＋7，再任取一个奇数461，那么461＝449＋7＋5，也是三个素数之和。461还可以写成257＋199＋5，仍为三个素数之和，这样，我发现：

任何大于5的奇数都是三个素数之和。

但又怎样证明呢？虽然任何一次试验都可得到上述结果，但不可能把所有奇数都拿出来检验，需要的是一般的证明，而不是个别的检验。

德国数学家欧拉在回信中说，这个命题看来是正确的，但他也给不出严格的证明。同时欧拉又提出了另一个命题：任何一个大于2的偶数都是两个素数之和。

但这个命题欧拉也没有能够给予证明。

请你验证这两位数学家在书信中提出的问题。

（二）算法分析

容易证明哥德巴赫的猜想是欧拉命题的推论。事实上，任何一个大于5的奇数都可以写成如下形式：$2N+1=3+2(N-1)$，其中$2(N-1)\geqslant 4$。若欧拉命题是正确的，则偶数$2(N-1)$可以写成二个素数之和。故对于大于5的奇数，哥德巴赫命题是正确的。

但若哥德巴赫的命题是正确的，并不能保证欧拉命题的正确，故欧拉的命题是基本的。现在通常把这两个命题统称为哥德巴赫猜想。

在过去的二百多年中，尽管许多数学家为解决这个猜想付出了艰苦的劳动，但迄今为止它们仍是一个没有被证明也没有被推翻的定理。

他们的猜想是合理的，例如哥德巴赫所提出的任何大于5的奇数总可分为三个素数之和：