

# 统一的现代数学

第四册第一分册

美国中学数学课程改革研究组编

.634

1

人民教育出版社

# 统一的现代数学

第四册第一分册

曹才翰 译

吴望名 校

人民教育出版社

1977. 北京

## 内 容 提 要

《统一的现代数学》是美国中学数学课程改革研究组编的一套中学数学现代化课本，全书共六册十二分册，内容除有一定的初等数学外，还包括集论、数理逻辑、近世代数、微积分、概率论、程序设计、线性规划等基本知识，并用现代数学的结构思想作了统一处理。

本书系内部参考资料，供研究外国中学数学教材用。

这套课本对我们了解国外中学数学现代化的动态，研究用现代数学观点处理中学数学教材有一定参考价值。但对这套课本内容中渗透的资产阶级思想意识，应当注意分析批判。

本分册是按该书第四册第一分册 1971 年版译出的，包括用 BASIC 语言作程序设计，二次方程和复数，圆函数，条件概率与随机变量等四章。

### 统一的现代数学

第四册第一分册

美国中学数学课程改革研究组

曹才翰 译

吴望名 校

\*

人民教育出版社出版

新华书店北京发行所发行

人民教育出版社印刷厂印装

\*

1977 年 11 月第 1 版 1978 年 8 月第 1 次印刷

书号 13012·051 定价 0.45 元

# 目 录

<b>第一章 用 BASIC 语言作程序设计</b> .....	1
1.0 引 言.....	1
1.1 PRINT, END, RUN 及 BASIC 四则.....	3
1.3 SCR, REM, LET, 修正.....	9
1.5 READ, DATA, GO TO 和 PRINT.....	12
1.7 IF... THEN.....	18
1.9 FOR, NEXT, STEP.....	24
1.11 附标变量, DIM.....	34
1.13 某些特殊函数和它们的应用.....	44
1.15 小 结.....	53
<b>第二章 二次方程和复数</b> .....	57
2.0 引 言.....	57
2.1 返回去看——关于一个变量的方程.....	57
2.3 二次公式.....	63
2.5 方程和计算机.....	69
2.7 到 $2 \times 2$ 矩阵中去旅行.....	77
2.9 复数系.....	86
2.11 复数平面.....	90
2.13 复数和变换.....	94
2.15 小 结.....	99
<b>第三章 圆函数</b> .....	103
3.1 圆函数——在进行之前复习一下.....	103
3.3 包卷函数.....	108
3.5 实数的圆函数.....	114
3.7 正弦和余弦的性质.....	118

3.9	圆函数和复数	125
3.11	圆函数、复数和变换	131
3.13	棣美弗定理	135
3.15	圆函数的图象	140
3.17	某些新的圆函数	147
3.19	三角方程	154
3.21	小结	156
<b>第四章</b>	<b>概率 条件概率和随机变量</b>	<b>160</b>
4.1	引言	160
4.3	条件概率	164
4.5	进一步讨论条件概率	177
4.7	独立事件	184
4.9	随机变量	191
4.11	数学期望	197
4.13	小结	202

# 第一章 用 BASIC 语言作程序设计

## 1.0 引言: 程序设计和 BASIC<sup>①</sup>

让我们用关于凯恩的简短叙述来开始这一章的计算机程序设计。假设凯恩是中学数学教材改革研究班的学生, 他将从事用 BASIC 语言作程序设计的学习。

凯恩宣称在他的数学课中他要学习计算机程序, 就已经向他提出在各种问题的求解中, 请提供一些“专门信息”。

I. 凯恩的小妹妹对质数的规律很感兴趣, 她的研究受到限制, 因为她不能找到一个大的质数表(譬如说含前 200 个质数), 而要动手列出这样一个质数表, 将花费大量的时间。于是, 她就去问凯恩, 计算机如何输出一个含前 200 个质数的质数表。

II. 凯恩的父亲是新英格兰地区一个小公司的销售经理, 每个月他必须就周销售量、月销售总量以及 20 个销售员薪金和调配工作提出一份报告, 他就去问凯恩, 要计算机来做这种日常的和乏味的工作。

III. 凯恩的哥哥最喜欢的数学课题是画函数的图象。近来, 他已经在做象  $x \xrightarrow{f} x^6 - 2x^3 + 1$  和  $x \xrightarrow{g} \frac{2x}{3x+1}$  这样函数的图象, 而他发现, 对这样复杂的函数, 画出其精确图象不是一件容易的事。他就去问凯恩, 要计算机在他画这些函数图象时帮助他作必要的计算。

IV. 凯恩的一个朋友近来是环境保护俱乐部的一个成员, 他

---

<sup>①</sup> BASIC 是 Boginneis All-purpose Symbolic Instruction Code (初学者通用符号指令码)的缩写, 是一种简单易学的计算机语言——译校者

的朋友在他的镇上指导空气污染研究。每天他在镇上 5 个地方取空气污染的数据, 在每 30 天周期的末尾, 他计划报告他的结果, 他就去问凯恩, 计算机能否可用来帮助他处理汇集的数据。

在上面四个问题中的每一个问题, 各人知道如何去解各自的问题。凯恩的小妹妹知道如何去检验一个数是不是质数, 但要去找前 200 个质数就不大好办。使用计算机去解这个问题, 包括告诉计算机要做什么 (看图 1.1) 以及去做这个工作。

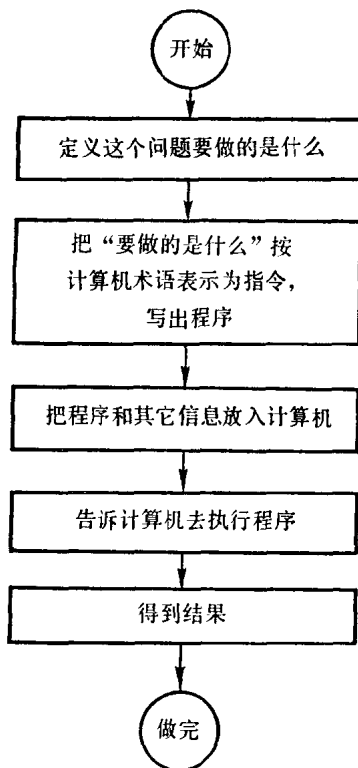


图 1.1

这一章的主要任务将集中在这个过程的核心——写出一个传达给计算机的、我们希望它去做什么的程序。传送给计算机的方

式将使用叫做 *BASIC* 的计算机语言。

今天,有许多计算机程序设计语言——例如, *APL*, *BASIC*, *FORTRAN*, *PL/I*, *ALGOL*, *COBOL*, 等等。为满足不同领域的需要,创造了这些语言。*BASIC* 语言相对地说来,指令较少而且很容易学习,然而在能行的范围内它是有力的,而且是多功能的。*BASIC* 通常可用于许多计算机系统(包括你学校里的)。在以后的几章中,将使用 *BASIC* 语言作程序设计以及用计算机来协助我们解题。本章只是表明用计算机工作的开始。

### 1.1 *PRINT*, *END*, *RUN* 及 *BASIC* 四则

假如我们要计算三个试验数据 97、88 和 95 的平均数,对于这个问题的 *BASIC* 程序给出如下:

输入

```
10 PRINT (97 + 88 + 95) / 3
```

```
20 END
```

```
RUN
```

输出

```
93.33333
```

输入的头两行(行数 10 和 20)都是指令<sup>①</sup>,而 *RUN* 是给计算机的命令,命令它去执行或实施这组指令。在 *BASIC* 程序中每个指令必须有一个数(从 1 到 9999)放在前头,这些指令数指导计算机执行程序。*END* 语句在程序中必须有最大的指令数,任何一组正整数均可被用来作为指令数,只要这组数的顺序符合计算机执行指令的顺序。

下面是另一个程序,它计算半径为  $2\frac{3}{4}$  吋的圆周长和圆面积

输入

---

① *PRINT*—打印、印刷, *END*—终止——译校者



```
10 PRINT 2*3.1416*2.75, 3.1416*2.75↑2
```

```
99 END
```

```
RUN
```

### 输出

```
17.2788
```

```
23.7584
```

计算机能采纳的唯一数字是小数，我们不能用  $2\frac{3}{4}$ ，我们能  
用 2.75，或用  $2+3/4$  来代替  $2\frac{3}{4}$ 。

在 10 这一行中，第一个表达式是表示圆周长的积：

$$2 \times 3.1416 \times 2.75$$

这个积就是在输出中给出的第一个结果。注意关于乘法的符号是星号“\*”，它对指令乘法来说是唯一的表示方式。

在第一个表达式后用逗号分开的第二个表达式是表示面积的积  $3.1416 \times 2.75^2$ ，这个积就是在输出中给出的第二个结果。在这里，我们看到使用了一个新的符号：向上的箭头↑，它可以翻译为“上升成幂”。在我们的问题中， $2.75\uparrow 2$  意即  $2.75^2$  或  $2.75 \times 2.75$ 。

第二个表达式揭示了计算机的另一个重要性质。关于执行计算的顺序你可能会感到惊奇，你可能认为计算机是在做  $3.1416 \times 2.75$  的平方，但并不是这样，而计算机所计算的如同被表示为

$$3.1416 * (2.75 \uparrow 2)$$

的这样的表达式。假如我们想要积的平方，那么就要用如下的括号

$$(3.1416 * 2.75) \uparrow 2$$

关于“乘法”和“上升成幂”的特别记法以及关于括号的规则，都是计算机语言 BASIC 结构的一部分。

和普通的语言一样，BASIC 语言也有词汇、文法和标点系统。为了使用 BASIC，你就要去学习这些东西，它们虽然是简单的，但是精确的。计算机不能为你作出什么判断，它只能精确地履行你用

*BASIC* 语言表达的意图。因此，关键在于在 *BASIC* 中的指令要非常仔细地准备和检查。你要在纸上准备好指令，使计算机运转时间不至于白白地浪费掉。

这里是 *BASIC* 的某些初步的习惯用法，其它的将在以后给出。

1. 一的符号“1”不是小写的 *L*，它有它自己的键。

2. 零的符号“0”不是 *N* 以后的那个大写字母<sup>①</sup>，零有它自己的键。

3. 圆点充当小数点。

4. 在数字中不能用逗号，二百万可记作“2000000”或记作“2 000 000”。最末尾用或不用小数点。这里空当处仅仅是为了便于检验，其本身没有意义。事实上，除了少数不重要的场合以外，计算机对出现的空当是无反应的。*P R I N T* 对计算机来说其意义和 *PRINT* 一样。

5. 最常用的运算符号如下：

+ （表示加法的加号）

- （表示减法的连字符）

\* （表示乘法的星号）

/ （表示除法的斜杠号）

↑ （表示上升成幂的向上箭头号）

6. 计算机有使用运算顺序的规定，但是，假如有疑问的话，那么总可使用作为预防错误的括号，于是

$$2*(A\uparrow 2) \text{ 可代替 } 2*A\uparrow 2$$

$$3-(2/7) \text{ 可代替 } 3-2/7$$

7. 运算顺序是：

---

① 为了区分零“0”与 *N* 以后的那个英文字母 *O*，有时也用  $\phi$  来表示英文字母 *O*——译校者

(a) 先做括号内的运算.

(b) 其次做上升成幂( $\uparrow$ ).

(c) 再做加性的逆元(即取负号), 如 $-5\uparrow 3$ 意即 $-(5\uparrow 3)$ .

(d) 再做 $*$ 或 $/$ , 假如这两者同时出现, 那么先做左边指明的运算. 如

$$8*3/2 \text{ 意即 } (8*3)/2$$

$$8/3*2 \text{ 意即 } (8/3)*2$$

(e)  $+$ 或 $-$ 最后做. 假如它们同时出现, 那么先做左边的运算.

$$8+4-2 \text{ 意即 } (8+4)-2$$

8. 在 *BASIC* 中输出通常是用 7 个(有时 6 个, 8 个或 12 个这要取决于你的计算机系统)数字的小数形式表达. 解释下述的输入和输出表达式:

输入	输出	输入	输出
3/4	.75	4*100000	400000
2/3	.6666667	40*100000	4.000000E+6
14/3	4.666667	9000 $\uparrow$ 2	8.100000E+7

注意 2/3 是由舍入的 7 个数字来表达. 数 1,000,000 或更大的数, 输出用指数(科学记数)形式.

4.000000E+6 意即

$$4.000000 \times 10^6 = 4.000000 \times 1000000 = 4000000$$

8.100000E+7 意即

$$8.100000 \times 10^7 = 8.100000 \times 10000000 = 81000000$$

注意所有的指令必须在一条线上相继给出.

为了计算  $\frac{3+8 \times 9}{6^3-5}$ , 我们能写为  $(3+8*9)/(6\uparrow 3-5)$ .

注意到, 我们也能按照一组带有括号的符号来做. 如象这样的表达式

$$\{[(3+8) \times 9]^2 - 5\}^3$$

能够写为

$$(((3+8)*9)\uparrow 2-5)\uparrow 3$$

计算机是连同括号一起翻译的。

下面是一些使用 *BASIC* 的例子:

传统的记法

*BASIC*

(1)  $3x^2+x-6$

(1)  $3*x\uparrow 2+x-6$

(2)  $\frac{5}{2x}$

(2)  $5/(2*x)$

(3)  $[(A+B)^2-C^2]^3$

(3)  $((A+B)\uparrow 2-C\uparrow 2)\uparrow 3$

(4)  $1 + \frac{1}{1 + \frac{2}{1 + \frac{3}{1+4}}}$

(4)  $1/(1+(2/(1+(3/(1+4))))))$   
或  $1/(1+2/(1+3/(1+4)))$

## 1.2 练 习

1. 对于下述的 *BASIC* 表达式写出等价的传统表达式:

(a)  $-5*3\uparrow 2$  (b)  $-3\uparrow 2$  (c)  $2+3/4$  (d)  $2*3/4$

(e)  $2*3/4/5$  (f)  $((1+2)\uparrow 3+4)\uparrow 5+6)\uparrow 7$

(g)  $A+B/C/D+A/B\uparrow 3$  (h)  $N\uparrow (1/N)$

2. 对于下述各传统的表达式写出等价的 *BASIC* 表达式:

(a)  $3.14 \cdot R^2$  (b)  $\frac{1-x^2}{1+x^2}$  (c)  $\frac{1-X}{1-\frac{Y}{1-Z}}$

(d)  $\frac{4\pi R^3}{3}$  (e)  $A^2+B^2-2AB$  (f)  $\frac{N(N-1)(N-2)}{1 \times 2 \times 3}$

(g)  $\{[(1-a)^2-b]^2-c\}^3$

3. 修正下述各 *BASIC* 表达式:

(a) 2,000 (b)  $A \div L$  (c)  $2(L+W)$  (d)  $\frac{1}{2}BH$

(e)  $2\frac{1}{2} * 4\frac{1}{3}$  (f)  $2 * 3$  (g)  $(7+8)/2$

4. 写出下述各式的 *BASIC* 程序并且执行它:

$$(a) (92+93+94+95)/4$$

$$(b) 986/3$$

$$(c) 3.1416 \times 5.5^2$$

$$(d) \frac{4}{3}(3.1416)(5.5)^3$$

$$(e) \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6}$$

$$(f) \frac{1}{2} + \frac{1}{2 \times 3} + \frac{1}{2 \times 3 \times 4} + \frac{1}{2 \times 3 \times 4 \times 5}$$

$$(g) \left(1 + \frac{1}{2}\right)^2, \quad \left(1 + \frac{1}{2}\right)^3$$

$$\left(1 + \frac{1}{3}\right)^3, \quad \left(1 + \frac{1}{3}\right)^4$$

$$\left(1 + \frac{1}{4}\right)^4, \quad \left(1 + \frac{1}{4}\right)^5$$

$$\left(1 + \frac{1}{5}\right)^5, \quad \left(1 + \frac{1}{5}\right)^6$$

5. 下述程序错误在哪里?

(a) 10 PRNT 3.1416 \* 2.71828<sup>2</sup>

20 END

30 RUN

(b) 10 PRINT 6(5,280+333)

9 END

RUN

问题 6—8 是关于计算机按不同形式输出的数，并且误差在计算机的限度内。

6. 在 BASIC 中，数 1,000,000 或更大的数是按指数形式打印出。

例如 4,200,000 用 4.20000E+6 打印出。而 4.20000E+6 的意思是

$4.20000 \times 10^6 = 4,200,000$ 。将下述的每一个按指数形式表达：

(a) 5,000,000

(b) 4,500,000

(c) 8,321,000,000

(d) 8,765,400,000

7. 下述各值是从 BASIC 程序中输出的。将这些输出表达为整数形式：

(a) 4.20000E+8

(b) 1.23400E+9

(c) 2.02022E+12

8. 在 BASIC 中，0 和 1 间的数能表达为指数形式，研究下面四个例子：

.0123456

1.23456E-2

.00000123456

1.23456E-6

.0000001

1.00000E-7

.000000000000023

2.30000E-13

注意  $10^2=100$  而  $10^{-2}=\frac{1}{10^2}=\frac{1}{100}=.01$

所以  $1.23456 \times 10^{-2} = 1.23456 \times .01 = .0123456$

(a) 将下述各值表达为指数形式:

(I) .0000003    (II) .00001278    (III) .0000000008

(IV) .00012321

(b) 将下述各值写为小数形式:

(I)  $1.2E-3$     (II)  $1.10000E-2$     (III)  $.654321E-6$

(IV)  $4.54545E-4$

(c) 一般地, 对于  $n > 0$  且  $a \neq 0$ , 我们定义  $a^{-n} = \frac{1}{a^n}$ .

求下述的值:

(I)  $3^{-2}$     (II)  $(-3)^{-3}$     (III)  $(10^{-2})^{-2}$     (IV)  $(-1)^{-10}$

### 1.3 SCR, REM, LET, 修正

早先我们曾计算过的具有半径  $2\frac{3}{4}$  吋的圆周长和圆面积的问题能编成下述程序:

SCR

5 REM CIR—AREA OF CIRCLE

10 LET R=2.75

20 LET P=3.1416

30 PRINT 2\*P\*R, P\*R↑2

40 END

SCR 代表 SCRATCH (清除), 是用在已经执行了一个程序之后紧接一个程序的开头处。它的功能是抹去早先的程序。SCR 也可用在程序的输出之后的结束处, 为计算机预备下一个程序。和 RUN 一样, 在 SCR 所在行的开始处没有数字。在每一个程序开始时, 用 SCR 是从事程序设计的好的习惯。

REM 代表 REMARK (注解, 说明), 是关于程序目的或在程

序中的特别语句提供说明的。在一个程序中，特别是在长而复杂的程序中，我们能插入“*REM*”语句去适当解释这个程序要做的是何工作，以及解释要去做的那个程序的每段的目的是什么。“5 *REM CIRC—AREA OF CIRCLE*”说明了这个程序是关于求圆的周长和面积。*REM*语句必须要有一个指令数。

*LET* (置, 令)指令用一个变量开始, 这个变量是这个程序中的单独字母, 并且它的指定值在等号的右边给出。假定变量出现在等号的右边, 那么它们必须早已在这个程序中指定了值。

变量能通过两种形式表达: 或者用一个单独的字母, 或者用一个单独的字母紧随着一个 0 到 9 的数。例如 *A, X, A1, A3, A9* 等可以在 *BASIC* 程序中作为变量使用。但 *AP, A12, X2T, 7W* 不能使用。

在一个程序的打字过程中, 人们可能会发生错误, 我们将给出下面两种方法来改正错误。

### 1. 使用反箭头“←”

反箭头每次抹去(前面的)一个符号或位置。反箭头被用来修改语句中的错误。例如,

<i>PRNT←←INT</i>	<i>PRINT</i>
<i>LW←*W</i>	<i>L*W</i>
<i>7+5←←-5</i>	<i>7-5</i>

### 2. 整个一行重打

假设在上面的过程中, 你想要半径  $5\frac{3}{4}$  吋而不是  $2\frac{3}{4}$  吋。假如你在打出“*RUN*”之前的任何时候发现了这个错误, 而在继续打出这个程序的其余部分前你必须做的是打出

10 *LET R=5.75*

现在就可看作下述程序中的随便哪一个, 可是仍然得到所要

求的结果.

10	LET	R=2.75	10	LET	R=2.75
20	LET	P=3.1416	20	LT←ET	P=3.1416
30	PRINT		30	PRINT	
		2*P*R, P*R↑2			2*P*R, P*P↑2←←←R↑2
99	END		99	EDN	
10	LET	R=5.75	99	END	
			10	LET	R=5.75

如我们在 *RUN* 和输出之后发现了错误,那么我们就需要打出

```
10 LET R=5.75
RUN
```

并且下一个输出恰好就是我们所希望的结果.

## 1.4 练习

1. 使用反箭头改正下述各式:

(a) 6.28RH      (b) (A+B)/2      (c) LET p=3.14.16  
(d) ENDD      (e) 2,000

2. 下述各程序的错误是什么?

(a) 10	LET	L=17	(b) 10	LET	L=M
20	LET	A=L*W	20	LET	12=W
30	PRINT	A	30	LET	A=L*W
END			40	PRINT	A
RUN			29	END	
				RUN	

3. 求没有反箭头的等价表达式:

(a) 3.1426←←16      (b) REED←←AM←D  
(c) ABSO←L←      (d) G\*T↑3/2←←←2/5←2  
(e) PRNT←←INT      (f) LEET←←T

4. 下述各式哪一个是 *BASIC* 关于变量的正确表达式:



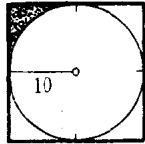
- (a) A1    (b) A2    (c) AVE    (d) 3A    (e) 5WIN  
 (f) XI23    (g) 007    (h) X0    (i) X10

5. 求下述 BASIC 程序的输出:

```
(a) 10 LET X=4
    20 LET Y=(X↑2)*X
    30 LET Z=Y-X
    40 PRINT Z←←←NT Z
    50 END

(b) 100 LET=A=2
    150 PRINT A
    200 LET A=A+A
    250 PRINT A
    300 LET A=A*A
    350 PRINT A
    400 LET A=A↑2
    450 PRINT A
    500 END
```

6. 使用重写一行的技术去求具有半径  $3''$ 、 $3\frac{1}{4}''$ 、 $3\frac{1}{2}''$ 、 $3\frac{3}{4}''$  的圆面积和周长。



(第7题)

- \*7. 半径为 10 的一个圆内切于一个正方形, 写出一个 BASIC 程序去计算阴影部分的面积。

## 1.5 READ, DATA, GO TO 和 PRINT

这一节是讨论在程序中, 所使用的数据的输入和输出。在计算 97、88 和 95 的平均数问题的过程中, 能使用指令 *READ*(读识)、*DATA*(数据) 而得到不同的程序。

输入

```
5 REM COMPUTE AVERAGE OF 3
  NUMBERS(计算三个数的平均数)①
10 READ A,B,C
20 LET V=(A+B+C)/3
30 PRINT V
```

① 程序部分 *REM* 语句中的英语词组, 为使读者便于理解, 附加了汉语翻译, 但汉语翻译不是程序的一部分, 因而不必(也不可能)输入到计算机中——译校者