



电脑随书附送



BASIC

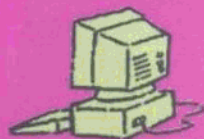


编著 叶彦 孙莉 绘画 荣非



49

人民邮电出版社



BASIC

现在让我们来
学习BASIC语言!



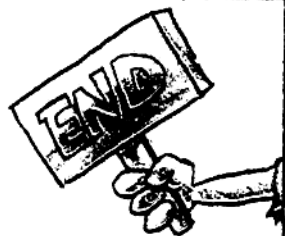
BASIC 语言是由一些很简单的英文语句构成的。由这些语句组成一串串的指令,让计算机去做各种各样的事,这就叫作程序。程序是人和计算机交谈的一种方式。

【例】 10 LET A = 3
20 LET B = 5
30 LET C = A + B
40 PRINT A, B, C
50 END



这是一个非常简单的 BASIC 程序。由这个程序我们可以看出:

1. 一个 BASIC 语言程序由若干行构成,一般一行一个语句,一行称为一个程序行。
2. 每个语句都有一个行号(即 10, 20, 30, 40, 50),行号必须是正整数(不能是小数、分数或负数)。
3. 行号的大小顺序是你所要求的工作次序。



4. 一个程序需包括两个或两个以上的语句。
5. 一个程序行一般由程序行号、语句定义符、语句体、语句间分隔符组成。
6. 每个程序必须有一条 END(结束)语句。

嘿!你这个
冒牌货!





组成 BASIC 语言的基本字符主要有三

种

1. 英文字母: A~Z, a~z, 大小写各 26 个

2. 数字: 即阿拉伯数字(0~9)10 个。

3. 符号(21 个)

. 小数点 * 乘号

, 逗号 / 除号

" 双引号 ^ (或 ▲) 乘方号

: 分号

(左括号

) 右括号

' 右单引号

' 左单引号

+ 加号

- 减号

< 小于号

> 大于号

= 等号

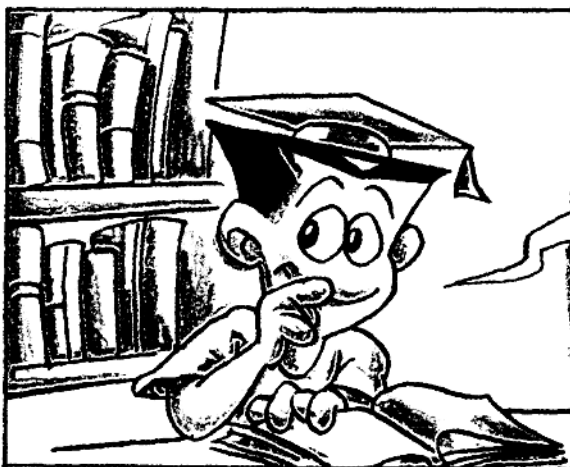
% 百分号

! 感叹号

井号

\$ 美元号





BASIC 语言中所用的常量分为数值常量和字符串常量。例如, 3.14159, -0.218, 9.42 等等, 均是数值常量, 而 "PEN", "286/9", "学生成绩" 等均是字符串常量。

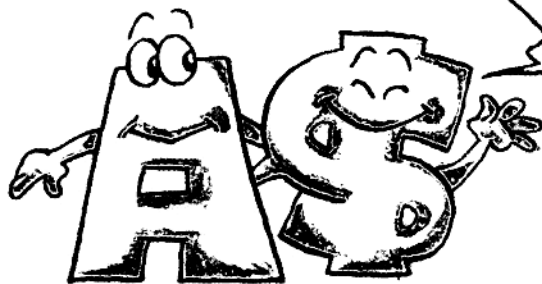
在 BASIC 语言中对应于常数和字符串常量还存在一种允许数值、字符串可以改变的量, 分别称为“数值变量”和“字符串变量”。数值变量和字符串变量又可以分为简单变量和下标变量两种。数值变量的值是数, 字符串变量的值是字符串常量。简单变量可以用英文字母中任一个字母表示, 也可以用字母后跟 0~9 中任一个数字来表示。如 A, F, W5, F7 等都是简单变量。

我代表
简单变量。



字符串变量通常用简单变量后跟一个串变量符号 \$ 来表示。例如: A\$, Z2\$ 等。

我们俩在一起就是
字符串变量。

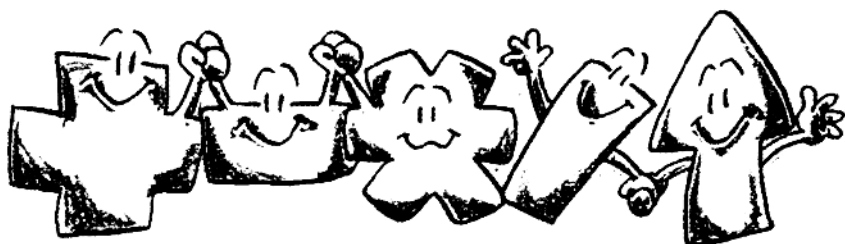


BASIC 语言的函数可以分成 3 类, 第一类是数值函数, 它可以是数学上常用的函数, 也可以是由用户自己定义的函数; 第二类是字符串函数; 第三类是其它函数。



由数、变量和函数组成，用 +、-、*、/、^ 五种运算符以及圆括号连接起来的式子，称为 BASIC 语言中的表达式。BASIC 语言的表达式包括算术表达式、字符串表达式和逻辑表达式 3 种，其中算术表达式最常用，它与人们数学上的算术式子基本相似。

【例】 $25 + A - 3 * B$
 $3.1416 * R ^ 2$



表达式运算的结果是一个数。运算的顺序规定为：一函数，二乘方，三乘除，四加减，有括号时括号内优先，同一级由左向右计算。

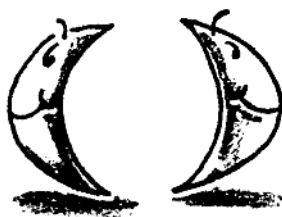
【例】代数式 $\frac{a+b}{c+d}$ 用 BASIC 表达式表示为：
 $(A + B) / (C + D)$



注意：

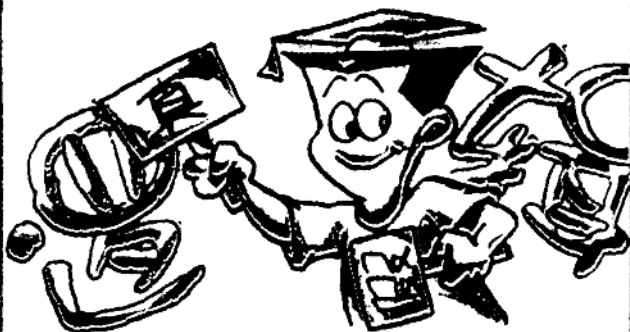
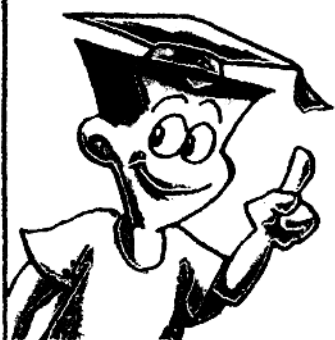
1. BASIC 规定两个运算符不能紧挨着。如 $\frac{A}{(-B)}$ 写成 $A / -B$ 是错误的，应写成 $A / (-B)$ 。

2. 在 BASIC 表达式中，每一个字符都占一格，所有字符必须一个一个并排写在同一横线上。

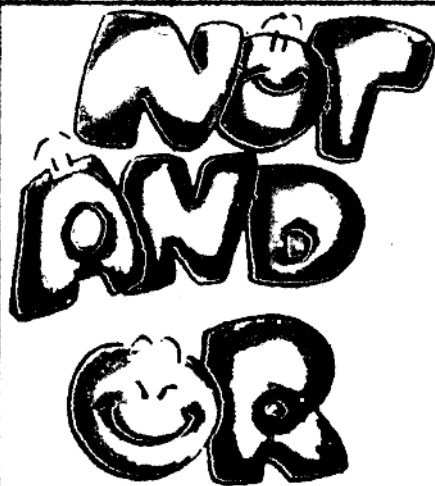


3. 表达式的括号可以嵌套，但一律用圆括号 ()，而不能使用中括号 [] 和大括号 { } 表达。

逻辑表达式也是表
达式中的一种。



它是用逻辑运算符和圆括号将关系式连接起来
的有意义的式子,逻辑表达式的值是真或假。



BASIC 语言的逻辑运算符有 NOT、AND、OR 三种。其
中 AND 和 OR 是一个运算符作用于两个运算对象,而
NOT 只作用于一个运算对象,并且该运算对象放在 NOT
运算符的后面。运算对象指关系式或逻辑表达式。

【例】 NOT(A > B)

$X - 5 > Y - 8$ AND $M = 17$

$I < = K + N$ OR $L > 84$

计算机在处理逻辑表达式时,先求出关系式的值,然
后再根据逻辑运算符的优先顺序进行逻辑运算。

NOT 运算 当运算对象 X 为真时,则 NOT X 的值为
假,若 X 值为假,则 NOT X 为真,NOT 运算称为非运算。

【例】 X = 8 时,NOT X = 8 的值为假

NOT X < > 8 的值为真。

AND 运算 当运算符两边的运算对象的值均为真时,则 AND 的运算结果为真,否则取假值。

【例】 若 X = 10, Y = 20, 则

$X > 0$ AND $Y > 0$ 取真值

$X > 0$ AND $Y < 0$ 取假值





OR 运算 当 OR 运算符两边的运算对象中有一个值为真,则经 OR 运算的结果为真,只有当两个运算对象的值均假时,OR 运算的结果才为假。

例如: $A=8, B=10$

$A > 0 \text{ OR } B < 0$ 取真值

$A < 0 \text{ OR } B > 0$ 取真值

$A > 0 \text{ OR } B > 0$ 取真值

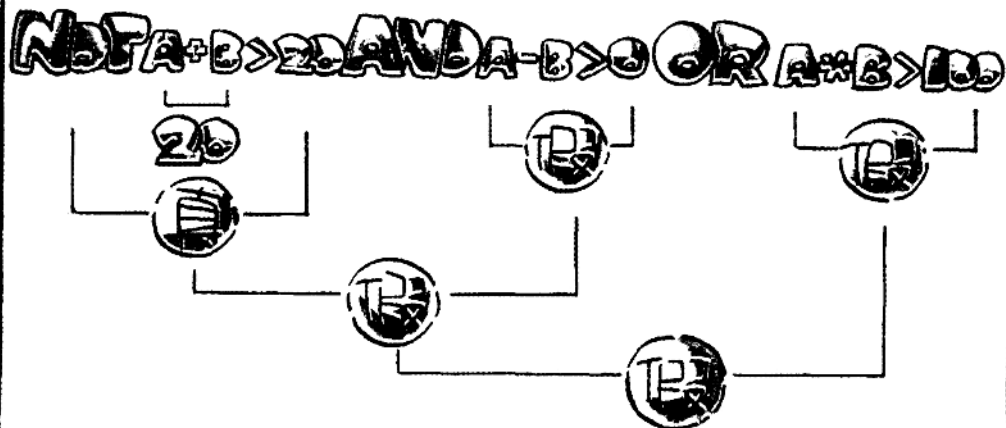
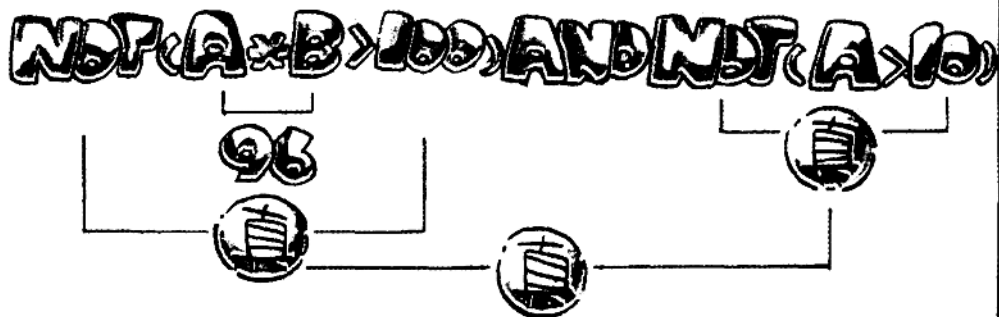
$A < 0 \text{ OR } B < 0$ 取假值

逻辑表达式运算的优先顺序是:

算术式中运算 $\rightarrow$ 关系运算 $\rightarrow$ NOT $\rightarrow$ AND $\rightarrow$ OR

即首先进行算术表达式或字符串表达式的运算, 然后进行关系运算, 最后进行逻辑运算, 而在进行逻辑运算时, 先进行 NOT 运算, 再进行 AND 运算, 最后进行 OR 运算。对于括号应先计算括号内, 再计算括号外, 多层括号嵌套时, 内层括号优先。

例如: $A=8, B=12$ 则





逻辑判断语句是根据逻辑判断式的值来判断“真”与“假”的。逻辑判断式的最终值为1,即条件为真,为0,即条件为假。

为了便于大家进行逻辑运算,现为人家列出三种逻辑运算的真值表。

A AND B 的真值表

A	B	A AND B
1	1	1
0	0	0
1	0	0
0	1	0

A OR B 的真值表

A	B	A OR B
1	1	1
1	0	1
0	1	1
0	0	0

NOT A 的真值表

A	NOT A
1	0
0	1





为了像使用计算器那样使用计算机,把运算结果显示在屏幕上,PRINT(打印)就是为此而设的一条指令。



例如:PRINT 3 + 5

在上述情况下,按 ENTER 键,会得到结果为 8。若将上式改变一下:

PRINT " 3 + 5 " 则得到的结果为 3 + 5

让我们再试一下:

PRINT " HELLO MY FRIEND "

结果为 HELLO MY FRIEND

从上面可以清楚地看到,引号内的字符和符号原样不变地显示在屏幕上了。

我们可以比较一下。

PRINT " 3 + 5 "

PRINT 3 + 5

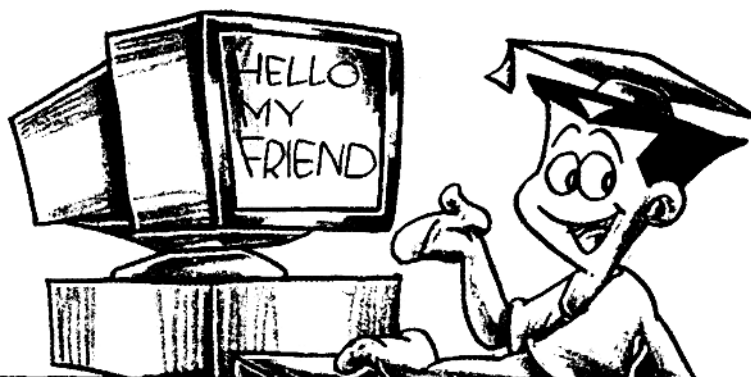


这是位于引号之间的字符串



这是表达式

在引号之间的字符或符号叫“字符串”。打印时它将照原样打印,而不执行任何运算。



BASIC 语言的语句是由单词
符号按照严格规则构成的。



嗨!
大家好。
我是大厨
BASIC。

BASIC 语言为了满足用户的各种不同需要, 设置了各
种不同功能的语句, 共有百余条, 常用的语句有:

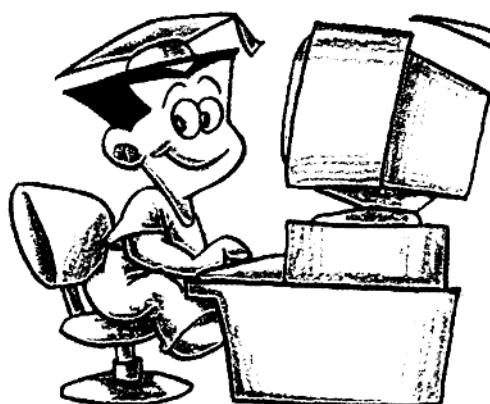
1. 赋值语句(LET)
2. 打印语句(PRINT)
3. 键盘输入语句(INPUT)
4. 读数、置数语句(READ, DATA)
5. 无条件转移语句(GOTO)
6. 条件转移语句(IF—THEN)
7. 循环语句(FOR/NEXT)
8. 子程序语句(GOSUB, RETURN)
9. 数组说明语句(DIM)
10. 文件类语句(OPEN, CLOSE)
11. 图形类语句(COLOR, LINE 等)

这些语句我们将在后面陆续为大家介绍它们的用法。



请尝尝我为
您准备的
自助大餐!





让我们
一起来学习
几个 BASIC
语言的基本
操作键和命
令。

1. $\checkmark$ 键。该键称为回车换行键，简称回车键（即键盘上标有 Enter 的键）。按下回车键后计算机的显示器的屏幕上的光标从光标的现行位置移到下一行开始的位置上。它一般是当用户在键盘上输完一个程序行、一条命令或回答一行数据后键入的。

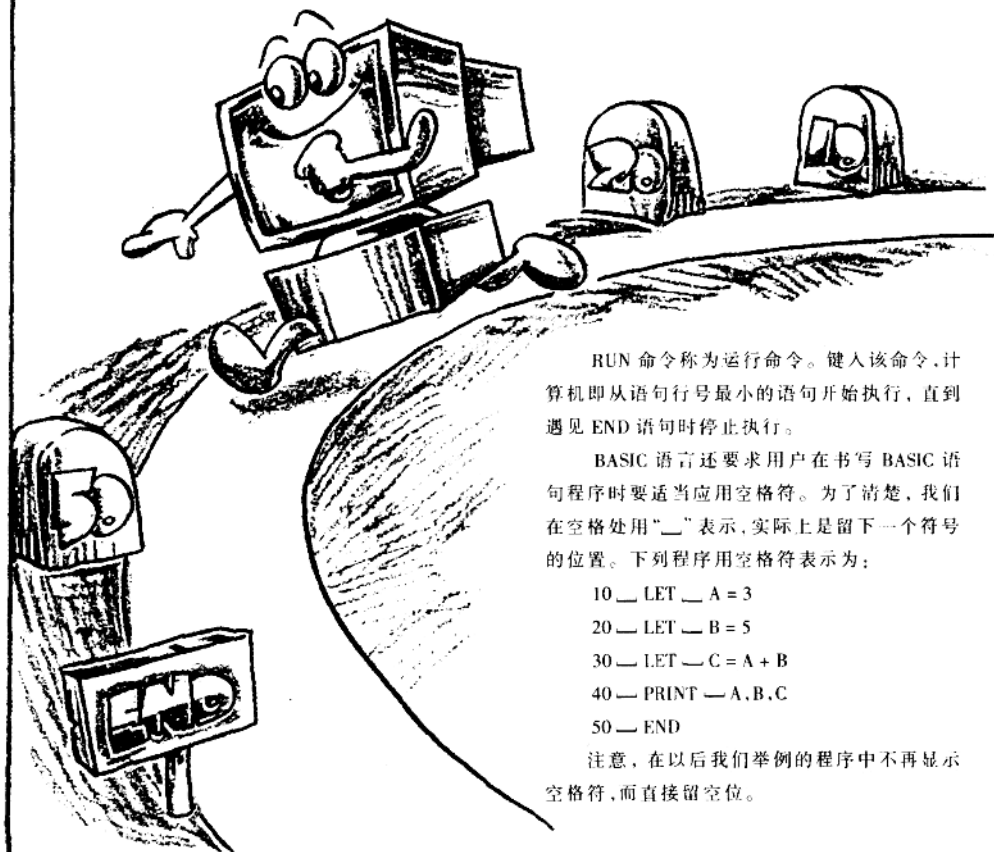
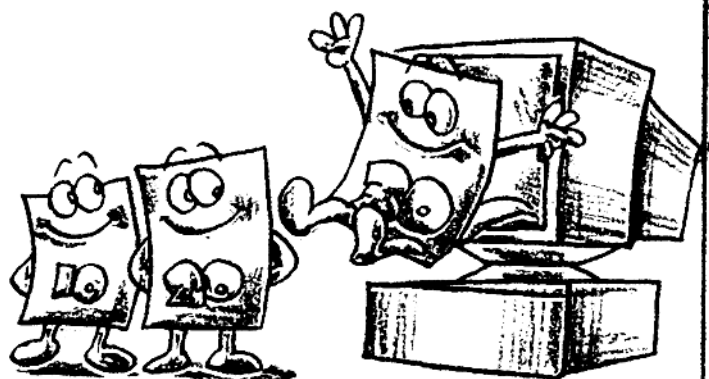


2. Ctrl-L 键。该键称为清除屏幕键，简称清屏键。该键是组合键，即按键时需将 Ctrl 键和 L 键同时按下。按下清屏键后，计算机将显示器屏幕上的内容全部清除，光标移至屏幕的左上角（即第一行第一列）上。但该键并不能清除已键入并存入到内存的 BASIC 程序。

3. NEW 命令，该命令叫清除内存命令，用户在新的一行键入：NEW $\checkmark$

计算机立即清除用户已送入到内存的全部 BASIC 程序，释放出内存中所有的用户工作空间，这一般用于向计算机输入一个新的 BASIC 程序前使用。

LIST 命令称为列程序清单命令。键入该命令计算机立即将已输入的程序按行号大小从小到大原样显示在屏幕上,以便用户进行检查和修改。如果用户需要将内存中已有的程序打印输出,只需要打开打印机后,在键盘上键入 LIST 命令,打印机则将内存中的程序原样打印出来。



RUN 命令称为运行命令。键入该命令,计算机即从语句行号最小的语句开始执行,直到遇见 END 语句时停止执行。

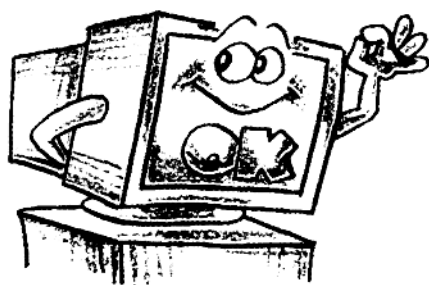
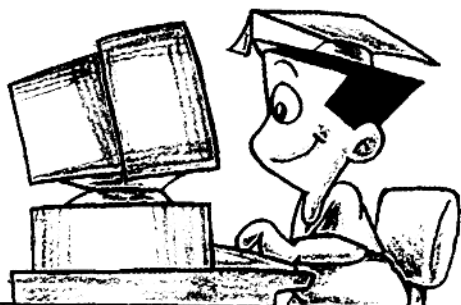
BASIC 语言还要求用户在书写 BASIC 语句程序时要适当应用空格符。为了清楚,我们在空格处用“_”表示,实际上是留下一个符号的位置。下列程序用空格符表示为:

```
10 _ LET _ A = 3
20 _ LET _ B = 5
30 _ LET _ C = A + B
40 _ PRINT _ A,B,C
50 _ END
```

注意,在以后我们举例的程序中不再显示空格符,而直接留空位。

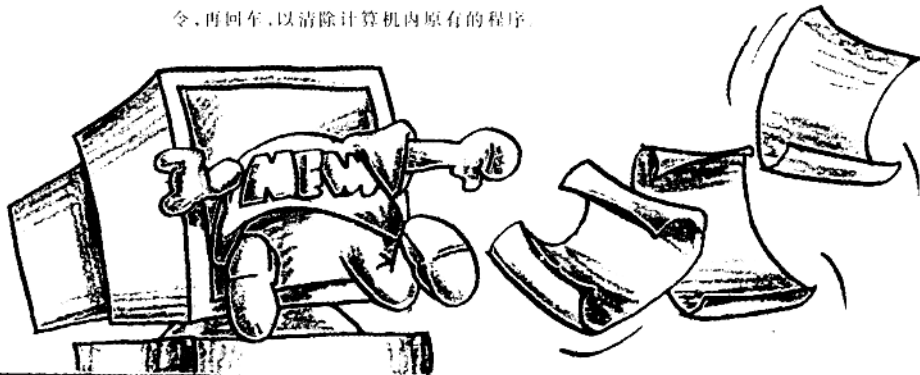


在装有 BASIC 系统的计算机上,开机后出现提示符 C),从键盘上键入 BASIC 及回车后,计算机就进入了 BASIC 操作状态。



在计算机进入 BASIC 操作状态后,计算机的显示器的屏幕上显示 OK,“OK”是 BASIC 的操作状态的提示符,“OK”下面的下横线表示光标,即下一个输入字符的显示位置。

在计算机进入 BASIC 的操作状态后,你就可以输入你的程序了,不过在输入一个新程序之前,要先键入 NEW 命令,再回车,以清除计算机内原有的程序。

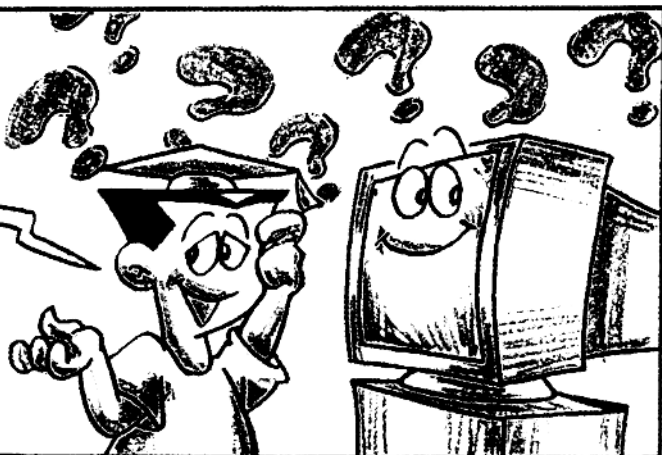


在光标处将程序逐行逐个输入(键盘上每按一个键,显示屏上的光标会指示当前要输入字符的位置)。每输入完一程序行都要按回车键,表示一程序行输入完毕。程序全部输入完毕后按 RUN 命令并回车,计算机才开始执行程序。

在一个程序行尚未输入完毕时,发现按错字符,可用退格键  进行修改,每按一次  键,光标向左退一格,当退回到出错字符处,输入要修改的内容。一个程序行输入完后要修改时,可重新输入行号和正确的语句,则同一行号原来的内容被后来输入的内容所代替。



我们用计算机进行计算后，都希望它能把结果打印或显示出来。如何实现这个愿望呢？



这需要在程序中编上打印结果的语句。PRINT 语句能实现我们的愿望。



PRINT 在英文中是打印的意思。PRINT 的用法有多种，这就看你怎样打印了。下面分别介绍：

★打印常数

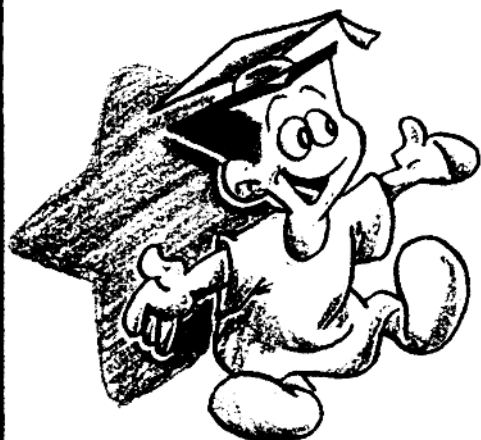
```
【例】 10 PRINT 5
        20 PRINT +5
        30 PRINT -5
        40 END
        RUN
        5
        5
        -5
```

注：第 20 行要打印 +5，在这里“+”号不打印，打印出 5。程序输入完成后，打 RUN 命令，使程序运行，并输出结果。

★打印字符串

在前面已介绍过，这里不重复。

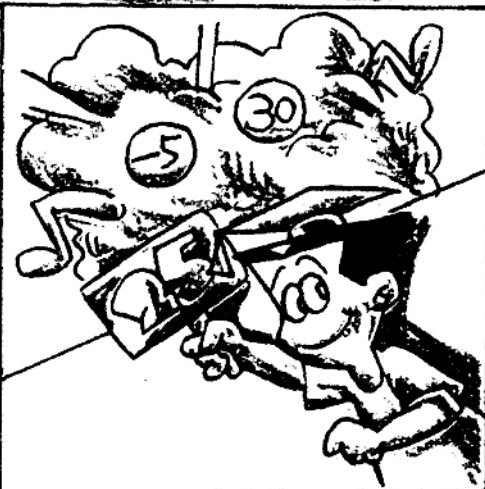
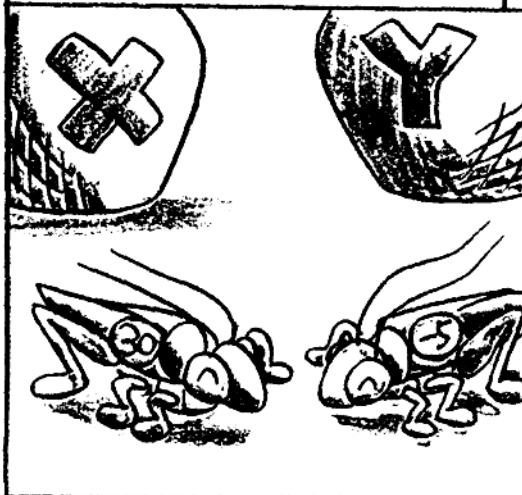




★打印变量或表达式的值

```
【例】 10 LET X = 30
        20 LET Y = -5
        30 PRINT X
        40 PRINT Y
        50 PRINT X + Y
        60 END
RUN
30
-5
25
```

第 10、20 行是给变量 X 和 Y 赋值。可以理解为把 30 赋给 X，-5 赋给 Y。执行第 30 行的结果不是打印 X，而是打印出 X 的值 30。同样，执行 40 行的结果，打印出的是 Y 的值 -5。执行 50 行的结果，打印出 X + Y 的值即 25。PRINT 语句遇到表达式是先进行计算，然后再输出运算结果，即输出表达的值。所以 PRINT 语句即有计算功能又有输出运算结果的功能。



★打印语句的打印格式

【例 1】 10 PRINT "3+5="
20 PRINT 3+5
30 END
RUN✓

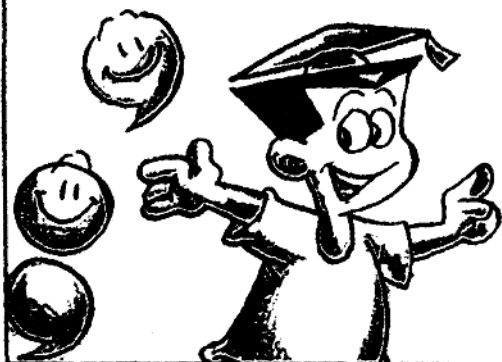
打印结果:3+5=
8

【例 2】 10 PRINT "3+5=";
20 PRINT 3+5
30 END
RUN✓

打印结果:3+5=8

【例 3】 10 PRINT "3+5=",
20 PRINT 3+5
30 END
RUN✓

打印结果:3+5= 8

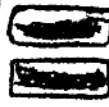


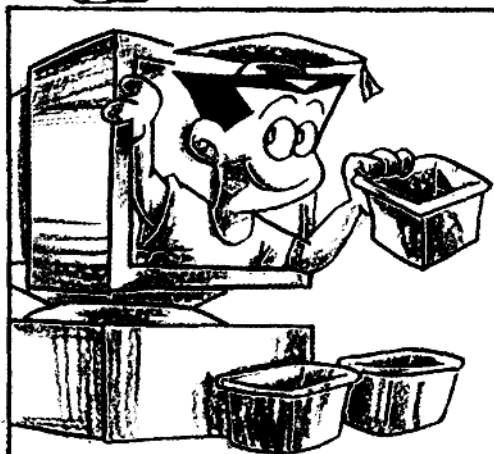
上面三个程序中,第 1 个程序语句结尾没有标点符号(BASIC 语言中一般语句结尾都有标点符号),第 2 个程序、第 3 个程序的第 10 条语句的结尾分别以“;”和“,”结尾,打印出的格式也各不相同,由打印出的结果可以看出,分号和逗号有以下区别:

用“;”时,运算结果紧挨着表达式

用“,”时,运算结果离表达式的开头有 10 个字符的空位。

都是你!让它离我们这么远!



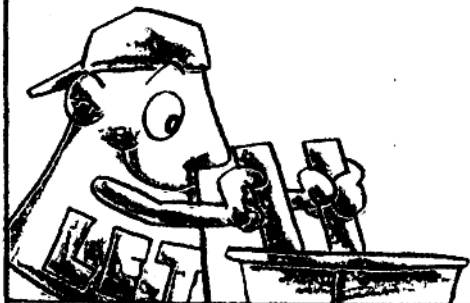


在计算机里有许多存储单元，它们像一个个“小盒子”，这些“小盒子”是存放数据的。我们叫它变量“小盒子”。如果我们要计算，就首先要给这些变量“小盒子”送数字。用LET(赋值指令)就可以给变量“小盒子”送数。

【例】 10 LET S = 0
20 LET I = 4
30 LET S = S + I
40 LET I = I + 1
END



程序中每条语句中都有“=”号，这不是算术中等于号的意思，它的意思是赋值，就是把“=”号右边的数或表达式的值，存放于左边变量的“小盒子”中去。



上例中第10条语句就是把0放到S变量“小盒子”中去。第20条语句就是把4放到I变量“小盒子”中去。第30条语句是把I变量“小盒子”中的数与S变量“小盒子”中的数加起来，并将和仍存放在S变量的“小盒子”中。而I变量“小盒子”的数还保持不变。此语句执行后S变量“小盒子”中的数由原来的0变为4，而I变量“小盒子”中的数仍为4。第40条语句是将I变量“小盒子”中的数加1，并将和仍旧放在I变量“小盒子”中。相加后，I变量“小盒子”的值为5。

