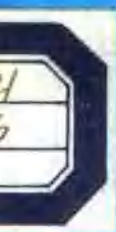


中等专业学校电子类教材

True BASIC

语言

游金水 主编



电子科技大学出版社

UEST PUBLISHING HOUSE

True BASIC 语言

游金水 主编

电子科技大学出版社

[川] 新登字 016 号

内 容 提 要

1984, BASIC 语言的创始人对 BASIC 语言作了重大的改进和发展, 推出了 BASIC 的新版本——True BASIC, 它既吸收了现有高级语言的优点, 又保持了 BASIC 的小型、通用、会话、易学的特点。

本书以结构化、模块化为主线, 自顶向下、逐步求精, 介绍 True BASIC 的基础知识, 顺序结构、选择结构、循环结构的程序设计, 数组、函数与子程序、字符串处理、图形、文件, 最后引入了综合实例及九个实验。使读者对 True BASIC 语言有系统、全面的了解, 并能进行实际的设计应用。

本书以中等专业学校计算机专业和非计算机专业学生为学习对象, 也可作为大专学生和从事计算机技术工作人员的培训和自学用书。

True BASIC 语言

游金水 主编

*

电子科技大学出版社出版

(成都建设北路二段四号) 邮编 610054

成都市农垦印刷厂印刷

新华书店经销

*

开本 787×1092 1/16 印张 10.875 字数 265 千字

版次 1995 年 2 月第一版 印次 1995 年 5 月第二次印刷

印数 5501—8501 册

ISBN 7-81043-099-8/TP·44

定价: 9.50 元

前 言

近年来,我国计算机应用事业迅速发展,计算机的应用在各行业得到普及。计算机知识已成为当代各行各业的知识结构中不可缺少的重要组成部分,因此计算机知识对非计算机专业学生也是一门重要的课程。

为了使用计算机,就必学会计算机语言。BASIC语言是为初学者设计的一种计算机语言。从1964年BASIC问世以来,它迅速在全世界推广和普及,受到了初学者的热烈欢迎,在我国同样如此。

由于BASIC语言出现的比较早,和早期其它语言一样,不是结构化的。在计算机科学迅速发展的今天,所有语言都在不断地发展和完善。BASIC语言也同样如此,1984年,BASIC语言的创始人John G. Kemeny和Thomas E. Kurtz对BASIC语言作了重大的改进和发展,推出了BASIC的新版本——TrueBASIC。它一方面吸收了现有高级语言的优点,又保持了BASIC的小型、通用、会话、易学的特点,采用结构化、模块化进行程序设计,简单、易学、又有足够的能力适应各种较大的应用课题。

本书是根据《非计算机专业计算机课程建议试行标准》编写的,适用于中等专业学校非计算机专业和职业中专计算机专业。也是计算机爱好者自学的好教材。全书分为十章,每章附上机指导。本书以结构化、模块化为主线,自顶向下,逐步求精,简明介绍各种基本结构程序的设计,并着重于应用。

本书由南平工业学校游金水老师担任主编,并编写了绪论、第五章、第六章、第七章、第八章、第九章、第十章以及相应章节的上机指导书和附录部分;龙岩工业学校童长卫老师编写了第一章、第二章、第三章、第四章和相应章节的上机指导书。游金水老师对全书进行了统稿。福建电子工业学校陈建辉老师担任主审。参加审稿的还有:福建侨兴工业学校陈能义老师、福建水利电力学校王集祥老师等。在此向他们表示感谢!

由于我们水平有限,难免存在错误之处,恳请专家和读者不吝指正。

编 者

1995年1月

目 录

绪 论

一、True BASIC 的发展及特点	(1)
二、程序设计语言	(2)
三、程序设计的一般过程	(3)

第一章 True BASIC 语言的基础知识

1.1 True BASIC 语言的基本字符集	(4)
1.2 常量与变量	(5)
1.3 标准函数	(7)
1.4 True BASIC 表达式	(10)
1.5 True BASIC 的语句和程序结构	(11)
1.6 流程图和 N-S 结构流程图	(13)
习题	(16)

第二章 顺序结构程序设计

2.1 变量的赋值	(17)
2.2 数据的输出与输出格式语句	(23)
2.3 顺序结构应用程序举例	(26)
习题	(28)

第三章 选择结构程序设计

3.1 关系表达式、逻辑表达式	(29)
3.2 简单分支选择结构	(32)
3.3 多路分支结构程序	(39)
3.4 选择结构应用举例	(40)
习题	(43)

第四章 循环结构程序设计

4.1 计数型循环 FOR—NEXT	(45)
4.2 条件型循环	(49)
4.3 循环结构程序设计应用举例	(53)
4.4 多重循环	(56)
习题	(60)

第五章 数组

5.1 数组的概念	(62)
5.2 数组的赋值与输出	(63)
5.3 数组应用举例	(66)
习题	(70)

第六章 函数与子程序

6.1 自定义函数及应用举例	(73)
6.2 子程序及应用举例	(78)
6.3 库文件	(80)
习题	(84)

第七章 字符串处理

7.1 字符串的输入与输出	(87)
7.2 字符串的连接与比较	(88)
7.3 字符串函数	(89)
7.4 字符串处理举例	(93)
习题	(95)

第八章 图形

8.1 图形坐标	(97)
8.2 基本画图语句	(98)
8.3 多窗口操作及动画显示	(105)
8.4 画图模块与图形变换	(109)
习题	(112)

第九章 文件

9.1 文件概念	(113)
9.2 正文文件	(115)
9.3 记录文件	(121)
9.4 字节文件	(124)
习题	(125)

第十章 综合实例

10.1 模块化程序设计	(126)
10.2 引例	(127)
10.3 程序质量分析	(135)
10.4 模块引例完成下面应用程序的设计	(136)

实 验

实验一	函数、表达式的上机练习.....	(137)
实验二	顺序程序的上机练习.....	(139)
实验三	选择结构的上机练习.....	(141)
实验四	循环结构程序上机与调试.....	(144)
实验五	数组上机与程序调试.....	(146)
实验六	函数与子程序上机运行与调试.....	(147)
实验七	字符处理程序上机运行与调试.....	(149)
实验八	图形程序运行与调试.....	(151)
实验九	文件上机运行与调试.....	(152)

附 录

附录 I	ASCII 码符号集	(154)
附录 II	True BASIC 保留字	(155)
附录 III	True BASIC 语法一览表	(155)
附录 IV	True BASIC 系统命令一览表	(164)

绪 论

一、True BASIC 的发展及特点

BASIC 语言是美国计算机语言学家 John G. Kemeny 和 Thomas E. Kurtz 吸收了 FORTRAN IV 和 ALGOL60 语言的优点于 1964 年推出的。由于它易学、简洁等优点，所以得到推广，并深受欢迎。到达了 1970 年 BASIC 语言已成为一种能处理相当复杂的应用问题的语言。在这同时，计算机科学本身发展迅速，“结构化程序设计”的概念被普遍接受。虽然 BASIC 中已建立许多结构，但它们并不能写出完成结构化的程序。为此，1984 年两位 BASIC 语言的创始人根据美国国家标准 (1984)，推出了 BASIC 语言的新纪元——True BASIC。

True BASIC 语言也称为真正的 BASIC。它保持了 BASIC 原有易学、简洁等特点，并吸收了 PASCAL, FORTRAN 等高级语言的优点。它不仅适用于科学计算、数据处理、文件管理，而且还具有较强的图形功能。为 BASIC 语言的发展开辟了新的前景。

True BASIC 语言是以结构化、模块化为主线，“自顶向下、逐步求精”。所谓的“自顶向下”是将一大问题的求解分解为若干个小问题的求解过程；所谓的“逐步求精”是指对问题的求解过程的方法与步骤的描述由粗到精逐步完善，直到能写出完整的程序为止。一个简单的应用问题，可以直接写出详细的算法。但是一个复杂的问题，要一下子直接写出其中每一步骤就困难了。这时我们可以采用“自顶向下，逐步求精”的方法，从上往下、从粗到细地逐步具体化。例如：缝制一件时装往往不是一下子就能拿起针线开始缝制的，而是先要想好时装的款式以及所用的而料，然后根据量体裁衣的原则，量出这种款式所需要的所有尺寸再进行裁剪，最后进行缝纫。这就是“自顶向下，逐步求精”的精神。

例 找出三个数中的最大数

要解决这一问题，首先要有三个数（输入三个数），然后再比较这三个数，最后打印出最大的数（输出最大数）。其中第一步（输入三个数）和第三步（输出）是计算机可以实现的基本操作。而第二步就要再对它进行细化（求精）了。如用 a, b, c 表示这三个数，并用 $\max$ 存放其中的最大数。把第二步再分成两步来求，1. 比较 a 和 b ，并将大者放在 $\max$ 中，然后再把 $\max$ 同 c 比较，如果是 c 大，再把 c 存放在 $\max$ 中，这样最终存放在 $\max$ 中的数就是三个数中最大的数。我们可以将其归纳为如下步骤：

- 1: 输入三个数（输入 a, b, c ）
- 2: 找出最大数
- 3: 打印最大数（打印 $\max$ ）

而第二步又可分为两步：

- 2.1: 找出 a, b 中大者，并存放在 $\max$ 中
- 2.2: 找出 $\max$ 和 c 中的大者，并存放在 $\max$ 中

我们把这样的解题步骤用流程图（N-S 图）来表示如下（图 A）

二、程序设计语言

自然语言是人们交流思想与信息的工具，为了让计算机执行解题算法（人们解题的思路和步骤的方法），必须把算法描述成计算机能够理解和执行的形式。程序设计语言就是人和计算机交流信息的工具。这就好像阅读外文书刊必须要懂外文一样的道理。

计算机语言有三大类：机器语言，汇编语言和高级语言。

1. 机器语言 (machine language)

电子计算机是由电子元件组成的。它们只有两种状态，即开或关，并分别用 1 或 0 来表示。所以采用二进制数。机器语言中的指令是一个由 1 或 0 码组成的序列。有的计算机用一个字节（8 位）二进制数信息来表示一个指令，也有的用两个字节或四个字节（16 位或 32 位）的二进制数信息来表示一个指令。

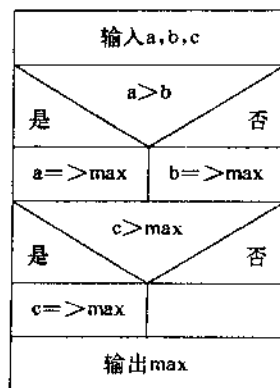


图 A

机器语言程序就是由机器指令构成的。这种程序虽然计算机能够理解并直接执行，但由于它都是用一串 0 和 1 的数字表示，对初学者来说不够直观难学、难记也难懂，如果程序中出现错误也不便发现和修改，而且不便于交流。所以机器语言的应用有很大的局限性，对广大用户是很不方便的。

2. 汇编语言 (assembly language)

汇编语言是用一些简单的“助记忆符号”来代替二进制数代码的，所以也称为“符号语言”。例如用 MOVE 表示传递数据，ADD 表示加法等。

然而计算机不能直接执行汇编语言写的程序，要使用一个称为汇编程序的软件将用汇编语言编写的程序翻译成机器语言程序才能执行。

汇编语言程序基本上保留了机器语言的优点，所占的内存少，执行速度快，提高了程序的易读性和程序的设计效率，但是，它同机器语言一样都是“低级语言”。要使用汇编语言，要了解计算机的内部结构。所以汇编语言对广大的用户来说也很不方便。

3. 高级语言 (high level language)

高级语言是为便于人们使用计算机而设计的计算机语言。它用类似于英语中的动词来发出各种操作命令及用类似数学中的表达式来进行数学运算。如：“PRINT A”表示“打印 A 的值”，用“LET X=A+C”表示“将算式 A+C 的结果赋给变量 X”。人们容易理解，而且用变量名代替了存储单元的地址，不必人工分配。

高级语言的种类很多，常用的高级语言有：BASIC，FORTRAN，COBOL，PASCAL，C 语言等。

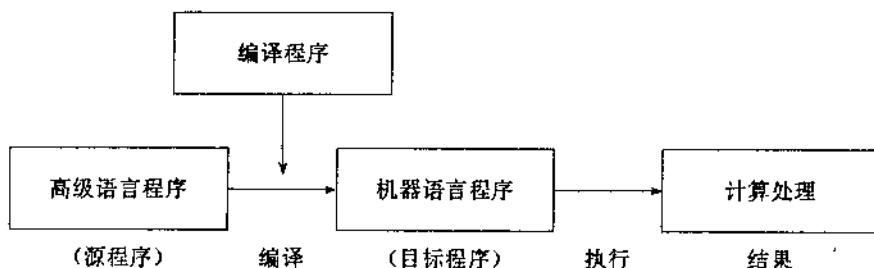
同样，高级语言也不能在计算机上直接执行，只有把它翻译成机器语言后才能执行。无论哪一种高级语言都配有相应的编译或解释程序，只有这样才能运行。编译或解释程序是

由系统配备的，所以设计程序时就不必了解计算机内部结构和工作原理，也不必去记忆指令。将高级语言（源程序）转换成机器语言程序（目标程序），有两种不同的方式：

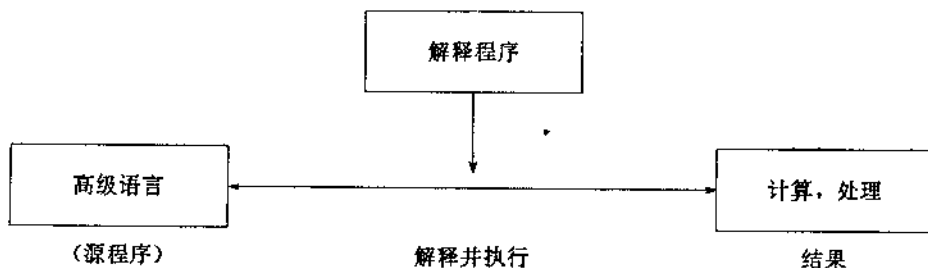
（1）编译方式：将程序全部译成机器语言后，再将机器语言的目标程序交给计算机执行。（如图 B）

（2）解释方式：它不是先产生目标程序再交付执行，而是边翻译边执行，翻译一句执行一句，不产生整个目标程序。（如图 C）

编译方式好象笔译，一次翻译出整个译文。解释方式如同口译，说一句译一句。用编译方式得到的目标程序执行速度快，但占计算机的内存空间大；解释方式执行慢，占内存空间小。多数高级语言都采用编译方式。True BASIC 既可以用解释方式也可以用编译方式。



图B 编译方式执行高级语言程序的过程



图C 解释方式执行高级语言程序的过程

三、程序设计的一般过程

学习计算机高级语言的目的是编写程序在计算机上解决应用问题。设计应用程序的一般过程是：

1. 解题分析 明确题目的要求。
2. 确定算法 算法是指人们为了解决某一问题所拟定的正确的基本操作序列，即人们解题的思想和步骤。
3. 编写程序 是按语言的语法规则，根据算法写出一系列计算机操作的指令。

第一章 True BASIC 语言基础知识

本章介绍有关 True BASIC 语言的基础知识。主要涉及 True BASIC 语言的基本字符集、常量、变量、标准函数、表达式以及 True BASIC 的语句与程序结构。

1.1 True BASIC 语言的基本字符集

每一种语言都是由某一字符集及定义在其上的语法所构成。如汉语，是由所有汉字及汉语语法所构成，但汉语中的汉字及语法都没有一个严格的定义，有许多习惯用语，比如：晒被子是把被子拿去晒，而晒太阳是被太阳晒。而计算机语言不同于自然语言，它有严格定义的字符集及不可能产生二义性的语法。它的基本字符集包括了能出现在用该语言编写的程序中的所有符号。True BASIC 语言的基本字符集可以分为以下几类：

一、数字

0、1、2、3、4、5、6、7、8、9

二、字母

字母分为大写和小写字母，它们是：

A、B、C、...X、Y、Z，a、b、c、...x、y、z

三、专用符号

表 1-1 专用字符

	名 称		名 称		名 称		名 称
+	加 号	(	左括号	"	双引号		竖直线
-	减 号	)	右括号	'	单引号	\	右斜线
*	乘 号	.	小数点	&	合并号	[	左方括号
/	除 号	,	逗 号	\$	美元号	]	右方括号
^	乘方号	;	分 号	#	#字符号	{	左花括号
=	等 号	:	冒 号	%	百分号	}	右花括号
>	大于号	!	叹 号	@	圆 A	␣	空 格
<	小于号	?	问 号	_	下划线		

必须要指出的是在程序中经常用到的空格，虽然在屏幕上什么也没有显示，但它却是

一个字符, 占有一个字符位置。

符号名是人们使用计算机时对程序中的各变量, 子程序及自定义函数等取的名字。True BASIC 规定符号名用英文字母(大小写均可)开头, 后跟字母、数字或下划线的一串字符来表示, 其最大长度为 31 个字符, 在 True BASIC 系统中规定的保留字符是由于它们有专门的含义, 不得作符号名使用。如果 True BASIC 是在汉字系统的支持下运行的, 那么还包括了汉字系统能提供的所有汉字和其它符号。

1.2 常量与变量

计算机语言是用来描述计算机程序的。一个程序的描述包括了两个方面, 即数据和操作步骤的描述。数据是操作的对象, 操作是对数据进行的加工。例如一个菜谱(对厨师而言, 它就是一个程序), 它包括了原料和加工步骤两个方面。原料指出需要哪些配料, 诸如肉、葱、盐、油等等。加工步骤指出按什么步骤, 如何对这些原料进行加工, 诸如先炸几分钟, 再煮几分钟等等。计算机程序也是如此, 因此, 著名计算机科学家沃思指出:

数据结构 + 算法 = 程序

True BASIC 语言的数据可分为:

1. 数值型 如: 1、2、-3.8、6.329、0.786、-0.7897、70 等。
2. 字符型 如: "ABC"、"张三"、"COMPUTER"、"LI MI" 等。
3. 逻辑型 如: $A > B$ 、 $B < D$ 、 $A < > 0$ 、 $B < = 3$ 等, 其值只有二个即真或假。

而每一种类型又有常量、变量、函数、表达式等几种形式。其中逻辑型只有函数与表达式两种形式。

一、常量

在程序的运行过程中其值不能被改变的量称为常量。常量在程序中可以直接被引用, 它分为两种类型。

1. 数值型常量

在 True BASIC 语言中数值型常量有两种书写格式: 一般格式与指数格式。

(1) 一般格式

如 3、6、-4.5、-7 等。但要注意不能有分节符。如: 1000000 不能写成 1, 000, 000。

(2) 指数格式:

书写格式为: $\pm d. dE \pm d$

其中 d 为正整数, 正号 (+) 可以省略。它表示 10 的若干次幂。如: $+3.1E3$ 表示 3.1×10^3 , $-0.6E3$ 表示 -0.6×10^3 , $1.0E-7$ 表示 1×10^{-7} 等。

2. 字符型常量

True BASIC 除了能处理数值型数据外, 还能处理字符型数据(字符型数据也称为字符串或字串)。如一个人的姓名, 一个单位的地址等等, 这些数据都是属于字符型的。字符型常量是由任意字符序列组成, 为表示区别 True BASIC 规定在字符串的首尾各用一个双引号将字符串括起来。一个字符串最多可以包含 32000 个字符。如: "1234", "ABCD" 等都是 True BASIC 的字符串常量。True BASIC 对字符串中字母的大小字是有区别的, 如: "ABC" 和 "abc",

“aBC”是不同的串。而空格也是一个字符。

没有任何字符的串称为空串，它由一个双引号紧跟一个双引号构成，即：“”。

如果字符串中包含有双引号，那么必须重复这个双引号，如：“”表示只由一个双引号字符组成的字串。

如果 True BASIC 在汉字系统的支持下运行，那么，汉字也可以作为组成字符串的符号，如“姓名 1”，“成绩 2”等。

二、变量

程序在运行过程中，其值可以被改变的量称为变量。变量用一个名来标识，在计算机内存中占据一定的单元，这些单元存放该变量的值，所以变量有名和值两个概念。在程序的运行过程中被改变的是变量的值而不是变量的名。

1. 变量名

变量名是我们给变量取的名字，它只能由字母、数字和下划线组成，但必须由字母开头。在 True BASIC 中变量名的长度最大可达 31 个字符。在变量名中，不分大、小写字母，所以 A1 和 a1 表示同一变量。给变量取名应遵循便于记忆，不易混淆的原则，而且不能和 True BASIC 语言中的保留字同名，同时也不能给两个不同的东西（象简单变量、数组变量、函数、子程序等）取相同的名字。

True BASIC 的保留字见附录 II

True BASIC 的其它关键字和函数名象 FOR、SIN 等最好也不要用来做变量名，否则容易引起混淆。

以下变量名均是合法的：

a, al, c, number, R-1, R2

以下变量名是非法的：

la, if, time, b=c, b-2, b/2

la 不是以字母打头，if, time 分别是 True BASIC 的语句关键字和函数名，其它几个都含有非字母字符。

2. 变量的类型

变量和常量一样也有类型。变量的类型确定了其值的取值范围。它分为数值型和字符型。字符型变量应在变量名后加字符“\$”以示区别。如 A 和 A\$ 是两个不同的变量，前者为数值型，其值只能是数值型的，后者为字符型，其值只能是字符型。

3. 简单变量和数组变量

一个变量只能存放一个数据。它通过变量名进行访问，这类变量称为简单变量。为了处理一组类型相同的数据，如数列、矩阵等，True BASIC 提供了数组。

一个数组由一系列的数组元素组成，每个元素相当于一个简单变量，它可以存放一个数据，但数组元素的访问和简单变量不同，它除了要指明数组名外，还要指明数组的下标。下标表明了该元素在数组中的“位置”。下标的个数可以是一个、二个或更多，下标的个数称为数组的维数。True BASIC 规定数组的最大维数为 10。每维的下标都有上、下界。在使用中下标的值不能超出这个界限。

例如：数组 A (10) 是由 A (1)、A (2)、A (3)、…、A (10) 等 10 个数组元素组成，

每个元素象 A (1), A (2) 等都可以存放一个数据, 数组元素括号中的数, 如 1、2、3 等称为下标。这个数组下标数只有一个, 所以称为一维数组。一维下标的上界是 10, 下界为 1。

数组在使用之前必须先定义。在 True BASIC 中用 DIM 语句定义数组。一条 DIM 语句可以定义一个或多个数组, 如果定义多个数组, 那么每个数组之间用逗号隔开。一个数组的定义包括数组名、数组类型和每维下标上、下界的说明, 其格式如下:

DIM (数组项) [, (数组项) ...]

其中 (数组项) = (数组名) (数组类型) ((下标说明) [, (下标说明) ...])

(数组名) 的规定和简单变量名的规定相同。对于数值型数组, 不要加 (数组类型) 说明, 对于字符型数组 (数组类型) 为 "\$"。而 (下标说明) 有两种表示方式, 一种用一正整数 N 表示, 这种方式其下标的下界自动定义为 1, 上界定义为 N。另一种方式为: N1 TO N2, 这种格式下标的下界定义为 N1, 上界为 N2, 其中 N1, N2 为整数且 $N1 < N2$ 。例如:

DIM A\$ (2, 3), B (-3 TO 4), C (20, 30)

此语句定义了三个数组。数组 A 为字符型的二维数组, 其元素为

A\$ (1, 1), A\$ (1, 2), A\$ (1, 3), ..., A\$ (2, 3) 共 6 个。

数组 B 为数值型的一维数组, 其元素为: B (-3), B (-2), ..., B (4) 共 8 个。

数组 C 为数值型的二维数组, 其元素为: C (1, 1), C (1, 2), ..., C (20, 30) 共 600 个。

True BASIC 语言中有些语句是对整个数组进行操作的, 有些是对一个数组元素进行的。有关数组的其它知识我们将在后继章节中加以详细介绍。

1.3 标准函数

简单地说一个函数就是一个“调用”。True BASIC 提供了许多标准函数, 对于这些函数我们只要在程序中按格式给出函数名和相应的自变量, 程序运行时系统将自动调用一段相应的程序对函数进行处理, 然后将处理结果——函数值返回给程序。例如, 要计算 $\sqrt{100}$, 我们在程序中只要按其格式要求给出函数名和自变量, 即 SQR (100), 程序执行时 True BASIC 将自动调用一段程序对其进行计算并将结果返回给我们, 这样在 SQR (100) 出现的地方就能得到结果 10。

根据自变量和函数返回值的类型, True BASIC 的标准函数可分为:

一、数值型

其自变量可以是数值型的常量、变量或表达式, 而其返回的函数值也是数值型的。常用的数值型函数主要有下面几个。

1. 求绝对值函数

格式: ABS (X)

功能: 求数值表达式 X 的绝对值。如 ABS (-3) 其返回值为 3。

2. 指数函数

格式: EXP (X)

功能：求 e^x

3. 对数函数

格式：LOG (x) 或 LOG2 (x) 或 LOG10 (x)

功能：分别求以 e、2、10、为底 x 的对数：即 $\log(x)$ ， $\log_2(x)$ ， $\log_{10}(x)$

4. 取整函数

格式：INT (X)

功能：求不大于 X 的最大整数。

例：INT (4.6) 和 INT (4.3) 其返回的函数值为 4。

而 INT (-4.5) 其返回的函数值为 -5。

5. 求余数函数

格式：REMAINDER (X, Y)

功能：求 X 除以 Y 时所得余数。

例 REMAINDER (5, 3) 返回的函数值为 2

REMAINDER (1.3, 1) 返回的函数值为 0.3

REMAINDER (-1, 3) 返回的函数值为 -1

REMAINDER (7, -3) 返回的函数值为 1

6. 符号函数

格式：SGN (X)

功能：若 $X > 0$ ，其返回值为 1。若 $X < 0$ ，其返回值为 -1。若 $X = 0$ ，其返回值为 0。

7. 平方根函数

格式：SQR (X)

功能：求 X 的平方根，X 必须大于等于 0。

8. 四舍五入函数

格式：ROUND (X) 或 ROUND (X, N)

功能：前者求 X 四舍五入后所得整数，后者对 X 第 N+1 位进行四舍五入。

例 ROUND (3.2) 其返回的函数值为 3

ROUND (-3.6) 其返回的函数值为 -4

ROUND (123.236, 2) 其返回的函数值为 123.24

ROUND (123.236, 0) 其返回的函数值为 123

ROUND (123.236, -1) 其返回的函数值为 120

ROUND (123.236, -2) 其返回的函数值为 100

9. 求两数中的较大数

格式：MAX (X, Y)

功能：求 X, Y 两数中的较大数。

例 MAX (-3, 4) 其返回值为 4

10. 求两数中的较小数

格式：MIN (X, Y)

功能：求 X, Y 两数中较小的一个数。

例 MIN (-3, 4) 其返回值为 -3。

11. 随机函数

格式: RND

功能: 产生一个 (0, 1) 之间的随机数。但每次运行程序时, 产生的随机数序列是相同的。要产生不同的随机数序列应在程序前加 RANDOMIZE 语句。

二、三角函数

三角函数全部涉及到角。一般地 True BASIC 用弧度来表示角, 但如果在程序前加一条 OPTION ANGLE DEGREES 语句, 则这些函数的自变量和结果都用角度来表示。常用的三角函数主要有:

1. π

格式: PI

功能: 返回 π 的值, 即 3.1415926……, 精确到机器所能处理的位数。(我们可以看成是由系统确定的常量)

2. 正弦

格式: SIN (X)

功能: 求 X 的正弦

3. 余弦

格式: COS (X)

功能: 求 X 的余弦

4. 正切

格式: TAN (X)

功能: 求 X 的正切

5. 反正切

格式: ATN (X)

功能: 求 X 的反正切值

6. 弧度到度的转换函数

格式: DEG (X)

功能: 将弧度 X 转换成对应的角度。不论程序中有否 OPTION ANGLE DEGREES 语句, X 总是当做弧度计。

例 DEG (PI) 返回值为 180

7. 度到弧度的转换函数

格式: RAD (X)

功能: 将度 X 转换成对应的弧度

例 RAD (180) 返回值为 3.1415926

三、字符串函数

这些函数或者其返回的函数值为字符串, 或者以字符串作为自变量, 或返回值和自变量都为字符串。

四、逻辑函数

这些函数的返回值为逻辑值，它们的主要用来检测文件或数据是否结束，以及是否从键盘上按了一键等，一般只用在分支或循环控制的条件下。

字符和逻辑函数我们将在后面学习。

1.4 True BASIC 表达式

所谓表达式就是用运算符和圆括号，将常量、变量、函数等连接起来具有一定意义的式子。一般地，单个常量、变量或函数作为表达式的特例也算一个表达式。

True BASIC 表达式分为三类，即数值表达式，字符表达式和逻辑表达式，本节只介绍数值表达式。

用算术运算符和圆括号将数值型常量、变量和返回值为数值型的函数连接起来的式子称为数值表达式。它的值为数值型的。注意，表达式中只使用圆括号。

1. 算术运算符

True BASIC 提供了以下算术运算符

符号	意义
+	加
-	减
*	乘
/	除
^	乘方

共 5 个。

2. 运算优先级

在一个表达式中，圆括号的优先级最高，（如果有多层，由内向外逐层进行）其次是函数，再其次是乘方，接下来是乘、除，最后是加减。同一级运算按从左到右的次序进行。即：（）、函数、^、*、/、+、-

如：数学式子： $\frac{2\sqrt{(10+56)^3}}{2}-4$ 其值是 532.187，它的 True BASIC 表达式：

$2 * \text{sqr}(10+56) \wedge 3 / 2 - 4$ ，其运算次序为：

- (1) $10+56$
- (2) $\text{sqr}(10+56)$
- (3) $\text{sqr}(10+56) \wedge 3$
- (4) $2 * \text{sqr}(10+56) \wedge 3$
- (5) $2 * \text{sqr}(10+56) \wedge 3 / 2$
- (6) $2 * \text{sqr}(10+56) \wedge 3 / 2 - 4$

按 True BASIC 表达式的运算次序计算的结果也是 532.187 跟用数学方式计算结果是一样的。

3. 数值表达式的取值范围和运算精度

在不同的计算机上数值表达式的取值范围以及运算精度是不同的。一般地，True BASIC