

计算机软件手册丛书

GW BASIC 使用手册

GW BASIC 使用手册

赵彦斌 黄国元 编

天津大学出版社

TP312

计算机软件丛书

GW BASIC 使用手册

匙彦斌 黄国元 编



天津大学出版社

内容提要

本手册包含了目前应用最为广泛的 GWBASIC 3.2 版本的全部语句、命令和函数,并给出了完整的语法规则和使用格式。

为了方便用户查阅,本书根据程序设计的需要,按语句和命令的类别划分成章节。本书可作为程序设计人员和初学者使用、学习 GWBASIC 语言的工具书和参考教材。

JS/82/17

计算机软件丛书

GWBASIC 使用手册

匙彦斌 黄国元 编

*

天津大学出版社出版

(天津大学内)

邮编:500072

河北省昌黎县印刷厂印刷

新华书店天津发行所发行

*

开本:787×1092 毫米 $1/16$ 印张:7 字数:157 千

1997 年 3 月第一版 1997 年 3 月第一次印刷

印数:1—5000

ISBN 7-5618-0936-0
TP·95 定价:8.00 元

序 言

一个以计算机、多媒体和通信技术为主要特征的信息革命,正在我国兴起。随着信息技术的高速发展,信息科学、技术及产业已成为推动国民经济增长和社会进步的巨大推动力。邹家华副总理在国务院信息化工作领导小组第一次全体会议上,就我国信息化建设的有关问题做了重要讲话。他指出,为了推动我国经济实现两个根本转变,必须把加快信息化进程放到重要的战略地位上来,并强调,我国的信息化建设,必须立足中国国情。无论是信息基础设施、重大的信息工程项目建设,还是信息资源的开发利用,都要充分考虑我国的经济、技术基础,要立足于我国的信息技术和信息产业。

国家信息化建设,涉及各行各业、各个部门。增强全民信息化意识,培养大批信息化人才,普及信息化技术和产品,促进公众信息化能力的提高,是信息化建设的重要措施。

为满足迫切需要掌握微型机常用软件 and 语言的使用和操作的读者的要求,我们组织了部分在信息化领域第一线从事教学和科研的教师、学者和专家,编写了这套微型机常用软件的语言系列丛书。根据作者使用的经验,有针对性地、较全面地介绍各个领域软件的功能、特点及使用方法。本丛书涉及面广、实用性强、概括简练,可满足不同读者的要求。

本系列丛书由边莫英负责组织有关人员编写、定稿。天津大学出版社给予了大力的支持,在此表示感谢。

边莫英
1996年7月

目 录

第 1 章 BASIC 语言的特点及运行环境	(1)
1.1 BASIC 语言的发展与版本简介	(1)
1.2 GWBASIC 语言的特点	(1)
1.3 GWBASIC 的运行环境	(1)
1.4 GWBASIC 的启动	(2)
1.5 GWBASIC 的使用方式	(3)
第 2 章 BASIC 的语法规则	(4)
2.1 BASIC 字符集	(4)
2.2 BASIC 保留字	(4)
2.3 BASIC 常数	(5)
2.4 变量	(7)
2.5 下标变量	(8)
2.6 BASIC 函数	(10)
2.7 运算符与表达式	(16)
第 3 章 BASIC 的程序结构	(21)
3.1 BASIC 语句和命令	(21)
3.2 BASIC 程序结构	(22)
第 4 章 BASIC 输入和输出	(23)
4.1 输入数据的语句	(23)
4.2 输出语句	(26)
4.3 输入、输出的辅助函数及变量	(29)
4.4 提供系统日期和时间的语句和变量	(30)
第 5 章 控制语句	(32)
5.1 无条件控制转移语句	(32)
5.2 有条件转移语句	(33)
5.3 循环控制	(39)
第 6 章 数据文件及操作	(42)
6.1 BASIC 数据文件的有关概念	(42)
6.2 顺序数据文件的建立和输入、输出	(43)
6.3 随机文件的建立和输入、输出	(47)
第 7 章 BASIC 绘图	(50)
7.1 绘图环境的设置	(50)
7.2 绘图语句及相关函数	(54)
7.3 窗口与视口的设置	(58)

7.4	动画设计语句及函数.....	(59)
7.5	DXY-880A 绘图仪 BASIC 命令	(61)
第 8 章	GW BASIC 的特殊语句及函数	(667)
8.1	BASIC 通信	(67)
8.2	机器级语句及函数.....	(68)
8.3	其他语句.....	(72)
第 9 章	BASIC 命令与键盘操作	(73)
9.1	编辑命令.....	(73)
9.2	文件的管理与操作命令.....	(75)
9.3	程序的执行命令.....	(77)
9.4	BASIC 屏幕编辑键	(79)
9.5	BASIC 功能键的定义及使用	(79)
第 10 章	编译 BASIC	(81)
10.1	编译 BASIC 的构成.....	(81)
10.2	编译 BASIC 的使用.....	(81)
10.3	编译 BASIC 与解释 BASIC 之间的主要区别	(83)
附录 A	GW BASIC 语句、命令和函数索引	(85)
附录 B	ASCII 字符代码表	(91)
附录 C	GW BASIC 错误信息	(96)
附录 D	GW BASIC 专用键	(102)

第 1 章 BASIC 语言的特点及运行环境

1.1 BASIC 语言的发展与版本简介

BASIC 语言是国际上通用的计算机语言之一,各种规模的计算机系统均配有 BASIC 语言。

BASIC 语言于 20 世纪 60 年代中期问世,是由美国达默思(Dartmouth)大学的约翰·凯梅尼(John Kemeny)和托玛斯·库尔兹(Thomas Kurtz)两人最早研制的。它是为教学服务的一种通俗、简洁、明了的教学版本。BASIC 为 Beginner's All-Purpose Symbolic Instruction Code 的缩写。当时仅有十几条命令、语句和十几个函数,称最小 BASIC。后来,最小 BASIC 扩充了一些语句和功能,形成了基本 BASIC(标准版本)、扩展 BASIC 和多用户 BASIC。70 年代以后,随着微型计算机的迅速发展和普及,微型计算机使用的各种版本相继问世。

1982 年,IBM 公司在推出 PC DOS 时,相继推出了磁盘 BASIC 和 BASICA(Advanced BASIC),Microsoft 公司在推出 MS DOS 时,也公布了 GWBASIC 和编译 BASIC 两种版本。

1985 年,BASIC 的创始人约翰·凯梅尼和托玛斯·库尔兹推出了 True BASIC,开创了 BASIC 语言的新篇章。1986 年,Microsoft 公司推出 Quick BASIC,Borland 公司推出了 Turbo BASIC。它们吸收了 FORTRAN、PASCAL 和 C 等语言的优点,BASIC 呈现出崭新的面貌。

90 年代,Microsoft 公司开发的 WINDOWS 界面获得了空前的推广,与之相应的 BASIC 语言出现了 Visual BASIC,向集成环境和智能化方向前进了一步。到目前为止,BASIC 语言在科学计算、工程绘图、实时控制、文学处理及广告游戏等方面均有很强的功能。

1.2 GWBASIC 语言的特点

GWBASIC 是一种功能十分完善的 BASIC 语言,它的前身是 BASICA。该版本有以下特点:

- (1)完全保留了 BASIC 语言易学、易懂、易于编程等优点。
- (2)有丰富的功能,包括计算、绘图、音乐、控制、动画及与其他语言的接口等。
- (3)支持使用汉字。
- (4)对环境的适应性较好,对硬件要求不高。

1.3 GWBASIC 的运行环境

1. 硬件环境

- (1)64k 以上内存存储器;
- (2)至少一台软盘驱动器;

- (3)一台彩色图形显示器;
- (4)80286 以上主机;
- (5)与系统有关的外部设备。

2. 软件环境

DOS 2.1 以上各版本,包括 MSDOS、UCDOS 等。

1.4 GWBASIC 的启动

启动 GWBASIC 的完整命令格式为:

GWBASIC[<文件名说明>][<<输入文件名>][>][><输出文件名>][/F:<文件数>][/S:<缓冲区>][/C:<通信缓冲区>][/M:<工作空间>[,<用户保留空间>]][/D]

各选项说明:

(1)<文件名说明>是启动 BASIC 的同时,从磁盘上装入并立即执行的一个文件说明。它包括文件所在的盘符:目录路径及文件名。其规定必须与 DOS 中的规定一致。

(2)<<输入文件名>是程序执行时提供数据的文件名。这种提供数据的方式改变了由键盘输入的常规形式,因此叫输入重定向,以区别于来自键盘的标准设备输入。缺省时为标准输入。

(3)><输出文件名>也叫输出重定向,即程序执行的结果由输出语句输出到该文件。标准输出是向屏幕写信息。当该文件名前有两个“>”时,则以补加形式输出到该文件。一个“>”将清除原有的信息。缺省时为标准输出。

以上两个重定向文件必须在开关参数的前面。

(4)/F:<文件数>开关参数,控制 GWBASIC 程序执行期间可在任一时刻打开文件的最多个数。缺省值为 3,最大值为 15,实际可打开的文件个数还取决于 DOS 中 CONFIG.SYS 文件中的配置情况。该文件中如果未指定“FILES=”项,则内定为 8,此时/F:4 为最大值。

(5)/S:<缓冲区>是随机文件使用的缓冲区大小。当用 OPEN 语句打开随机文件时,记录长度不应超过该值。内定隐含缓冲区大小为 128 个字节。

(6)/C:<通信缓冲区>为异步通信接口设置的缓冲区大小。用户可设置成 32767 个字节,内定为 256 个字节。如果设置成/C:0,则失去支持通信功能。

(7)/M:<工作空间>为 BASIC 设置的工作空间字节数,内定值为 64k,用户设置的最大值也为 64k。

<用户保留空间>的作用是:当企图在 BASIC 工作区间装入程序时,启动 BASIC 时应选此参数开关。单位为段,一段是 16 个字节。系统内定值为 4096,实际为 BASIC 的数据区和栈工作区分配 $4096 \times 16 = 65536$ 个字节的工作区间。

(8)/D 为计算双精度函数开辟附加工作区间(约 3000 个字节),可进行双精度计算的函数有 ANT、COS、EXP、LOG、SIN、SQR 和 TAN。如果缺省/D,将不考虑进行双精度函数的计算,并释放附加工作区间。

例如:

GWBASIC B:\GW\PR1>PRN

该命令在启动 GWBASIC 的同时,将 B 盘上 GW 目录下的文件 PR1.BAS 装入内存并执行,

输出结果送打印机。

1.5 BASIC 的使用方式

BASIC 有两种使用方式,一为直接方式;二为间接方式。

1.5.1 直接方式

直接方式也叫交互方式。它是在 BASIC 启动后,从键盘上输入 BASIC 语句或命令并立即执行的一种操作方式。这种方式使用的语句和命令前面不允许使用行号,在临时计算一些表达式时使用,因此又叫计算器方式。

例如:

```
C>GWBASIC
OK
PRINT EXP(8.2)
```

1.5.2 间接方式

间接方式也叫程序方式,是将计算和处理的问题用一个语句序列(程序)实现。这种方式下,每一个语句均有行号,并用 RUN 命令启动程序执行。

例如:

```
OK
10 PRINT SIN(0.417)
20 PRINT COS(0.417)
RUN
```

OK

1.5.3 BASIC 的退出

BASIC 退出到 DOS 状态使用的命令为:

```
SYSTEM
```

该命令可在程序中写入,也可由键盘直接输入。当命令执行时,可立刻退回到 DOS 命令状态,并关闭所有被打开的文件。

如果启动 BASIC 的同时将执行程序装入,由于未直接进入 BASIC 控制状态,故文件执行后,自动返回 DOS 命令状态。

第 2 章 BASIC 的词法规则

2.1 BASIC 字符集

BASIC 的字符集由字母、数字和专用字符组成。这些字符都是 BASIC 系统能够识别和解释的符号。

一、英文字母

①英文的大写字母 A、B、C、…、Z；

②英文的小写字母 a、b、c、…、z。

当用英文的小写字母时，将自动转换成大写字母。

二、数字

从 0 到 9 共 10 个。

三、专用字符

专用字符是在 BASIC 系统中有特殊含义和用途的字符。它们是：

①数学表达式中的运算符()、+、-、*、/、^、\；

②关系比较符号<、>、=；

③标志符号!、#、%、\$、;、,、?、:、.、'、"、_。

除上述有特殊含义的字符外，BASIC 对一些 ASCII 代码字符可以放在显示和打印的位置进行显示和打印输出。

2.2 BASIC 保留字

在 BASIC 中，有些字符的组合具有专门的意义，称为 BASIC 保留字。BASIC 保留字包括 BASIC 中的语句名、内部函数名、内部变量名和 BASIC 命令及某些特殊运算符。

一、BASIC 语句保留字

BEEP	COM	CALL	COMMON
CHAIN	DATA	CIRCLE	DATE
CLOSE	DEF	CLS	FN
COLOR	SGN	USER	LET
DIM	LINE	DRAW	END
LOCATE	ERASE	LPRINT	ERROR
LSET	FOR	RSET	NEXT
USING	GET	MID\$	GOSUB
MOTOR	RETURN	ON	GOTO
PEN	IF	PLAY	THEN
STRIG	ELSE	TIMER	INPUT

OPEN	INPUT #	OPTION	BASE
KEY	OUT	PAINT	POKE
PRINT #	WRITE	PRINT	WRITE #
PSET	DEFSTR	PRESET	DEFINT
DEFDBL	PUT	DEFSGN	RANDOMIZE
READ	REM	RESTORE	RESUME
RETURN	SOUND	STOP	SWAP
VIEW	WAIT	WHILE	WEND
WIDTH	WINDOW		

二、BASIC 命令保留字

AUTO	RESET	BLOAD	RMDIR
BSAVE	RUN	CHDIR	SAVE
CLEAR	SYSTEM	CONT	TRON
DELETE	TROFF	EDIT	FILES
KILL	LLIST	LIST	LOAD
MERGE	MKDIR	NAME	NEW
RENUM			

三、BASIC 函数与内部变量保留字

ABS	HEX \$	ASC	INKEY
ATN	INP	CDBL	INPUT \$
CHR \$	INSTR	CINT	INT
COS	LEFT \$	CSNGT	LEN
CSRLIN	LOC	CVI	LOF
CVS	LOG	CVD	LPOS
EOF	ERR	MKI \$	ERL
MKS \$	EXP	MKD \$	FIELD
OCT \$	FIX	PEEK	FRE
PMAP	POINT	VAL	POS
VARPTR	VARPTR \$	RIGHT \$	RND
SCREEN	SGN	SIN	SPACE \$
SPC	SQR	STICK	STR \$
STRIG	STRING \$	TAB	TAN
TIME \$	TIMER	USR	

四、BASIC 的其他保留字

AND	NOT	EQV	OFF
MOD	XOR	OR	

用户不能使用 BASIC 保留字作其他标识名。在使用过程中,必须用空格将保留字与其他成分分隔开。

2.3 BASIC 常数

GW BASIC 的常数有三种类型。

2.3.1 数值常数

1. 整型常数

整型常数用 2 个字节存储,取值在-32768 到 32767 之间不带小数点的数。例如:

0, -32767, 32767, -11111

2. 八进制常数

以 &O 或 & 打头的八进制字符串,2 字节存储。取值范围也必须是十进制的 -32768 到 32767 之间。最多用 6 位八进制数表示。它们是 0、1、2、...7。例如:

&O3476, &O12345

3. 十六进制常数

以 &H 打头的十六进制字符串,2 字节存储。表示范围是十进制的 -32768 到 32767 之间,最多用 4 位十六进制数表示。它们是 0、1、2、...A、B、C、D、E、F。例如:

&H32F, &HE120

八进制和十六进制均不能超过十进制整数-32768 到 32767 取值范围,即十六进制必在 &H8000 到 &HFFFF 之间,八进制必在 &O100000 到 &O017777 之间。例如

&O210000 和 &H7FFF0

尽管它们是八进制和十六进制的表达形式,但由于超过取值范围,均为非法的表示。同样,十进制数 32768 和 -32769 也为非整型数。

4. 单精度常数

有以下三种表达方式:

① 7 位或 7 位以下的数,但不包括整型数-32768 到 32767 之间的整数,例如

-3.1416, 0.0007, 1234567, 99999, 32768, -32769

② 用指数形式表示的数,例如

-1.09E-3, 1.0E+10, 2.E5, .2E-4

③ 尾部带有! 的数,例如

2!, 1234!, 2.1102!, 1.2!

单精度数在内存中占 4 个字节,每个数存放 7 位,仅前 6 位是精确的,输出时可输出全部 7 位。

5. 双精度常数

有以下三种表达方式:

① 8 位或 8 位以上的数字,可以带或不带小数点,例如

123456789, 0.001234079, 3.14159265

② 用指数形式表示的数,指数字母以 D 表示,例如

1.2D+7, 1.234D-9, 0.214D04

③ 尾随一个# 的数,例如

1#, 0.1#, 123456789#, 1.24#

2.3.2 字符常数

字符常数是用双撇号括起来的 ASCII 字符序列,但字符序列中不允许有双撇号。该序列

的字符个数为 0 到 255 个。例如

“HELLO”, “1234”, “Name, sex”, “□”, “”

2.3.3 逻辑常数

逻辑常数只有两个, 一是真(以非 0 表示); 另一为假(用 0 表示)。

2.4 变 量

2.4.1 变量的类型

变量分为三种类型, 即数值型变量、字符型变量和逻辑型变量, 它们与数据类型相匹配。

1. 数值型变量

数值型变量有以下三种使用形式。

(1) 整型数变量 用 % 作为最后一个符号的变量, 例如 A%、YF%、SEX%。

(2) 单精度变量 用 ! 作为最后一个符号的变量, 例如 A!、VAK!、ADDR!。

单精度变量后面的 ! 可省略, 例如 A 和 A!、BDT 和 BDT! 是等值的。

(3) 双精度变量 用 # 作为最后一个符号的变量, 例如 X#、SRTQ#、PI#。

2. 字符型变量

字符型变量是以 \$ 作为最后一个符号的变量, 例如 NAME\$、YER\$、SEX\$。

3. 逻辑型变量

逻辑型变量无单独的标志, 一般可使用单精度变量或整型数变量。因此, 逻辑型变量也可认为是数值型变量的特殊情况。

2.4.2 变量的命名规则

变量的名字可为 1 到 40 个字母、数字和小数点组成的字符序列, 但第一个字符必须是字母。当超过 40 个字符时, 仅前 40 个字符有效。例如

A, ALPHABATA, A123.45, XYO24

变量名不能与保留字同名, 也不得在保留字后加上 \$、%、!、# 等类型说明符。例如, 下面的变量名是非法的:

COS, SIN#, EXP\$, SQR!, ABS

2.4.3 变量类型的说明

变量类型说明有两种方式。

1. 用类型说明符说明

% 说明为整型数变量;

! 说明为单精度变量, ! 可省略;

说明为双精度变量;

\$ 说明为字符型变量。

类型说明符号必须在变量名的末尾, 单独使用无效。

2. 用类型定义语句定义

可用以下类型定义语句某些字母打头的变量：

DEFINT 定义整型变量

DEFSNG 定义单精度变量

DEFDBL 定义双精度变量

DEFSTR 定义字符型变量

变量的类型说明符%、!、#优先于类型定义语句对变量的定义。上面的变量也叫简单变量。

2.5 下标变量

下标变量是具有下标的变量，它们是数组的相应元素。

2.5.1 数组

1. 数组的概念

数组是与一个名字有关的一组数或一个数值表。数组中每个值叫数组元素。

2. 数组的定义或声明

定义数组是声明数组的名称和类型、设置数组元素的个数(称作数组的体积)及各元素的排列方式。

用数组说明语句定义数组，即

DIM(数组说明表)

其中(数组说明表)为被说明的数组名及其相关参数的列表。一般形式为

(数组名)((表达式表)), (数组名)((表达式表)), ...

(1)(数组名)是对数组的命名。命名的规则、要求及类型标识均同简单变量。

(2)(表达式表)是用逗号隔开的一组数值表达式，定义了相应数组中元素个数及其在数组中的排列方式。

当(表达式表)中有一个表达式时，表示该数组以一维队列的形式排列。元素个数为表达式的值加1。例如：

DIM A%(5)

定义整型数组A%有6个元素，按一维队列的形式排列。

当(表达式表)有两个用逗号分开的表达式时，表示该被说明的数组各元素以二维表的形式排列。第一个表达式是表的行，第二个表达式是表的列。每一维的元素个数(维的长度)是该维方向上表达式的值加1。因此，二维数组的元素个数为两个表达式的值加1之后的乘积。例如，

DIM XY(4,4)

说明语句定义了一个单精度实型数组XY。这是一个有5行、5列的二维表形式，共有元素25个。

当(表达式表)有三个用逗号分开的表达式时，表示该被说明的数组各元素以三维空间形式排列。第一个和第二个表达式组成一个二维表，第三个表达式是该二维表的厚度。例如

DIM GW(4,3,2)

语句定义 GW 数组为单精度实型数组。元素的体积为 $(4+1) * (3+1) * (2+1)$ ，共有元素 60 个。

2.5.2 下标变量

下标变量就是数组中各元素。数组维数不同，下标变量也有区别。

1. 一维下标变量

一维下标变量的表达形式是：

〈数组名〉(〈下标表达式〉)

其中〈数组名〉应含有类型定义符。〈下标表达式〉说明该元素在数组中的存放位置号。GWBASIC 规定，一维数组中的元素，下标表达式的值即为该元素的位置。值越小，序号越在前。0 序号元素是第一个元素。例如，

DIM A(3)

各元素的排列如图 2—1 所示。

2. 二维下标变量

二维下标变量的表达形式是：

〈数组名〉(〈下标表达式 1〉,〈下标表达式 2〉)

其中〈数组名〉应含有类型说明符。〈下标表达式 1〉和〈下标表达式 2〉反映该元素所在的行和列。GWBASIC 规定，二维数组中各元素按行优先的顺序排列。例如，

DIM X(3,3)

各元素的排列位置如图 2—2 所示。

A(0)
A(1)
A(2)
A(3)

图 2—1

X(0,0)	X(0,1)	X(0,2)	X(0,3)
X(1,0)	X(1,1)	X(1,2)	X(1,3)
X(2,0)	X(2,1)	X(2,2)	X(2,3)
X(3,0)	X(3,1)	X(3,2)	X(3,3)

图 2—2

3. 三维下标变量

三维下标变量的表达形式为：

〈数组名〉(〈下标表达式 1〉,〈下标表达式 2〉,〈下标表达式 3〉)

例如：

DIM B(3,2,2)

各元素的排列位置如图 2—3。

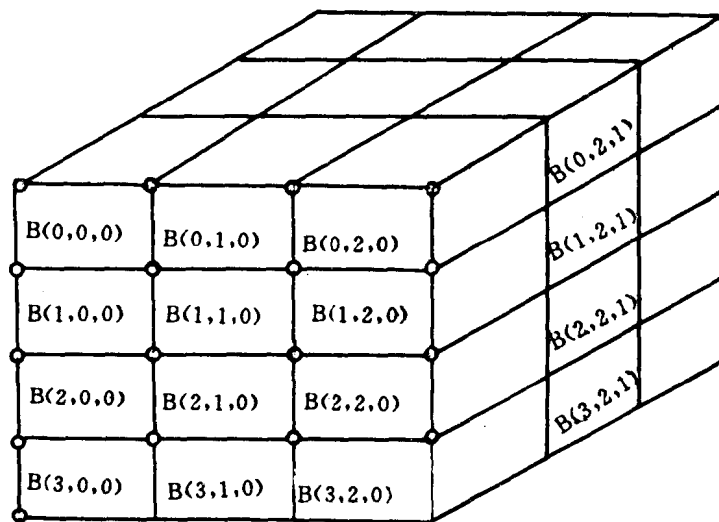


图 2-3

2.5.3 下标变量的有关规定

① 下标表达式的个数(数组的维数)规定为 1 到 255。每一下标表达式的取值是 0 到 32767,但具体取值受内存限制。

② 当一维数组的下标变量的下标表达式最大值不超过 10 时,可以不经定义而直接使用下标变量。

③ 如未经说明,下标表达式的最小值是零。

④ 下标变量和简单变量可以同名。

⑤ 下标值从 0 开始,由 OPTION BASE 语句设置后可从 1 开始。

使用格式为:

OPTION BASE n

其中 n 取 0 或 1,表示下标的最小值。如不用该语句,则下标的最小值是 0。

⑥ 数组定义之后,有可能占用大量内存空间,不用时应及时释放,此时可用如下语句:

ERASE<数组名>[,<数组名>]...

该语句将给定数组占用的空间释放。

2.6 BASIC 函数

2.6.1 内部函数

内部函数也叫系统函数。内部函数有数值计算函数、字符串函数及软硬件操作函数。内部函数的名称、自变量要求及含义等见附录 A。本节仅介绍几种常用的数值函数及字符串函数。

1. 数值函数

(1) 常用的三角函数

(a) SIN 函数

使用格式:

$$v = \text{SIN}(x)$$

功能: 计算自变量 x 的正弦值。

说明: x 为实型表达式, 并以弧度表示。

(b) COS 函数

使用格式:

$$v = \text{COS}(x)$$

功能: 计算自变量 x 的余弦值。

说明: x 为实型表达式, 以弧度表示。

(c) TAN 函数

使用格式:

$$v = \text{TAN}(x)$$

功能: 计算自变量 x 的正切。

说明: x 为实型表达式, 以弧度表示。

(d) ATN 函数

使用格式

$$v = \text{ATN}(x)$$

功能: 计算自变量 x 的反正切值。

说明: x 为任何类型的数值表达式, 结果为单精度值, 并给出弧度表示的结果, 其范围为 $-\pi/2$ 到 $\pi/2$ 。

(2) 常用的数值函数

(a) ABS 函数

使用格式:

$$v = \text{ABS}(x)$$

功能: 取 x 的绝对值。

说明: x 为任何类型的数值表达式。

(b) SQR 函数

使用格式:

$$v = \text{SQR}(x)$$

功能: 计算 x 的平方根。

说明: x 的任何类型的数值表达式, 但其值必须大于零。

(c) EXP 函数

使用格式:

$$v = \text{EXP}(x)$$

功能: 计算 x 的指数 e^x 。

说明: x 为任意形式的数值表达式, 但其值必须小于、等于 88.02969, 否则将产生溢出错误。

(d) LOG 函数

使用格式