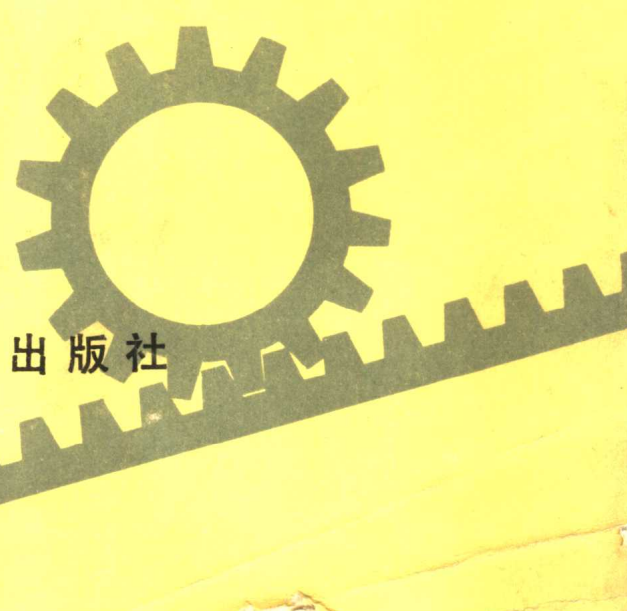


机械零件设计

BASIC 程序

成都科技大学校办总厂微机应用研究室编著



成都科技大学出版社

机 械 零 件 设 计

BASIC 程 序

马济永 罗敬生 谢国光
魏巨炎 宿泽君 王捷坤 冯永钦 编著

成 都 科 技 大 学 出 版 社

内 容 简 介

本书比较详细地介绍了有关数据、表格和线图的处理方法；机械零件程序设计步骤和方法；常用机械零件设计通用程序的编制和使用，附有零件设计系统框图、程序清单和例题。

全书分为 BASIC 语言、资料表格和线图的处理方法、带传动、链传动、圆柱齿轮传动、圆锥齿轮传动、蜗杆传动……传动轴、弹簧、轴承等十三章。

本书所提供的各种零件设计通用程序，由四川省高教局组织了鉴定，获得成都市1985年科技进步奖和四川省1986年优秀软件奖。它可用作设计或校核，并可以移植到各种微机上使用。设计内容包括几何尺寸计算；强度、刚度和稳定性校核；尺寸公差和光洁度选择等。

此书可供从事机械设计或修理的工程技术人员使用，也可供高等工业院校机械类专业的师生参考。

机 械 零 件 设 计

BASIC 程序

成都科技大学校办总厂微机应用研究室 编著

成都科技大学出版社出版

四川省新华书店 发行

四川省崇庆县印刷厂印刷

开本787×1092 1/16 印张20 6/16

1986年12月第1版 1986年12月第1次印刷

字数480千 印数1—4,000

书号：15475·12

定价：4.90元

目 录

第一章 BASIC 语言.....	(1)
1—1 概述.....	(1)
1—2 基本词法.....	(4)
1—3 赋值和输入输出语句.....	(12)
1—4 注释和说明语句.....	(19)
1—5 简单控制语句.....	(21)
1—6 复合语句.....	(23)
1—7 字符串.....	(31)
第二章 设计资料图表的处理.....	(36)
2—1 概述.....	(36)
2—2 计算式的处理.....	(37)
2—3 标准数据和资料表格的处理.....	(37)
2—4 实验数据表格的处理.....	(41)
2—5 计算线图的处理.....	(48)
2—6 数据文件.....	(51)
2—7 优化计算.....	(52)
第三章 三角胶带传动.....	(55)
3—1 三角胶带传动的设计系统.....	(55)
3—2 三角胶带设计资料的系统化.....	(61)
3—3 三角胶带传动通用设计系统框图.....	(65)
3—4 三角胶带传动设计的源程序.....	(67)
3—5 三角胶带传动设计的操作说明和实例.....	(68)
第四章 套筒滚子链传动.....	(71)
4—1 套筒滚子链传动设计内容.....	(71)
4—2 滚子链传动设计资料的处理.....	(76)
4—3 滚子链传动设计系统框图及源程序.....	(78)
4—4 滚子链传动设计程序操作说明及实例.....	(79)

第五章	渐开线圆柱齿轮传动	(80)
5—1	渐开线圆柱齿轮传动设计	(80)
5—2	渐开线圆柱齿轮传动设计资料的处理	(103)
5—3	渐开线圆柱齿轮传动设计程序	(107)
5—4	圆柱齿轮传动设计程序操作说明及实例	(110)
第六章	正交螺旋齿轮传动	(112)
6—1	正交螺旋齿轮传动设计	(112)
6—2	正交螺旋齿轮传动设计资料的处理	(115)
6—3	正交螺旋齿轮传动设计程序	(118)
6—4	正交螺旋齿轮传动设计程序操作说明及实例	(119)
第七章	渐开线直齿圆锥齿轮传动	(120)
7—1	直齿圆锥齿轮传动的设计系统	(120)
7—2	直齿圆锥齿轮传动设计资料的系统化	(125)
7—3	直齿圆锥齿轮传动通用设计系统框图	(130)
7—4	直齿圆锥齿轮传动设计的源程序	(131)
7—5	直齿圆锥齿轮传动设计程序的操作说明及实例	(132)
第八章	蜗杆传动	(136)
8—1	蜗杆传动的设计系统	(136)
8—2	蜗杆传动设计资料的系统化	(147)
8—3	蜗杆传动通用设计系统框图	(150)
8—4	蜗杆传动设计的源程序	(152)
8—5	蜗杆传动设计程序的操作说明和实例	(153)
第九章	丝杠螺母传动	(156)
9—1	丝杠螺母传动的设计系统	(156)
9—2	丝杠螺母传动设计资料的系统化	(161)
9—3	丝杠螺母传动通用设计的系统框图	(163)
9—4	丝杠螺母传动设计的源程序	(166)
9—5	丝杠螺母传动设计程序的操作说明和实例	(166)
第十章	传动轴	(169)
10—1	传动轴的设计系统	(169)
10—2	传动轴设计资料的系统化	(177)
10—3	传动轴通用设计系统框图	(180)
10—4	传动轴设计的源程序	(180)
10—5	传动轴设计程序的操作说明及实例	(182)

第十一章 圆柱压缩、拉伸螺旋弹簧..... (185)

11—1 圆柱压缩、拉伸螺旋弹簧的设计系统.....(185)

11—2 弹簧设计资料的系统化.....(191)

11—3 弹簧设计系统框图及源程序.....(193)

11—4 弹簧设计程序的操作说明和实例.....(196)

第十二章 螺栓联接..... (199)

12—1 重要螺栓组联接的设计系统.....(199)

12—2 螺栓联接设计资料的系统化.....(201)

12—3 螺栓联接通用设计的系统框图.....(203)

12—4 螺栓联接设计的源程序.....(205)

12—5 螺栓联接设计程序的操作说明和实例.....(205)

第十三章 滑动轴承..... (207)

13—1 滑动轴承的设计系统.....(207)

13—2 滑动轴承设计资料的系统化.....(211)

13—3 滑动轴承设计系统框图.....(211)

13—4 滑动轴承设计的源程序.....(213)

13—5 滑动轴承设计程序的操作说明和实例.....(213)

附录

A—1 三角胶带传动设计程序清单..... (1)

A—2 三角胶带传动设计例一..... (9)

A—3 三角胶带传动设计例二..... (10)

A—4 三角胶带传动设计例三..... (11)

B—1 套筒滚子链传动设计程序清单..... (12)

B—2 套筒滚子链设计实例..... (14)

C—1 渐开线圆柱齿轮传动设计程序清单..... (15)

C—2 渐开线圆柱齿轮传动设计实例..... (41)

D—1 正交螺旋齿轮传动设计程序清单..... (44)

D—2 正交螺旋齿轮传动设计实例..... (47)

E—1 直齿圆锥齿轮传动设计程序清单..... (48)

E—2 直齿圆锥齿轮传动设计例一..... (57)

E—3 直齿圆锥直轮传动设计例二..... (58)

F—1 蜗杆传动通用设计程序清单..... (59)

F—2 蜗杆传动设计例一..... (67)

F—3 蜗杆传动设计例二..... (68)

G—1 丝杆螺母传动通用设计程序清单..... (69)

G—2 丝杆螺母传动设计例一..... (74)

G—3	丝杆螺母传动设计例二.....	(75)
H—1	传动轴设计清单.....	(76)
H—2	传动轴设计实例.....	(85)
I—1	圆柱螺旋弹簧设计程序清单.....	(86)
I—2	圆柱螺旋弹簧设计例一.....	(91)
I—3	圆柱螺旋弹簧设计例二.....	(92)
J—1	螺栓联结设计程序清单.....	(93)
J—2	螺栓联结设计实例.....	(96)
K—1	滑动轴承设计程序清单.....	(97)
K—2	滑动轴承设计实例.....	(99)
	参考文献.....	(100)

第一章 BASIC语言

1—1 概述

一、计算机的语言

要使计算机按人的意图工作，就必须使计算机懂得人的意图，能够接受人向它发出的命令和信息。人和机器交换信息就要解决“语言”的问题。计算机并不懂得人的语言，它只能识别0和1两种状态，所以它只能识别由0和1组成的指令和信息。

Y E E—8100微型机采用Z—80中央处理器，具有8位字长，因此它的指令和信息是由8个二进制数（0或1）所组成的代码，作为它的机器语言或机器指令。例如有一程序要求从一个输入通道接收8位字长的数据，然后将其送到某个输出通道。完成这一操作程序的第一条指令是输入指令，它的机器语言编码是：

11011011

紧接着信息是由256个通道中的那一个输入。例如从一号通道输入，它的代码是：

00000001

第2条指令是输出指令。输到3号通道应该有下面的代码：

11010011

00000011

每项指令和信息都需要一个存贮单元，上述指令可存于0000H至0003H单元。

由上可见，用机器语言编写程序是一件十分繁琐的工作，要记住这些指令代码和含义是不容易的。为了便于记忆和书写，人们常用汇编语言来编写程序。汇编语言不同于机器语言，其特点是：（1）、用记忆符（操作码和操作数）来表示指令，（2）、用标号代替存贮器地址。如将机器语言中的例子改成汇编语言，则写成如下程序：

标号	记忆符	操作数	注解
NAME	IN	A, (01)	INPUT FROM PORT 1
	OUT	(03), A	OUTPUT TO PORT 3
	JP	NAME,	KEEP GOING (继续)

由此程序可看出汇编语言程序的每一句是由四部分组成。它们是（1）标号，（2）记忆符（或操作码），（3）操作数，（4）注解段。这些语句比机器语言指令好理解和记忆，便于书写程序。

由汇编语言编写的程序，执行时必须先将其翻译成机器语言。这个翻译过程是由一个翻译程序来完成。将汇编语言转换成为机器语言的翻译程序叫汇编程序。

在Z—80指令系统中，有74种记忆符，25个操作数关键字，有693种操作码和操作数的合法组合。

汇编语言使用记忆符比机器语言易于理解和记忆，但它仍不能直接用来表达数学式，如 $C = A + B$ 。因此，人们一直在设想找到一种比较接近于人们习惯的“数学语言”，而又能为机器所接受的过渡性语言。在1956年人们终于创造出“程序设计语言”，又叫“算法语言”，它比较接近于人们的习惯语言和数学语言。它允许用英文写解题的计算程序，所用的运算符号和运算式子也和数学式差不多，例如 `LET C=A+B`；`PRINT "C=" C`，计算机都能接受，并能将结果打印出来。

使用“算法语言”来编写计算程序就显得极为方便。人们就可以完全不管什么机器指令，也可以不懂得计算机的内部结构和存贮单元，就能得心应手地使用计算机进行各种设计计算和事务管理了。使用算法语言编写的程序，对于不同的计算机来说，它具有一定范围的通用性。算法语言又称高级语言。目前国内外比较通用的算法语言有十几种，适用范围也不同。其中最常用的有：

<code>FORTRAN</code>	(适合于科技计算)	<code>BASIC</code>	(小型会话式语言)
<code>ALGOL 60</code>	(适合于科技计算)	<code>PASCAL</code>	(结构程序计算语言)
<code>COBOL</code>	(适合于数据处理)		

用算法语言编写的计算程序，称为源程序。源程序并不能直接在计算机上执行。计算机归根到底还是只能直接执行机器语言程序，机器语言又叫目标程序，或目的程序。因此必须把源程序翻译成目标程序。这种翻译有两种方法：编译方法和解释方法。

`FORTRAN`、`ALGOL`等算法语言均采用编译方法。就是事先编好一个叫编译程序的机器语言程序，将其放入计算机中；再把源程序输入计算机，由编译程序把源程序整个地翻译成目标程序，然后执行目标程序，得到计算结果。如图1—1所示。



图 1—1 编译方法

`BASIC`算法语言则采用解释方法。就是事先编好一个叫解释程序的机器语言程序，将其放入计算机中，再把源程序输入计算机中。执行时，它逐句地翻译执行，即边翻译边执行。如图1—2所示。

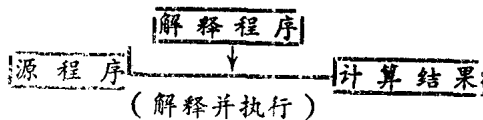


图 1—2 解释方法

二、BASIC 语言的特点及构成

1、BASIC 语言的基本特点

`BASIC`语言是目前国际通用的计算机算法语言。由于它具有简单易懂和采用解释方法比较节省内存等特点，对微机比较有利（微型机的内存容量较小），故绝大多数的微型机都配有`BASIC`语言。

`BASIC`是 `Beginner's All—Purpose Symbolic Instruction Code`

(初学者通用符号代码)的缩写。BASIC系统常用的有单用户基本 BASIC, 扩展 BASIC和多用户分时BASIC。

YEE—8100和TRS—80微型机所用的BASIC语言分两级, 它是经过扩展的单用户BASIC, 即BASIC I 语言, 它比基本BASIC功能强得多; 带磁盘驱动器系统使用的DISK BASIC, 它是对BASIC I 的进一步扩展, 主要增加了字符串处理, 格式输出和磁盘文件管理功能。APPLE II 微型机CP/M操作系统下所用的MBASIC, 即BASIC—80, 它与DISK BASIC相近, 其GBASIC更增加了高清晰度绘图功能。IBM PC BASIC和GBASIC非常接近。

BASIC语言具有以下几个特点:

(1)、BASIC语言比较简单。它在语句和命令中所用的词及运算符号, 都与英语中的词和数学中的运算符差不多, 故比较直观, 易于理解和记忆。例如READ的意思是读; “END”的意思是结束; “>”是大于。

(2)、BASIC语言具有人机对话性能。当源程序输入计算机执行时, 计算机立即检查程序, 如果句法有错立刻由显示器显示出来, 可以在键盘上修改程序。修改后的程序执行时, 计算机又继续检查程序, 这样边算边改, 直到得出满意结果, 十分方便。零件设计时, 用它显示数据范围、输入已知条件, 特别灵活方便。

(3)、BASIC语言允许直接键盘运算, 不必专门编写程序, 如同使用计算器一样。这对程序的调试检查特别有用。

(4)、扩展BASIC有较强的字符串处理功能; 特殊函数和语句, 用于小型数据处理辅助设计, 事务管理和情报检索方面是很方便和灵活的。

(5)、BASIC是一种小型的算法语言。它的数值范围, 变量的数量, 以及自定义函数等都有一定限制, 故不宜用作大型题目。

(6)、BASIC语言是解释性语言, 要占用较多的机器时间, 降低了解题的速度。

2、BASIC程序的构成

今用一实例说明什么是BASIC程序及其构成规则。

已知梯形的上底 $a=3.5$ 米, 下底 $b=4.2$ 米, 高 $h=1.8$ 米, 试求梯形的面积。梯形面积 S 的计算公式为 $S=(a+b)h/2$ 。

用BASIC I 语言编成下面一段程序

```
10  A=3.5: B=4.2: H=1.8
20  S=(A+B)*H/2
30  PRINT "S=" S
40  END
```

这种利用BASIC语言编的解题程序, 称为BASIC语言源程序。由此可以看出:

(1)、一个BASIC程序是由若干行组成的。BASIC程序中, 一行允许写多个语句, 但是, 必须用“:”号隔开。每一个语句分别让计算机执行某一方面的功能。因此, 程序是由若干语句所组成, 上例中包括6个语句。

第40行是显示出 S 的值来。在BASIC中, 语句的长度不能超过255个字符。

(2)、每一个语句可分为三部分:

行号

语句定义符

语句体

每行起始的语句前都冠有数字,称为行的标号或行号。行号必须是0~65529之间的任一整数,但计算机的BASIC解释系统会把源程序按标号大小顺序排列好,从小到大顺序执行。标号不一定要连续,留上间隔以便在修改程序时增添新语句。

语句定义符,它是一些单词,用它来表示语句的性质。例如PRINT表示显示,LET表示赋值。BASIC中规定,LET可省去不写,上例第10和20行就没有写LET,这些单词则由字符构成。

语句体,即跟在语句定义符后面的、需要执行的具体内容。例如 $A=3.5$, $S=(A+B)*H/2$ 。它也是由字符所构成。

(3)、通常程序的最后是以END结束。但是BASIC程序也可以没有END语句。在这种情况下,也同有END一样,将显示READY进入待命状态。

综上所述,构成程序的最基本的单位是字符。由字符组成行号、语句定义符和语句体,再由它们构成各个语句。各个语句组成程序行,再由各个程序行构成整个程序。

上述程序可以通过键盘输入计算,或者由已存有上述程序的磁带或软磁盘输入计算机。输入后就存在计算机内,计算机并不立即执行此程序。必须由使用者发出“运行”的命令RUN并按下NEW LINE键(APPLE II则按下RETURN键),计算机才执行此程序。此时计算机按照行号从小到大顺序执行上述各语句,执行后显示屏显示:

YEE8100机		APPLE II机
S=6.93		S=6.93
READY	或	OK
>—		□(光标)

这就是所要求的梯形面积。数字下方的“READY”和“>—”或“OK和□”表示计算机处于命令状态等待工作。再打入运行命令RUN,按NEW LINE键(或RETURN键),则再次执行此程序。如果要把计算结果打印在行式打印机的纸上,只要在PRINT前加上L(即LPRINT)即可以打印出S值。

由上可见,BASIC有一套严格的词法、句法和操作方法。只有学习和掌握这些法则,才能用BASIC语言编写出零件设计程序、绘图程序和上机操作。

1—2 基本词法

一、BASIC语言的基本符号

任何一种语言都有它自己的基本符号。例如,英语的基本符号由二十六个字母及一些标点符号组成的。BASIC语言是应用ASCII(美国信息交换代码)码的字符集作为基本符号,它由英文字母、数字和一些专用符号所组成。在编写BASIC源程序时,不允许使用其它符号。

- 1、英文字母(26个): A—Z只能用大写
- 2、数字(10个): 0—9
- 3、标点符号(8个): . (小数点), : (冒号), ; (分号), , (逗号), ((左圆括号),) (右圆括号), " (引号), □ (空格)
- 4、算术运算符(5个): + (加), - (减), * (乘), / (除), ↑ (乘

幕, 也有用^符号)

5、关系运算符(6个): <(小于), >(大于), =(等于), <=(小于等于), >=(大于等于), <>(不等于)

6、逻辑运算符(3个): AND(与), OR(或), NOT(非)

7、类型说明符(4个): %(整型), ! (单精度型), # (双精度型), \$(字符串型)

8、指数符号(2个): D(双精度), E(单精度)

9、字符串运算符(7个): +(联结), <(居前), >(居后), =(等于), <=(居前等于), >=(居后等于), <>(不等)

10、控制符号(5个): →, ←, ↑, ↓, @

11、缩写符号(3个): ? (代替PRINT), ' (代替REM), . (代替刚打进的行号)

二、常量

BASIC语言的常量指的是常数、逻辑值和字符串常量。在程序运行过程中, 常量的值不变。

BASIC语言中的常数一律采用十进制。常数可分为整型、单精度型和双精度型。

数的类型	有效数字	例子
整型	-32768~+32767间的整数	5001, -30
单精度	$10^{-38} \sim 10^{38}$ 之间的实数输出6位有效数字	1.3, -23.2756
双精度	$10^{-38} \sim 10^{38}$ 之间的实数输出16位有效数字	-300.123456789

单精度和双精度又分定点和浮点两种表示方法。

定点数只包括数字, 小数点和正负号, 正号一般可省略。绝对值太大或太小的常数, 可写成浮点数形式。浮点数是在定点数后加指数 $E \pm e$ (单精度) 或加指数 $D \pm e$ (双精度), E和D都表示10的方次e是一位或两位整数。 $E \pm e$ 或 $D \pm e$ 表示乘上 10^e 。指数部分的正号可省略。例如:

123 = 1.23×10^2	1.23 E 2
-1149.54	-1.14954 E 3
0.00051	5.1 E -4
230,000,000 (不能输入)	23 E 7

逻辑值只有两个, 即真(用1表示)和假(用0表示)。

有关字符串常量见后述。

三、简单变量

变量和数学中变量概念一样, 变量是指程序运行时, 可以改变其取值的量。

变量有两种形式: 简单变量和下标变量。

1、简单变量名

简单变量名用英文字母A~Z中任一个字母来表示, 也可以用一个字母后跟一个数字或后跟一个字母来表示。例如, A、A3、BC等都是变量名, 而1Z、β、4等名字

则不合法，机器将拒绝接受。

简单变量名可以多于两个字符，如SUM、DELN、SUB等。但计算机只认开头两个字符，SUM和SUB被看作同一个变量SU。

简单变量名不能包含BASIC的专用名词（见计算机使用手册附录）。例如ON和AND等。

2、简单变量的类型

BASIC语言的简单变量有四种类型：整型；单精度型；双精度型和字符串型。前三种用来存贮具有不同精度的数字常量，字符串变量用来存贮字符串常量。变量类型用变量名后加一个类型说明符来表示，如下所示：

变量类型	整型	单精度型	双精度型	字符串型
说明符	%	!	#	\$
例	A%, C1%,	A!, BD!	B#, K1#	B\$, WR\$

不同的变量类型可以使用相同的变量名，只要它们具有不同的类型说明符就行了。例如A%，A!和A#是不同的变量。没有类型说明符的变量或没有用定义变量类型语句所定义的变量均作为单精度变量处理。

四、数组和下标变量

1、数组

数组是类型相同，按一定方式排列的一组常量的集合。根据集合中元素的排列不同，数组又分为一维数组、二维数组、三维数组。一维数组中各元素排列成一个有限的序列，二维数组的元素排成一矩阵，三维数组则排成长方体例如：

一维数组 A: 2, 3, 21, 1, 1.423, 5.1243579

20 30 26

30 20 25

二维数组 B: 25 50 20

46 15 10

35 15 12

100 110 120 160 170 180

130 140 150 190 200 210

三维数组 D:

220 230 240 280 290 300

250 260 270 310 320 330

BASIC语言规定，任何变量名都可以用作数组名。

数组有四种类型，即整型、单精度型、双精度型和字符串型。数组的类型是根据数组内元素的类型来定的。例如上述数组A是一个一维的双精度数组，B是一个二维的整型数组，D则是三维整型数组。

2、下标变量

构成数组的各个数或变量，称之为数组元素或下标变量。在BASIC语言中，下标

$$\begin{array}{lll} A\#(0) = 2, & A\#(1) = 3, & A\#(2) = 21, \\ A\#(3) = 1, & A\#(4) = 1.423 & A\#(5) = 5.1243579 \end{array}$$

数组名	类型说明	下标	
	A # (K)		$0 \leq K \leq 5$

$B\% (0, 0) = 20$	$B\% (0, 1) = 30$	$B\% (0, 2) = 26$
$B\% (1, 0) = 30$	$B\% (1, 1) = 20$	$B\% (1, 2) = 25$

数组名	类型说明	行号、列号	
_____↑ ^B	_____%	(I, J)	$0 \leq I \leq 4$
			$0 \leq J \leq 2$

式中D为数组名，X为类型说明符， $E_i (1 \leq i \leq N)$ 是下标。E的个数N叫做维数，前述的数组D有三个下标，称为三维数组。下标可以是常数、变量，也可以是表达式，如D(1, N, M+2)。执行时，下标表达式的值如不是整数，则会自动取成不大于该数的最大整数。

(1)、SIN (X)	Sinx	(3)、TAN (X)	tgx
(2)、COS (X)	Cosx	(4)、ATN (X)	arctgx

7

$\text{COS}47^\circ$ 应为 $\text{COS}(47 * 3.14159 / 180)$ 。函数值是单精度，调用时只能得到六位有效数字。下列其它函数亦如此。

(5)、ABS(X)

绝对值函数，

即 $\text{ABS}(X) = |X|$

(6)、SGN(X)

符号函数，

$$\text{SGN}(X) = \begin{cases} 1 & \text{若 } X > 0 \\ 0 & \text{若 } X = 0 \\ -1 & \text{若 } X < 0 \end{cases}$$

(7)、SQR(X)

平方根，即 $\text{SQR}(X) = \sqrt{X}$ 要求 $X > 0$

(8)、EXP(X)

指数函数，即以 $e = 2.7182$ 为底、 $\text{EXP}(X) = e^x$

(9)、LOG(X)

自然对数，得到 X 的以 e 为底的对数值，即 $\ln X$ 。

(10)、INT(X)

求不大于 X 的最大整数，例如：

$\text{INT}(3.75) = 3$ ， $\text{INT}(-3.5) = -4$

(11)、FIX(X)

向原点取整。当 $X > 0$ 时， $\text{FIX}(X) = \text{INT}(X)$ 。当 $X < 0$ 时，如果 X 本身是一个负整数， $\text{FIX}(X) = \text{INT}(X) = X$ ，如果 X 不是整数， $\text{FIX}(X) = \text{INT}(X) + 1$ 。例如 $\text{FIX}(-3.5) = -4 + 1 = -3$

BASIC语言中的数分为整型、单精度、双精度，有时在运算前需要先转换类型。下面三种类型转换函数为数的类型转换提供了手段。

(12)、CINT(X)

将 X 转换成整数表示，从数值上说 $\text{CINT}(X) = \text{INT}(X)$ 。但是 $\text{INT}(X)$ 是单精度表示，而 $\text{CINT}(X)$ 是整型表示。自变量必须在 -32768 至 32767 之间。

(13)、CSNG(X)

将 X 转换成单精度表示。如果 X 原来是双精度的话，则弃去其最后的二进制的32位尾数，得到一个单精度数。

(14)、CDBL(X)

将 X 转换成双精度表示。为了得到精度较高的计算结果，用函数 $\text{CDBL}(X)$ 来预先提高运算数的精度往往是有效的。

(15)、RND(X)

随机函数。当 $X = 0$ 时，这个函数产生一个在 0 与 1 之间均匀分布的单精度随机数；当 $X > 0$ 时，产生一个在 1 与 X 之间的随机整数。在需要用计算机来模拟某种随机现象的时候，函数 $\text{RND}(X)$ 就很有用。

六、算术表达式

常数、变量或函数用算术运算符和圆括号连接起来就构成表达式。BASIC表达式

和通常的数学表达式很相似。

BASIC表达式

$1/2$

$(-2)^{\uparrow 5}$

$(X-5)*(A+B)$

$X^{\uparrow Y}$

$\text{COS}(3.14159 * L) / \text{LOG}(X)$

数学式

$1/2$

$(-2)^5$

$(X-5)(A+B)$

X^Y

$\text{COS}(\pi L) / \ln X$

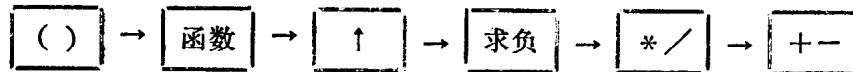
写算术表达式应注意以下几点:

(1)、乘号不能省略, $A * B$ 不能写成 AB

(2)、只允许用圆括号

(3)、不允许两个运算符相邻, $3 * (-2)$ 不能写成 $3 * -2$

表达式中运算顺序为: 括号内优先, 然后再先乘幂($\uparrow$), 再求负($-$), 后乘除($*/$), 最后加减($+-$), 同一级由左向右计算。用图表示:



算术表达式执行结果得到一个数值, 它可与常数、变量一样地用作语句的成分。单个的常数与变量可看着算术表达式的特殊情形。

BASIC允许不同类型的数在一起运算, 并按两个运算数中精度较高的一个数的精度进行。例如 $A! * B \%$, 是按单精度进行, 结果是单精度; $2/3 \#$ 或 $12/CDBI(3)$, 结果是双精度。

" $\uparrow$ " 表示乘方, 在显示或打印时, 换成 $\square$ 或 $\wedge$ 。 $X^{\uparrow Y}$ 即是 X^Y , 这里有一个限制: 当 Y 为非整数时, X 必需为正, 否则无意义。

七、数关系式

算术表达式用数关系运算符连接起来就构成数关系式。即:

(算术表达式) 数关系运算符 (算术表达式)

例如: $X \leq Y$

当 $X \leq Y$ 时, 关系式取真值

$E = F$

当 $E = F$ 时, 关系式取真值

$X^{\uparrow 2} + 1 > 5$

当 $X^2 + 1 > 5$ 时, 关系式取真值

$A + B <> C + D$

当 $(A + B) \neq (C + D)$ 时, 关系式取真值

计算机首先算出关系运算符左右两边的表达式的值, 然后再看这两个值是否满足关系运算符所确定关系。事实上只有满足, 即关系式为真; 或不满足, 即关系式为假两种情况。所以关系式的值是一个逻辑值。

八、字符串

字符串是指用英文字母、数字、空格以及其它基本符号等所组成的一串字符。

在字符串中, 我们以 $\square$ 来表示空格。而在键盘打入空格时则按空格键。

1、字符串常量

字符串常量通常用双引号 " $"$ " 括起来, 以表明字符串常量的起止界限。例如: " $N=? KW$ ", " $HT20-40$ ", " $BELT'S MODEL \& LENGTH:$ "

2 字符串变量

BASIC语言的字符串变量与简单变量类似，可以把整个字符串“赋值”给字符串变量。这种用来存贮字符串常量的变量称为字符串变量。字符串变量与数值变量不同，它不代表数值，而是代表一串可以变化的字符。

字符串变量名是在简单变量后加上字符串说明符而组成。例如，A\$, YJ\$等都是字符串变量名，而A\$=YES, YJ\$=RRRR。BASIC规定，一个字符串变量可存贮255个字符，即字符串变量的长度可达255个字符。

3、字符串型下标变量

BASIC语言的下标变量除有整型、单精度型、双精度型以外，还有字符串型。字符串型下标变量名是在字符串变量名后，加上一个内有相应下标的圆括号所组成。例如T\$(1), J\$(1), LG\$(GR, GC)。同样，下标值从零开始，下标可以是常数、变量和表达式。例如：J\$(0) = "FINISH: 7", J\$(1) = "T-H>60%", J\$(2) = "T-L>65%", LG\$(GR, GC)

"+-19", "+-26", "+-30", "+-36", "+-42", "+-75"
 "+-30", "+-42", "+-48", "+-115"
 "+-48", "+-65", "+-190"

4、字符串相加表达式

字符串相加表达式又称字符串表达式，它是由联结号“+”将字符串常量、字符串变量和字符串下标变量连接起来的式子。字符串相加表达式的运算结果是一个字符串常量。例如：

相加表达式	运算结果
"1982" + "." + "3"	"1982.3"
"SHAFT" + "-K-C-M-X-C"	"SHAFT-K-C-M-X-C"
A\$+B\$(设A\$="SLEEVE", B\$="-C-R-C-M")	"SLEEVE-C-R-C-M"

显然，字符串常量或字符串变量是字符串相加表达式的简单情况。

5、字符串关系式

用字符串关系符(=, <>, <, >, <=, >=)把两个字符串相加表达式连接起来的式子，称为字符串关系式。其一般形式为：

(字符串表达式) 字符串关系符 (字符串表达式)

字符串关系式的值是逻辑值。例如：

字符串关系式	实际含意
W\$="WR"	若W\$等于"WR"，则关系式取真值
Y\$="YES"	若Y\$等于"YES"，则关系式取真值
G\$="ZQSN10-1"	若G\$等于"ZQSN10-1"，则关系式取真值

字符串比较实际是对两个字符串的对应字符的ASCII代码进行比较。BASIC中规定，两个单字符比较，代码数小的字符居前，代码数大的字符居后，只有两个代码数相同时，字符才相等。例如A的ASCII代码是65，B的代码是66，故A居前，因而“A”<“B”。

两个字符串比较时是按照这样的步骤进行，先由左起比较第一个字符，然后由左到