

838961

851

3426

微型计算机在 岩石学和矿物学上的应用

沈步明 从柏林 编著

科学出版社

1
26

351

3426

838961

351
3426

参5087

微型计算机在 岩石学和矿物学上的应用

沈步明 从柏林 编著

科学出版社

1987

内 容 简 介

本书共分八章,内容包括 BASIC 语言的基础知识;程序的调试方法和技巧;岩石学和矿物学上的数据处理方法以及数据处理、数据统计和检验、数据的存取、绘图等 20 多个实用程序,这些实用程序已经构成了一个比较完整的岩石学和矿物学数据处理系统。

本书可供地质、岩石、矿物、矿床等专业的生产、科研人员以及有关院校师生参考。

微型计算机在 岩石学和矿物学上的应用

沈步明 从柏林 编著

责任编辑 谢洪源

科学出版社出版

北京朝阳门内大街 137 号

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1987 年 9 月第 一 版 开本:787×1092 1/16
1987 年 9 月第一次印刷 印张:17 1/2
印数:0001—1,900 字数:400,000

统一书号:13031·3671

本社书号:5178·13—14

定价: 4.15 元

前 言

在地质学领域中,随着科学技术的发展和研究工作的深入,能获得的数据和信息越来越多,不是成倍,而是以数十倍速度增长。这么多的数据和信息,靠人工处理、靠脑子记忆是完全不可能的。因此,利用计算机,特别是利用微型计算机进行数据处理、模拟、存贮和检索,即地质科学中的电脑化,将给地质科学的发展带来巨大的活力,产生深远的影响。在工业发展的国家,微型计算机已经相当普及。近几年来,国内也引进了不少微型计算机,在许多地质研究单位,甚至一些地质大队或中心实验室都已有了微型计算机。但到目前为止,尚无一本面向地质工作者的有关微型计算机的书籍。本书利用 BASIC 语言,编制了岩石化学数据处理, Rittmann 岩石分类, 矿物晶体化学式和矿物端员分子计算, 地质温度计和地质压力计计算, 晶胞参数最小二乘法修正, 计算和核对矿物 X 光衍射线的指标化, 数据的统计计算和检验, 相关系数计算, 多元线性回归参数计算, 点群分析, 绘图, 数据存贮等 20 多个常用程序, 构成了一个比较完整的岩石和矿物数据处理系统。这个系统的大部分程序都已在中国科学院地质研究所岩石室投入使用, 经过二年多的实际运行, 说明程序运行正常、速度快、精度高。

本书共分八章, 第一章是微型计算机简介和 BASIC 语言基础。为了便于读者阅读和移植本书的程序, 在本章中, 综合了 CROMEMCO II 机的扩展 BASIC、TRS-80 机的 BASIC II 和 APPLE II 机的 MBASIC 指令对照表, 并论述了这三种微型计算机 BASIC 语言的差异。第二章是岩石学、矿物学和地球化学数据处理的方法。第三章到第八章是岩石学和矿物学数据处理系统的实用程序。本书的第二章由从柏林执笔, 其余各章由沈步明执笔。编书过程中, 承蒙中国科学院地质研究所李秉伦、叶大年教授, 清华大学计算机系杨德之同志, 国家地震局地质研究所王启鸣同志仔细审稿, 并提出了许多宝贵意见, 还得到王晓春、金成伟、郑学正等同志的帮助, 谨此一并致谢。

由于我们的水平有限, 编写时间匆促, 因此本书错误在所难免, 敬请读者批评指正。

作者

一九八五年四月于北京

目 录

第一章 微型计算机简介和 BASIC 语言基础	I
第一节 微型计算机简介	1
第二节 微型计算机的语言	1
第三节 CROMEMCO II, TRS-80, APPLE II 微型计算机的 BASIC 语言对照	2
一、算术运算符	2
二、关系运算符	2
三、布尔运算符	3
四、运算符运算的优先等级	3
五、内部函数	3
六、自定义函数	4
七、字符串函数	4
八、赋值语句	7
九、注释语句	8
十、初始化语句	8
十一、数据类型定义的语句	8
十二、输入和输出语句	11
十三、控制转移语句	13
十四、程序调试指令	14
十五、系统文件状态函数	16
十六、数据文件指令	18
第四节 程序设计方法和技巧	26
一、建立数学模型	26
二、画框图(流程图)	27
三、变量和变量类型的确定	27
四、三种赋值语句的灵活应用	28
五、判断转移语句的运用	32
六、循环语句的使用	32
七、格式打印	36
八、程序调试	39
九、重编语句行号	39
十、程序的结构化	39
第二章 岩石学、矿物学和地球化学中的数据处理方法	40
第一节 岩石学、矿物学和地球化学数据的特点	40
一、数据误差的来源	40
二、频度分布和概率	46
第二节 主要统计量的计算以及一元线性回归分析	49

一、主要统计量的计算	49
二、相关系数	51
三、一元线性回归	52
第三节 数据图解的几何学及其应用	53
一、空间中两点之间的距离	54
二、空间中直线方向的确定	56
第四节 体系成分变异图的数学解释	58
一、体系成分变异趋势的图解法	58
二、线的方程式和加减问题	60
第五节 多元回归分析与岩石混合模式	63
一、多元线性回归分析	63
二、岩石学混合模式	65
第三章 岩石学数据处理程序	73
第一节 标准矿物分子计算程序 (CIPW 算法)	73
一、程序设计方法	73
二、程序说明	74
三、功能	77
四、使用说明	78
五、程序文本和算例	78
第二节 简化的 Rittmann 岩石分类法程序 ^①	88
一、程序设计方法	88
二、程序说明	92
三、功能	93
四、使用说明	93
五、程序文本和算例	94
第三节 单位岩胞质能参数计算程序	102
一、程序设计方法	102
二、程序说明	104
三、功能	105
四、使用说明	105
五、程序文本和算例	106
第四章 矿物学数据处理程序	112
第一节 矿物晶体化学式和某些矿物端员分子数计算程序	112
一、程序设计方法	112
二、程序说明	114
三、功能	115
四、使用说明	117
五、程序文本和算例	119
第二节 地质温度计和压力计计算程序 (一)	129
一、程序设计方法	129
二、程序说明	131
三、功能	133

四、使用说明	133
五、程序文本和算例	134
第三节 地质温度计和压力计计算程序(二)	145
一、程序设计方法	145
二、程序说明	145
三、功能	146
四、使用说明	146
五、程序文本和算例	146
第四节 X射线粉末法晶胞参数最小二乘法修正程序	153
一、程序设计方法	153
二、程序说明	155
三、功能	157
四、使用说明	157
五、程序文本和算例	158
第五节 计算和核对矿物X光衍射线的指标化程序	167
一、程序设计方法	167
二、程序说明	168
三、功能	170
四、使用说明	170
五、程序文本和算例	171
第五章 数据统计的计算和检验程序	180
第一节 数据统计分析程序	180
一、程序设计方法	180
二、程序说明	183
三、功能	184
四、使用说明	185
五、程序文本和算例	185
第二节 岩石中主元素相关系数计算程序	193
一、程序设计方法	193
二、程序说明	193
三、功能	194
四、使用说明	194
五、程序文本和算例	195
第三节 相关系数计算程序	199
一、程序设计方法	199
二、程序说明	199
三、功能	199
四、使用说明	199
五、程序文本和算例	199
第四节 多元线性回归参数计算程序	203
一、程序设计方法	203
二、程序说明	204

三、功能	204
四、使用说明	204
五、程序文本和算例	205
第五节 聚类分析计算程序	208
一、程序设计方法	208
二、程序说明	210
三、功能	212
四、使用说明	212
五、程序文本和算例	214
第六章 绘图程序	223
第一节 日本 SR-6602 绘图仪指令简介	223
一、定义笔的移动速度: SP	223
二、初始化指令: IN	223
三、结束指令: TE	223
四、定义比例尺: DF	223
五、定义点线和点-虚线的长度: DS	224
六、定义直线的种类: LT	224
七、定义字符高度: SH	224
八、定义字符宽度: SI	224
九、定义坐标轴标记的长度: DH	224
十、斜体字符的起始指令: SC	225
十一、斜体字符的终止指令: EC	225
十二、笔的选择指令: NP	225
十三、笔的移动: AP, RP	225
十四、起伏线或点的绘制指令: AM, RM	225
十五、写字符指令: AS, RS	225
十六、轴的绘制指令: AX, RX	226
十七、圆、圆弧和弧的绘制指令: RC	226
十八、圆、圆弧和螺(旋)线的绘制指令: AC	226
十九、指令之间的分隔符: EB	226
二十、指令串的结束符: EX	226
二十一、带笔还回到初始位置指令: HM	227
第二节 半对数直角图绘图程序	227
一、程序设计方法	227
二、程序说明	227
三、功能	228
四、使用说明	229
五、程序文本和算例	230
第三节 直角坐标绘图程序	233
一、程序说明	233
二、功能	233
三、使用说明	234

四、程序文本和算例	235
第四节 三角图绘图程序	237
一、程序说明	237
二、功能	237
三、使用说明	238
四、程序文本和算例	239
第五节 双三角图绘图程序	241
一、程序说明	241
二、功能	241
三、使用说明	241
四、程序文本和算例	243
第七章 数据存取程序	246
第一节 数据存取程序的说明	246
第二节 数据存取程序的使用说明	248
第三节 数据存取程序的文本和算例	248
第八章 字符、字符串、数值大小顺序排列程序以及计算机内部码转换程序	253
第一节 字符大小顺序排列程序	253
一、程序说明	253
二、程序文本和算例	254
第二节 字符串大小顺序排列程序	254
一、程序说明	254
二、程序文本和算例	255
第三节 数值大小顺序排列和存取程序	257
一、程序说明	257
二、程序文本和算例	258
第四节 把十进制转换成二、八、十六进制程序	260
一、程序说明	260
二、程序文本和算例	262
第五节 把二、八、十六进制转换成十进制程序	264
一、程序说明	264
二、程序文本和算例	265
参考文献	268

第一章 微型计算机简介和 BASIC 语言基础

第一节 微型计算机简介

微型计算机又称微电脑,所谓电脑,是因为它具有类似人脑的功能,比如:具有记忆、运算和判断的功能。微型计算机主要由三部分组成:主机、终端显示器和打印机。主机是计算机的主要部分,它包括运算器、存储器和控制器。如果把计算机的计算比作算盘计算的话,那么运算器相当于算盘,存储器相当于纸和笔,控制器相当于脑和手。终端显示器包括荧光屏显示器和键盘二部分。打印机有 80 行、132 行打印机。计算机的主机和外围设备统称为“硬件”。计算机算法语言的编译程序、解释程序、汇编程序、诊断维修系统、故障处理系统、标准程序库、操作系统等统称为“软件”。用户编制的程序也属软件。

目前微型计算机的主机大多数采用 Z-80, Z-80A, 8080、8088 等大规模集成电路作为中央处理机的核心部件,更高档微型计算机采用 80186, 65000 等大规模集成电路作为中央处理机。显然大规模集成电路的类型决定了微型计算机的功能。决定微型计算机的另一个重要指标是内存,内存越大,贮存和计算能力就越大,目前市场上出售的微型计算机内存有 16KB、32KB、48KB、64KB、128KB、512KB 等,甚至可达一兆以上。

近几年来,我国的微型计算机有了大幅度的增加,大多从美国进口,目前国内比较多的微型计算机有 TRS-80, APPLE II, CROMEMCO II 或 III, IBM PC/XT, IBM PC/AT 等,还进口了一大批与 IBM 系统兼容的微型计算机。

第二节 微型计算机的语言

微型计算机使用的语言比较多,包括 BASIC, FORTRAN, COBOL, PASCAL 语言等。但是, BASIC 语言是微型计算机中应用最广,使用最方便的语言。因为 BASIC 语言所使用的指令和语句中的语言以及运算符与英语中使用的词和数学中的符号差不多,比较简单、直观、易学好用。另外 BASIC 语言是会话式的语言,可以进行人机对话,但是 FORTRAN, COBOL, ALGOL 等语言就不如 BASIC 语言那么方便,这是因为 FORTRAN 等语言采用编译方式,即:事先编好一个称编译程序的机器指令程序,然后放在计算机中,把用算法语言写的源程序输入计算机后,编译程序便把源程序整个地翻译成用机器指令表示的目的程序,最后执行目的程序。而 BASIC 语言则基本上采用解释方式,它与编译方式不同,它事先编好的是一个解释程序,它并不是象编译方式那样把源程序整个地翻成目的程序,然后再执行目的程序,而是逐句地翻译,译出一句,立即执行,即边解释,边执行。正因为如此,在执行 BASIC 语句的程序过程中,用户可以干预程序的执行,或暂停,或修改,或终止,可以边执行、边修改,十分方便。

因为 BASIC 语言不仅具有计算能力,而且还具有一定的数值处理的能力,特别是扩

展 BASIC 语言增加了字符串处理和自选打印格式的功能,所以,它适用于小型的数据处理和事务管理工作。当然 BASIC 语言是一种小型的算法语言,由于允许数的范围以及变量、数组和自定义函数的数目都受到限制,因此不能进行大型课题的计算。

BASIC 语言可分为基本 BASIC 语言、扩展 BASIC 语言、结构 BASIC 语言三种,因为基本 BASIC 语言过于简单,结构 BASIC 语言又过于复杂,语言本身要占用大量内存(32KB),因此本书的程序采用扩展 BASIC 语言编写。

第三节 CROMEMCO II, TRS-80, APPLE II 微型计算机的 BASIC 语言对照

因为本书程序都是在 CROMEMCO II 型机上使用扩展 BASIC 语言编成的,虽然各种微型计算机使用的 BASIC 语言的指令大体相同,但是还有些小的差别,为了便于读者阅读本书的程序,为了便于读者将本书的程序转移到其它微型计算机上使用,我们对 CROMEMCO II 微型计算机的扩展 BASIC 语言,TRS-80 微型计算机上的 BASIC II 和 APPLE II 微型计算机的 MBASIC 语言进行了对比。为什么选用这三种微型计算机的 BASIC 语言进行对比呢?其一是:这三种微型计算机的 BASIC 语言大致可以代表所有微型计算机中 BASIC 语言的特点;其二是:目前国内具有这类微型计算机的用户较多。

对这三种微型计算机中差别较大的 BASIC 语句将作些说明,对某些难于用文字解释清楚的语句,将举出程序例子给予说明。

一、算术运算符

CROMEMCO II	TRS-80	APPLE II	功 能
+	+	+	正号(正数)
-	-	-	负号(负数)
**或↑	↑	^	指 数
*	*	*	乘
/	/	/	除
+	+	+	加
-	-	-	减
()	()	()	括 号

从对比表中可以看出,除了指数以外,这三种微型计算机的算术运算符的指令是完全相同的。

二、关系运算符

CROMEMCO II	TRS-80	APPLE II	功 能
=	=	=	等于
<	<	<	小于

>	>	>	大于
<=	<=或=<	<=或<=	小于或等于
>=	>=或=>	>=或=>	大于或等于
<>或#	<>或><	<>或><	不等于

很明显, CROMEMCO II 型机的关系运算符除了不等于 (#) 关系运算符以外, 都能在 TRS-80 机和 APPLE II 机上运行, 但反过来却不然, TRS-80 和 APPLE II 机上的 =, <, =>, >< 关系运算符不能在 CROMEMCO II 机上运行。

三、布尔运算符

CROMEMCO II	TRS-80	APPLE II	功 能
AND	AND	AND	逻辑与
OR	OR	OR	逻辑或
NOT	NOT	NOT	逻辑非或反

四、运算符运算的优先等级

1. () 括号(从里到外)
2. ↑或** 指数
3. - 负数
4. *, / 乘, 除(自左至右)
5. +, - 加, 减(自左至右)
6. <, >, =, <=, >=, <> 关系运算符
7. NOT 逻辑非
8. AND 逻辑和
9. OR 逻辑或

五、内部函数

CROMEMCO II	TRS-80	APPLE II	功 能
ABS(X)	ABS(X)	ABS(X)	取 X 的绝对值
EXP(X)	EXP(X)	EXP(X)	取 e ^x 的值
FRA(X)			取 X 的小数部分
INT(X)	INT(X)	INT(X)	取 X 的整数部分 (小于或等于 X 的 最大整数)
	FIX(X)	FIX(X)	取 X 被截尾后的 整数
LOG(X)	LOG(X)	LOG(X)	求 log _e X 的值
SQR(X)	SQR(X)	SQR(X)	求 X 的开平方值

SIN(X)	SIN(X)	SIN(X)	求 $\sin X$ 的值
SGN(X)	SGN(X)	SGN(X)	取符号函数值(当 $X > 0$, $SGN(X) = 1$; 当 $X = 0$, $SGN(X) = 0$; 当 $X < 0$, $SGN(X) = -1$)
COS(X)	COS(X)	COS(X)	求 $\cos X$ 的值
TAN(X)	TAN(X)	TAN(X)	求 $\tan X$ 的值
ATN(X)	ATN(X)	ATN(X)	求 $\arctan X$ 的值
IRN(X)	RND(X)	RND(X)	整型随机函数
RND(X)	RND(0)	RND(0)	小数随机函数
RANDOMIZE	RANDOM	RANDOMIZE	随机化函数
MIN($X_1, \dots, X_n$)			求 n 个算术表达式 中的极小值
MAX($X_1, \dots, X_n$)			求 n 个算术表达式 中的极大值

从表中可以看出: 反三角函数只有 ATN(X) 一种, $\sin^{-1}x$ 和 $\cos^{-1}x$ 可用下列公式计算:

$$\cos^{-1}x = \arccos x = -\arctg(x \div \sqrt{1-x^2}) + 2\arctg 1$$

$$\sin^{-1}x = \arcsin x = \arctg(x\sqrt{1-x^2})$$

在 BASIC 语句中, 若不作特殊声明, 那么所有三角函数计算都以弧度制表示。在 CROMEMCO II 型机中, 可以利用 DEG 指令, 指定三角函数用角度制表示, 但在 TRS-80 机和 APPLE II 机中, 没有 DEG 指令, 所有三角函数都以弧度制表示。

六、自定义函数

CROMEMCO II	TRS-80	APPLE II	功 能
DEF FN (自定义函数)	DEF FN (自定义函数)	DEF FN (自定义函数)	

七、字符串函数

CROMEMCO II	TRS-80	APPLE II	功 能
+	+	+	字符串加法运算符
ASC(字符串)	ASC(字符串)	ASC(字符串)	求字符串第一个字 符的 ASC II 码
CHR\$(表达式)	CHR\$(表达式)	CHR\$(表达式)	求十进位制的 ASC II 码所对应的字符

LEN (字符串)	LEN (字符串)	LEN (字符串)	求字符串实际字符长度
POS (字符串1, 字符串2, 表达式)	INSTR (表达式, 字符串1, 字符串2)	INSTR (表达式, 字符串1, 字符串2)	求子字符串位置函数
STR\$ (表达式)	STR\$ (表达式)	STR\$ (表达式)	数值→字符串转换函数
VAL (字符串)	VAL (字符串)	VAL (字符串)	字符串→数值转换函数
字符串 (0, n)	LEFT\$ (字符串, n)	LEFT\$ (字符串, n)	取字符串的前 n 个字符
字符串 (a ₁ , a ₂)	MID\$ (字符串, n, p)	MID\$ (字符串, n, p)	从字符串中的第 n 个位置开始, 取 p 个字符
字符串 (n)	RIGHT\$ (字符串, n)	RIGHT\$ (字符串, n)	取字符串的最后 n 个字符
	MID\$ (X\$, N, M) = Y\$	MID\$ (X\$, N, M) = Y\$	从 X\$ 字符串中第 N 个位置开始, 用 Y\$ 字符串字符逐个替代 M 个字符
字符串 = "字符" + 字符串 (-1)	STRING\$ (n, 字符)	STRING\$ (n, 字符)	取 n 个指定字符的字符串

从字符串函数的对照表中可以看出: TRS-80 机和 APPLE II 机的指令, 除了个别以外, 绝大部分是相同的, 但是和 CROMEMCO II 型机的指令相比, 却有很大的差别。主要差别是:

1. 字符串长度的定义, 在 CROMEMCO II 型机中, 字符串变量是由一个字母, 跟一个 \$ 或一个字母, 跟一个数字 (0-9), 再跟一个 \$ 来表达, 如果字符串的长度不超过 11 个字符, 它的长度就可以不定义, 如果超过 11 个字符的话, 要用 DIM 语句定义字符串变量的长度, DIM 指令只能定义一维下标字符串长度, 而不能定义带二维或三维字符串的长度, 但是使用带二维变量的字符串不会出错, 因为带二维下标的字符串是带一维下标字符串的子字符串, 其中字符串的第一个下标代表字符串的长度, 其长度可用 DIM 定义一维字符串的方法定义, 第二个下标是引用字符串的全体或部分时而使用的。

在 TRS-80 机和 APPLE II 型机中, 字符串的定义与 CROMEMCO II 型有重要差异。比如: 所有的变量名都可以用来装字符串值, 但变量名必须使用 DEFSTR 语句或加上类型说明符来定义。字符串数组也是这样, 比如 C\$(15) 中的下标可以被解释为串数组的体积大小, 而不是象 CROMEMCO II 机那样被解释为字符串的长度, 因此, 串数组 C\$(15, 16), C\$(15, 16, 18) 也是合法的。

在 TRS-80 机中, 字符串的长度是靠 CLEAR 指令来定义的, 当计算机一打开时, 自动地为字符串开辟了 50 个字节的内存。如果要增加就要利用 CLEAR n 指令, 为字符串存储开辟指定数目 n 个字节内存。用 CLEAR 开辟的字符串存储的数量必须等于或大于字符串变量在执行过程中需要存储字符的最大数目, 否则将出错。

2. 在字符串函数中, TRS-80 机、APPLE II 型机与 CROMEMCO II 机另一个重要差别

是字符串的输出。为了比较清楚地说明这些差别,现举例说明如下。

程序例一:

```
10 DIM C$(15),D$(15)
20 A1$="ADCG"
40 C$="123416890125636241"
50 D$="MOPUY"
52 @ C$(13)
53 @ C$(-1)
55 @ A1$(0,3)
56 @ A1$(1,3)
70 @ A1$(2,3)
75 @ A1$(2,-1)
76 @ A1$(-1,2)
80 @ C$(5,8)
81 @ C$(6,4)
85 @ C$(7,4)
120 @ D$(16,5)
```

```
>>RUN
362
1234168901256362
ADCG
DCG
CG
C
ADCG
6890
8901256362
901256362
```

Error 15 at line 120 -- Subscript value(s)

* * * End * * *

程序例一是在 CROMEMCO II 型上编成的¹⁾。

语句 10 给字符串 C\$ 和 D\$ 定义字符串长度。如果字符串赋值时超过了字符串所定义的长度,则多余部分不能输出,如语句 40 给 C\$ 字符串赋值了 18 个字符,但实际上 C\$ 只定义了 16 个字符长度(0—15),所以 C\$ 字符串的最后二个字符“41”被删去,不能输出。因为语句 120 的 D\$(16,5) 字符串中第一个下标是 16,超过定义的 D\$(15),所以,运行时给出错误信息。

语句 55—85 是利用带二个下标的子字符串来输出字符,比较复杂,大致可综合成下列几种情况。

(1) A\$(a)——字符串只有一个下标。a 表示从字符串的第 a 个位置开始输出,直到第 n 个字符为止, n 是被定义字符串的长度,如语句 52。如果 a 为负数时,则输出整个字符串,如语句 53。

(2) A\$(a₁, a₂)——如果下标 a₁, a₂ 都是正数,并且 a₁ < a₂, 那么 a₁ 表示输出字符串的起始位置, a₂ 表示终止位置,如语句 55—70 中的 A1\$(0,3), A1\$(1,3), A1\$(2,3)。如果 a₁ 和 a₂ 都是正数,但 a₁ > a₂, 则 a₁ 为起始位置,直到整个字符串结束,如语句 81—85 中的 C\$(6,4), C\$(7,4)。如果 a₁ 为负数,则输出整个字符串,如语句 76 中的 A1\$(-1,2)。如果 a₂ 为负数,则 a₁ 表示输出起始位置, a₂ 的绝对值表示输出字符串的长度,如语句 75 的 A1\$(2,

1) @ 表示指令 PRINT,下同。

程序例二:

```
>>RUN  
DFERTYUIJ  
LEFT$(A$,3)=DFE  
MID$(A$,3,2)=ER  
MID$(A$,3)=ERTYUIJ  
RIGHT$(A$,3)=UIJ  
-----  
A$      =DFERTYUIJ  
A$ *    =DFERTY***  
-----  
STRING$(15,&)=#####  
  
***End***
```

语句 150,得到 15 个“&”字符串.

3. 在字符串函数中, CROMEMCO II 机与 TRS-80 机的第三个重要差别是求子字符串位置函数, 在 CROMEMCO II 机中, 求子字符串位置函数是 POS。在 TRS-80 机中, 虽然也有 POS 指令, 但其功能是不同的, 它代表光标正处在显示器上的位置, 求子字符串位置函数的指令是 INSTR, 这一点是特别要注意的。

八、赋值语句

CROMEMCO II	TRS-80	APPLE II	功 能
LET	LET	LET	给变量赋值
MAT			给矩阵赋值

要注意的是：在扩展 BASIC 语言中，LET 可以省略，可给变量或字符串直接赋值。另外，CROMEMCO II 机的扩展 BASIC 语言只有矩阵赋值功能，没有矩阵打印、矩阵相乘和矩阵求逆等功能。TRS-80 机和 APPLE II 机的 BASIC 语言都没有矩阵赋值语句的功能。

九、注释语句

CROMEMCO II	TRS-80	APPLE II	功 能
REM	REM	REM	注释说明

十、初始化语句

CROMEMCO II	TRS-80	APPLE II	功 能
DEG			指定三角函数以角度制计算
RAD			指定三角函数以弧度制计算
DIM	DIM	DIM	给数组指定长度

如前面所述，在 TRS-80 机和 APPLE 机中，所有三角函数都以弧度制计算，而在 CROMEMCO II 机中，则可以使用角度制计算，也可以使用弧度制计算，本书中的实用程序大多数以角度制计算（即利用 DEG 指令，指定以角度制计算），所以，要把本书的程序移植到其它机上应用时，要作弧度转换。

在 CROMEMCO II, TRS-80, APPLE II 机中，虽然都有 DIM 指令，可是它的功能却有很大的差别。在 CROMEMCO II 机中，利用 DIM 指令定义的数组是双精度的，而在 TRS-80 机和 APPLE II 机中，利用 DIM 定义的数组是单精度的。另外，在 CROMEMCO II 型机中，利用 DIM 指令定义的字符串数组的下标，表示字符串的长度，并且只能定义一维数组的字符串，不能定义二维或三维字符串数组，而在 TRS-80 机和 APPLE II 机却可以有一维、二维、三维字符串的数组，其下标表示字符串数组的大小，所以如果把本书的实用程序移植到其它的机上使用时，利用 DIM 定义字符串长度的语句可以删去，但是如果其字符串的总长度在运行时超过 50 个字节，就要用 CLEAR 指令，开辟字符串存贮的内存。

十一、数据类型定义的语句

CROMEMCO II	TRS-80	APPLE II	功 能
SHORT	DEFSNG	DEFSNG	定义单精度变量
LONG	DEFDBL	DEFDBL	定义双精度变量
INTERGER	DEFINT	DEFINT	定义整型变量
SFMODE			指定所有变量为短浮点型
LFMODE			指定所有变量为长浮点型
IMODE			指定所有变量为整型
	DEFSTR	DEFSTR	定义字符串变量