

四川省计算机基础教育研究会推荐系列教材

Quick BASIC

结构化程序设计

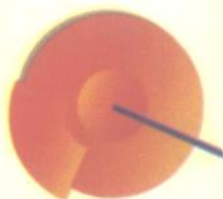
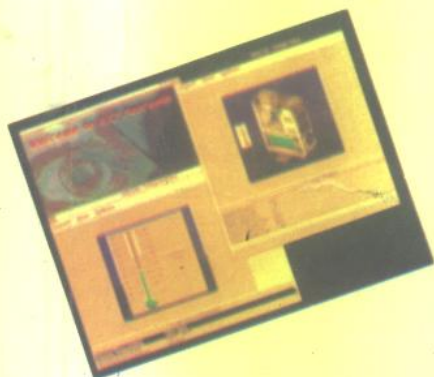
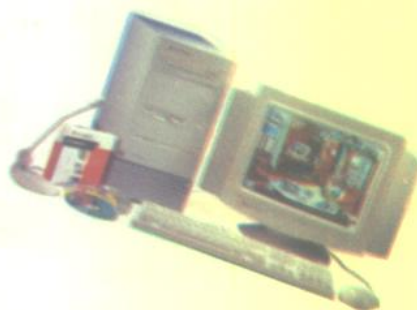


主编 黎 明

编著 黎 明

主审 张慧云

陈继明 许泽明



电子科技大学出版社

四川省计算机基础教育研究会推荐系列教材

Quick BASIC 结构化程序设计

丛书顾问 兰家隆 古天祥 王正智
黄迪明 梁光春 惠林

丛书编委会 张慧云 陆成 李光琳 向孟光
谢惠娟 蔡学望 吴家培 杨国才
黎明 杨明广

主编 黎明
编著 黎明 陈继明 许泽明
主审 张慧云

电子科技大学出版社

内 容 简 介

BASIC 语言是国际上通用的普及面最广、发展最快的一种计算机语言。在我国也是广大计算机用户最常用的一种计算机语言。Quick BASIC 语言是美国Microsoft 公司在1987 年11 月推出的BASIC 版本。它集中了解释BASIC 的交互作用长处和编译器的快速性,加上模块化结构的优点,具有入门容易,功能很强,使用方便等特点。

本书详细介绍了Quick BASIC 语言的基本知识。全书共分十一章,内容包括Quick BASIC 的结构特点、语法规则、基本语句和控制结构语句、数组、函数、子程序、磁盘文件、图形、常用算法、上机操作等部分。每章附有习题,供读者练习和思考。

本书可作为高等院校计算机基础教育的教材,也可供各类工程技术人员,管理部门自学或参考。

四川省计算机基础教育研究会推荐系列教材

Quick BASIC 结构化程序设计

主编 黎 明

编著 黎 明 陈继明 许泽明

主审 张慧云

*

电子科技大学出版社出版

(成都建设北路二段四号) 邮编 610054

四川省自然资源研究所印刷厂印刷

新华书店经销

*

开本 787×1092 1/16 印张 16.75 字数 407 千字
版次 1997 年 2 月第一版 印次 1997 年 7 月第三次印刷

印数 7001—12000 册

ISBN 7—81043—573—6/TP·229

定价:16.50 元

序

随着科学技术的飞速发展,计算机已成为各行各业不可缺少的应用工具,计算机知识和应用能力已成为当代大学生培养质量高低的一个重要组成部分,是面向 21 世纪的高等教育培养高质量人才最突出需要加强的教学环节之一。目前在全国高校中普遍开展的计算机知识和应用能力等级考试制度正有效地推动这一目标的实现,四川省已成功地举行了五次这样的考试正充分地证明了这一点,而已成立三年多的四川省计算机基础教育研究会为保证我省计算机知识和应用能力等级考试制度和方法的不断完善提供了有力的保证。

到目前为止,有关计算机应用等级考试的图书不少,对推动我省计算机知识和应用能力的普及起了积极的作用,但由于他们中的大多数是个别作者独立写作,难以反映众多学生对教材的广泛需求。这一套计算机等级考试二级系列通用教材及配套复习资料和实验指导书一共十五种的主编和执笔者是来自省内许多大专院校,他们均是长期工作在该领域教学第一线,有丰富教学经验的优秀教师,他们中大多数是四川省计算机基础教育研究会成员单位的代表,他们的

参与为本套教材的质量和水平提供了可行的保证,也使本套教材有十分广泛的代表性和符合四川省等级考试大纲的要求。电子科技大学出版社为这套系列教材的成功编写作了大量的组织工作。

综上所述,本系列教材可作为我省高校非计算机专业计算机等级考试(二级)的教材或参考书。我相信,这套系列教材的出版将进一步推动我省高校计算机基础教育质量的提高,推动计算机应用的进一步普及,对我国全民族现代化素质的进一步提高有所裨益。

四川省计算机等级考试委员会副主任

兰家隆

1997年元月30日于成都

前 言

随着计算机应用的日益广泛和深入,让各个领域的科技工作者快速地掌握一门计算机算法语言,是当前普及计算机文化的重要任务。Quick BASIC 语言具有入门容易、交互性的长处、编译器的快速性、程序结构模块化、高级功能强、使用方便等特点,深受广大用户的欢迎。本书按照四川省普通高校计算机等级考试大纲的要求编写,既注意了知识的系统性,又突出了精选内容面向应用。参考学时数为40~60学时。

全书共分为十一章。详细介绍了Quick BASIC 语言的基本特点、语法规则、基本语句、控制结构语句、数组、函数及子程序、磁盘文件、图形功能、常用算法、上机操作等内容。各章的习题对培养学生的编程水平和上机动手能力都具有较大益处。

本书由黎明主编。第一至四章由黎明编写,第五至八章由许泽明编写,第九至十一章由陈继明编写。国家教委全国高等医药院校计算机教学指导委员会副主任委员、四川省教委计算机考试中心委员、华西医科大学张慧云教授仔细审阅了全部书稿,并对本书提出了许多宝贵意见,在此表示衷心的感谢。由于编者水平有限,时间仓促,书中难免有不妥之处,恳切希望读者批评指正。

编 者

一九九六年六月

目 录

第一章 基础知识	1
1.1 Quick BASIC 语言简介	1
1.1.1 从传统 BASIC 到 Quick BASIC	1
1.1.2 Quick BASIC 程序结构简介	2
1.2 字符集和程序行	2
1.2.1 字符集	2
1.2.2 程序行	3
1.3 数据类型	4
1.4 常量和变量	6
1.4.1 常量	6
1.4.2 变量	8
1.4.3 运算中数值精度的类型转换	10
1.5 表达式与运算符	11
1.5.1 算术运算符	12
1.5.2 关系运算符	12
1.5.3 逻辑运算符	13
1.5.4 内部算术函数简介	13
1.5.5 表达式的执行顺序	14
1.6 结构化程序设计简介	15
1.6.1 程序设计的三种基本结构	15
1.6.2 结构化程序设计方法简介	16
1.6.3 N-S 结构化流程图简介	19
1.6.4 结构化程序设计举例	19
习题一	21
第二章 顺序结构程序设计	24
2.1 Quick BASIC 基本语句	24
2.1.1 赋值语句(LET)	24
2.1.2 注释语句(REM)	25
2.1.3 暂停语句(STOP)	25
2.1.4 结束语句(END)	26
2.1.5 交换变量语句(SWAP)	26
2.2 输入数据语句	29
2.2.1 键盘输入语句(INPUT)	29

2.2.2	行输入语句(LINE INPUT)	31
2.2.3	读数据语句(READ-DATA)	32
2.2.4	恢复读数据语句(RESTORE)	33
2.3	显示输出语句	35
2.3.1	标准显示输出语句(PRINT)	35
2.3.2	与标准显示输出语句有关的函数	36
2.3.3	屏幕显示语句(WRITE)	40
2.3.4	指定格式显示语句(PRINT USING)	41
2.4	打印输出语句	43
2.4.1	打印语句(LPRINT 与 LPRINT USING)	43
2.4.2	测定当前打印位置的函数(LPOS)	43
习题二		44
第三章	流程控制结构程序设计	46
3.1	转移控制语句	46
3.1.1	无条件转移语句(GOTO)	46
3.1.2	多分支选择转移语句(ON-GOTO)	47
3.2	判别条件选择语句	49
3.2.1	单行结构判别条件语句(IF-THEN-ELSE)	49
3.2.2	块结构判别条件语句(IF-THEN-END IF)	50
3.3	情况语句	53
3.4	循环语句	55
3.4.1	FOR-NEXT 循环语句	55
3.4.2	当循环语句(WHILE-WEND)	59
3.4.3	DO 循环语句(DO-LOOP)	61
3.4.4	多重循环与非正常退出循环	63
习题三		69
第四章	数组	72
4.1	静态数组	72
4.1.1	静态数组的定义	72
4.1.2	数组下标下界的修改	73
4.1.3	数组下标界限的测试函数	74
4.2	动态数组	75
4.2.1	动态数组的定义	75
4.2.2	数组的删除	75
4.3	数组的基本操作	76
4.3.1	数组元素的输入输出	76
4.3.2	数组元素的复制	79

4.3.3 数组与矩阵运算.....	83
习题四	87
第五章 函数	91
5.1 标准函数.....	91
5.2 自定义函数.....	97
5.2.1 单行自定义函数.....	97
5.2.2 多行自定义函数.....	99
习题五.....	100
第六章 过程.....	102
6.1 GOSUB 子程序	102
6.1.1 转子语句和返回语句	102
6.1.2 GOSUB 子程序的调用	102
6.2 FUNCTION 过程	108
6.2.1 引例	108
6.2.2 FUNCTION 过程的定义	108
6.2.3 FUNCTION 过程的调用	109
6.2.4 举例	110
6.3 SUB 过程	111
6.3.1 引例	111
6.3.2 SUB 过程的定义	111
6.3.3 SUB 过程的调用	112
6.3.4 举例	112
6.4 过程的递归调用	114
6.4.1 FUNCTION 过程的递归调用	114
6.4.2 SUB 过程的递归调用	116
6.5 全局变量和局部变量	117
6.5.1 全局变量	117
6.5.2 局部变量	118
6.6 数据传送	119
6.6.1 虚实结合	119
6.6.2 共享变量	123
6.7 应用举例	126
习题六.....	131
第七章 字符串.....	133
7.1 字符串常量	133
7.1.1 可变长字符串	133

7.1.2 固定长字符串	133
7.2 字符串变量和字符串数组	134
7.2.1 字符串变量	134
7.2.2 字符串数组	135
7.3 字符串变量的赋值	135
7.3.1 用LET 语句	135
7.3.2 用INPUT 语句	136
7.3.3 用READ/DATA 语句	136
7.4 字符串运算	137
7.4.1 字符串连接运算	137
7.4.2 字符串关系运算	138
7.5 字符串函数	138
7.5.1 测字符串长度函数(LEN)	138
7.5.2 求子字符串函数	139
7.5.3 查子字符串位置函数(INSTR \$)	141
7.5.4 数值与字符串相互转换函数(STR \$、VAL)	142
7.5.5 ASCII 字符与序号相互转换函数(ASC、CHR \$)	143
7.5.6 字符串生成函数(String \$)	144
7.5.7 产生空格字符串函数(SPACE \$)	144
7.5.8 字符串大小写转换函数(LCASE \$、UCASE \$)	145
7.5.9 删除字符串首尾空格函数(LTRIM \$、RTRIM \$)	145
7.6 应用举例	146
习题七	149
第八章 图形和音乐	152
8.1 屏幕坐标系统	152
8.1.1 字符模式	152
8.1.2 图形模式	154
8.2 屏幕控制语句	154
8.2.1 SCREEN 语句	154
8.2.2 COLOR 语句	155
8.3 基本绘图语句	158
8.3.1 画点语句(PSET、PRESET)	158
8.3.2 画线语句(LINE)	159
8.3.3 连续画线语句(DRAW)	159
8.3.4 画圆和圆弧语句(CIRCLE)	161
8.4 图形着色	162
8.5 窗口和视见区	163
8.5.1 窗口语句(WINDOW)	163

8.5.2 视见区语句(VIEW)	164
8.5.3 文本视见区语句(VIEW PRINT)	166
8.6 简单动画	167
8.6.1 字符动画	167
8.6.2 图形动画	167
8.7 应用举例	169
8.8 音乐	172
习题八	175
第九章 Quick BASIC 数据文件	177
9.1 文件概述	177
9.1.1 文件的分类及组成	177
9.1.2 文件性质	178
9.1.3 文件命令语句	178
9.2 顺序文件	179
9.2.1 顺序文件操作语句	179
9.2.2 向顺序文件添加数据	182
9.2.3 从顺序文件中读取数据	183
9.2.4 修改顺序文件	185
9.3 随机存取文件	187
9.3.1 建立随机文件	188
9.3.2 记录中域的定义	188
9.3.3 将数据写入随机文件	189
9.3.4 从随机文件中读取数据	191
9.4 包含文件(蕴含文件)	193
9.4.1 包含文件的定义	194
9.4.2 包含文件的调用	194
9.5 二进制文件及其它文件方式	197
9.5.1 二进制文件	197
9.5.2 二进制存取与随机存取文件	198
9.5.3 文件指针定位	198
9.5.4 LOC 函数	199
9.5.5 设备文件	200
习题九	201
第十章 常用算法程序设计	204
10.1 算法的概念	204
10.2 递推化方法	205
10.3 方程求根	207

10.3.1 二分法.....	207
10.3.2 牛顿(Newton)迭代法	208
10.4 数值积分.....	210
10.4.1 矩形法.....	210
10.4.2 梯形法.....	211
10.5 排序方法.....	212
10.5.1 选择法排序.....	212
10.5.2 插入排序法.....	214
10.5.3 希尔排序.....	216
10.6 查找.....	219
10.6.1 顺序查找.....	220
10.6.2 折半查找.....	221
10.7 有序数列的插入、删除操作	225
10.7.1 有序数列的插入操作.....	225
10.7.2 有序数列的删除操作.....	227
第十一章 Quick BASIC 上机操作	230
11.1 概述.....	230
11.1.1 Quick BASIC 的运行环境	232
11.1.2 Quick BASIC 4.5 系统文件.....	232
11.1.3 进入Quick BASIC 环境	233
11.1.4 退出Quick BASIC	234
11.2 Quick BASIC 环境的操作	235
11.2.1 Quick BASIC 屏幕	236
11.2.2 Quick BASIC 的窗口操作	236
11.2.3 鼠标器的使用.....	237
11.3 Quick BASIC 菜单操作	237
11.3.1 源文件(File)菜单	237
11.3.2 编辑(Edit)菜单	238
11.3.3 View 菜单	239
11.3.4 Search 菜单	240
11.3.5 RUN 菜单	240
11.3.6 调试(Debug)菜单	244
11.3.7 Call 菜单.....	244
11.3.8 Option 菜单	245
11.3.9 Help 菜单	245
附录一 Quick BASIC 4.5 语句和函数一览表.....	246
附录二 代码表.....	253
主要参考文献.....	256

第一章 基本知识

1.1 Quick BASIC 语言简介

1.1.1 从传统 BASIC 到 Quick BASIC

BASIC 是英文 Beginner's All-purpose Symbolic Instruction Code 的缩写(初学者的通用符号指令代码)。它是由美国 Dartmouth 学院的 John. G. Kemeny 和 Thmas. E. Kurtz 两位教授开发的一种程序设计语言。BASIC 最初问世于 1964 年,当时只有十几条语句,由于简单实用而受到用户欢迎,很快被广泛应用于各个领域。随着计算机科学技术的发展,计算机厂商纷纷在原有的基础上不断地添加新的功能,进行扩展强化,各种 BASIC 版本应运而生,使得它成为最常用的计算机程序设计语言。

BASIC 语言具有两种工作方式:程序运行方式和键盘命令方式。程序运行方式是将编好的程序送入计算机后,通过解释方式边解释边执行,或者通过编译方式去执行。键盘命令方式是用键盘直接键入不代行号的命令或语句,计算机立即执行。例如,用键盘键入输出语句:

```
PRINT 5 * 7 - 10
```

计算机立即显示运算结果: 25。

BASIC 语言主要以程序运行方式来工作。基本 BASIC 通常只能以解释方式运行,它编程调试比较方便,但执行时需要逐条语句解释译码,执行速度比较慢,而且所编的程序只能在 BASIC 环境中运行。

Quick BASIC 是美国 Microsoft 公司 1987 年 11 月推出的 BASIC 版本,它克服了原有的 BASIC 结构化程度差和运算速度慢的缺点,并且在功能上大为扩充。它把编辑、运行和调试有机地结合起来,编辑程序时可以自动将关键字转换为大写,还能校正遗漏和错误。调试程序时,可方便地将运行程序随时暂停,进行编辑。可以修改正在运行程序中的数据。Quick BASIC 还能快速生成可执行文件及多个模块同驻内存。经过调试后的程序,可以生成 DOS 状态下可直接运行的文件。4.0 版以上的 Quick BASIC 支持多个模块同时驻留内存,可一步实现将内存中的全部模块编辑连接成一个可执行文件。Quick BASIC 可将用户编写的过程和运行的程序变为 BASIC 语言的虚拟扩充部分,自动进行库管理。它还支持多种图形方式,对于微机上常用的 VGA、MCGA、CGA、EGA 和单色的图形适配器及 ps/2 图形方式,其处理方法是相同的。

Quick BASIC 问世以来,已经推出了 2.0、3.0、4.0、4.5 等几个版本。目前较流行的是 4.5 版,本书所介绍的程序均以 Quick BASIC 4.5 版为参考。

Quick BASIC 与以前的传统 BASIC 兼容。大多数用传统 BASIC 开发的程序不做修改就可在 Quick BASIC 环境下运行或编译。

Quick BASIC 语言的源文件是 ASCII 格式,或 Quick BASIC 格式。因此,对于用 BA-

SIC 建立的程序必须以 ASCII 码文件存盘(使用“A”参数)。

1.1.2 Quick BASIC 程序结构简介

下面通过一个简单例子介绍 Quick BASIC 程序的结构。

例:用 Quick BASIC 程序计算下式的值。

$$Y = X^2 + \sqrt{1+2X^2}$$

程序如下:

```
REM 计算表达式的值
INPUT X          ' 键入X
S = 1 + 2 * X * X ' 求根号内表达式的值
Y = X * X + SQR(S) ' 求Y 的值
PRINT Y          ' 输出Y 的值
END              ' 程序结束
```

这个程序的首行是注释语句,它的作用是对程序做注释。REM 是英语单词remark 的缩写形式。在REM 后面到行尾的所有字符,是为了读程序方便而附加的说明或标记。通常可以把本段程序的名称,计算方法,程序目的等放入注释语句。注释的内容可以是英文字符或汉字。程序运行时,注释语句不起实质的作用,它并不被执行。仅当程序列表输出时,它才按原样输出。

从程序的第二行到第六行,各行后半部分以单引号开头,加入了中文注释。单引号开头的注释作用与REM 开头的注释语句相同,也是为了增加程序的可读性,便于人们阅读和理解。

程序的第二行是输入语句,它的作用是把键盘输入的数赋给变量X。

程序的第三、四行都是赋值语句。平常我们习惯使用的等号“=”,在语句中起赋值的作用,在这里被称为赋值号。赋值语句的作用是计算赋值号右端表达式的值,并把结果赋给左端的变量。表达式中的星号“*”代表算术运算中的乘号。其中第四行的 SQR(S) 的作用是计算变量S 的平方根。

程序的第五行是输出语句,它将变量Y 的值向显示器或打印机输出。

程序的第六行是结束语句,它用来终止程序的执行。在执行了 END 语句以后,将关闭所有的文件并返回命令状态。

从上面的例子可以看到,Quick BASIC 程序是由若干程序行组成;每个程序行又由有关语句组成;语句中包括一些特定的关键词。这些特定的关键词称为“保留字”。例如:REM、INPUT、PRINT、END、SQR 等,它们在 Quick BASIC 中具有特定的含义。程序中最基本的成分是字符。

1.2 字符集和程序行

1.2.1 字符集

Quick BASIC 的字符集与其它高级程序设计语言的字符集相似,包含字母、数字和专用字符三类,共90 个字符,它们是:

1. 字母：大写英文字母 A~Z；小写英文字母 a~z。
2. 数字：0~9；对于十六进制数，则包含大写英文字母 A~F，小写英文字母 a~f。
3. 专用字符：共28个，即：

<CR>	回车键	'	单引号
	空格	(	左圆括号
!	感叹号(单精度数据类型说明符)	)	右圆括号
#	镑号(双精度数据类型说明符)	*	星号(乘号)
\$	美元号(字符串数据类型说明符)	+	加号
%	百分号(整型数据类型说明符)	,	逗号
&	和号(长整型数据类型说明符)	-	减号
.	实心句号(小数点)	?	问号
/	斜杠(除号)	@	花a号
:	冒号	[	左方括号
;	分号	]	右方括号
<	小于号	\	反斜杠(整除号)
=	等于号	^	向上箭头(乘方号)
>	大于号	_	下画线(续行符号)

1.2.2 程序行

Quick BASIC 内部带有编辑程序。若用它来输入源程序，每个程序行的最大宽度不能超过 256 个字符，并且不能用下画线作续行符号。若使用其它编辑程序用了下画线作续行符号，则 Quick BASIC 在装入源程序时，将自动去掉下画线，把各个接续行连接为一行。但要注意不能用下画线接续 REM (注释)或 DATA (数据)语句。

Quick BASIC 的程序行的一般格式为：

[行标识符] [语句] [:语句] ... [注释]

【注】 方括号所包含的部分可以缺省，下同。

一、行标识符

Quick BASIC 的行标识符分为两种类型：行号和行标号。

1. 行号

行号是 0~65529 的整数。为了避免在某些场合下产生误会，最好不要用 0 作行号。在传统 BASIC 版本中，行号的大小决定语句执行的先后顺序；而在 Quick BASIC 中不是这样。例如有下列程序段：

```
50 LET X=1
60 LET Y=2
80 PRINT "X="; X
70 PRINT "Y="; Y
```

在基本 BASIC 版本中的执行顺序是：50、60、70、80，而在 Quick BASIC 中的执行顺序是：50、60、80、70。一般情况下，Quick BASIC 仅有少数语句使用行号。

2. 行标号

行标号也简称标号,它由字母和数字组成。行标号必须以字母开头并以冒号结束,它最长不超过40个字符。每行最多只能有一个行标号。要注意不能用 Quick BASIC 的保留字(见附录一)作为行标号。下面是合法的行标号:

abcd:

A123bcde:

一般情况下,在源程序中的行号和行标号可以混合使用。大部分语句中都可以用行标号代替原来程序中的目标行号。

二、Quick BASIC 的语句

Quick BASIC 的程序行中,可以有一个或多个语句。同行的各个语句之间用冒号分隔开。语句可分为两类:执行语句和非执行语句。执行语句可以完成某项指定的操作。譬如:读、写、计算、赋值、打开文件、转移控制等,都属于执行语句。通过执行语句把程序的逻辑流程向前推进。而非执行语句只是完成说明诸如为变量分配存储单元、说明变量类型、指明源文件中被过程共享的变量等任务。

Quick BASIC 的非执行语句包括:

- .REM 或 '(注释开始)
- .COMMON (申明变量可以传送到程序的其它模块)
- .CONST (定义符号常量)
- .DATA (指定一系列数据)
- .DECLARE (定义由子程序或函数语句说明的过程名称和参数列表)
- .DEF type (设置变量的默认类型)
- .DIM (为数组确定维数或申明变量类型)
- .OPTION BASE (设置0或1作为数组元素的最小下标)
- .SHARED (申明可以由模块级代码中函数、子程序、及语句共享的变量)
- .STATIC (申明可以在函数或子程序调用过程中能保留值的局部变量)
- .TYPE...END TYPE (定义新的变量类型)

语句格式的最后部分为注释。注释以单引号或 REM 开头,用来说明程序行的含义。此外,注释也可单独为一行,放在程序段的前面,用来说明该程序段的功能。

Quick BASIC 程序行中的各个部分都可以为选择项。若都缺省,则为空行。空行也可以插入程序段的任何地方,作为各分段程序的间隔,以增加程序的可读性。

1.3 数据类型

Quick BASIC 不但有系统定义的基本数据类型,而且还可以让用户根据需要定义数据类型。在这里仅介绍基本数据类型,用户自定义数据类型在数据文件一章中做介绍。

基本数据类型分为字符串型和数值型两种数据。

1. 字符串型

组成字符串的字符可以是任何可打印字符,包括常用英文字符和汉字。字符串分为变长字符串和定长字符串两种。

(1) 变长字符串

变长字符串的长度为 0~32767 个字符。

(2) 定长字符串

定长字符串含有确定个数的字符,最大长度不超过 32767 个字符。

2. 数值型

Quick BASIC 的数值型数据分为整型数和浮点数两类。其中整型数又可分为整数和长整数,浮点数可分为单精度浮点数和双精度浮点数。

(1) 整型数

Quick BASIC 中的整型数是指不带小数点和指数符号的数。在计算机内部,它以二进制补码的形式来表示和参加运算。整数的存储长度为两个字节(16 位),数的表示范围是 $-32768 \sim +32767$ 。长整数为四个字节(32 位)存储长度,范围为 $-2147483648 \sim +2147483647$ 。

例如,整数 9 和 -9 分别用下面的形式存放:

0000000000001001

1111111111110111

长整数 9 和 -9 分别用下面的形式存放:

00000000000000000000000000001001

11111111111111111111111111110111

(2) 浮点数

Quick BASIC 使用 IEEE (Institute of Electrical and Electronics Engineers, Inc) 格式的浮点数,它比早期版本使用 Microsoft 二进制格式给出的结果更精确,并能使用 80x87 数学协处理器。

浮点数由三部分组成:符号、指数以及尾数。指数分别用“E”表示单精度和“D”表示双精度。浮点数的符号均占一位。

单精度浮点数以四字节形式存储(32 位)。指数部分除了符号占一位外,数值占 7 位;尾数部分除了符号占一位外,数值占 23 位。单精度数可以精确到 7 位十进制数。

单精度负数的范围为: $-3.402823E+38 \sim -1.401298E-45$ 。

单精度正数的范围为: $+1.401298E-45 \sim +3.402823E+38$ 。

双精度浮点数以八字节形式存储(64 位)。指数部分除了符号占一位外,数值占 11 位;尾数部分除了符号占一位外,数值占 51 位。双精度数可以精确到 15 位或 16 位十进制数。

双精度负数的范围为:

$-1.797693134862315D+308 \sim -4.940656458412465D-324$

双精度正数的范围为:

$+4.940656458412465D-324 \sim +1.797693134862315D+308$

除了上述的十进制数以外,Quick BASIC 中还使用十六进制数(前面冠以 &H)和八进制数(前面冠以 &O)。此外,还有逻辑型数据,逻辑型数据只取“真”和“假”两种值,在不至于混淆的情况下,通常把逻辑型数据也看作是数值型数据。