

方鸿辉 沈美新 译

Logo 语言 入门



上海教育出版社

LOGO 语言入门

方鸿辉 沈美新 译

上海教育出版社

内 容 提 要

本书是介绍计算机“智能化”高级语言 Logo (Apple 文本) 以及使用这种语言进行程序设计的入门书。

全书共分十章。第一、二章介绍用海龟在显示屏上作图和关于过程的基础知识；第三章提供了数十个海龟图形设计实例；第四章讨论如何进行文件管理和程序调试；第五、六章介绍了 Logo 语言中的数、字和表以及如何运用它们进行程序设计；第七、八章介绍了递归调用和动态海龟的知识；第九章讨论了进一步使用表结构编制程序；第十章汇编了 Logo 基本命令。附录部分提供了有关 TI Logo 与 Terrapin Logo 的资料。

本书图文并茂, 内容充实, 语言通俗易懂, 分析深入浅出。书中提供不少程序实例, 便于读者学习和掌握这种新型高级语言。

本书的读者对象主要为广大计算机语言的初学者, 尤其适合中等学校师生。

LOGO 语言入门

方鸿辉 沈美新 译

上海教育出版社出版

(上海永福路 123 号)

总发行所上海发行所发行 江苏溧阳印刷厂印刷

开本 787×1092 1/32 印张 9 字数 195,000

1985 年 7 月第 1 版 1985 年 7 月第 1 次印刷

印数 1—46,000 本

统一书号: 7150·3418 定价: 1.40 元

前 言

Logo 是教育原理的名称,也是不断发展的计算机语言家族中一名新成员的名称。它向人们展示了:怎样将计算机与科学教育、人工智能等建立起联系,促使人们更好地学习和进行智力开发。

Logo 语言是在一九六七年由美国麻省理工学院佩帕特教授指导下的人工智能研究小组所开发的。它比 BASIC 语言更容易学习和掌握,能够使人们跨越“数学才能”的障碍,理解程序设计思想,自己动手编制各种应用程序。因此, Logo 语言具有十分广泛的使用者范围。在国外,广大青少年尤其乐意采用 Logo 语言来学习程序设计,解算各类问题。如果说, BASIC 语言是作为培养程序员的入门课程而开发的语言,那么, Logo 语言应该说是以青少年和非计算机专业人员为对象而设计的计算机语言。我们相信,对于我国广大读者来说,也只需要学习少量英语词汇,就能很好地掌握这种语言。

Logo 语言虽然结构十分简单,却具有相当优异的功能和丰富的表达方式,是一种通用的程序设计系统。Logo 语言具有下述特点:

- Logo 是一种面向过程的语言。程序由称为过程的命令群组成,它的每一步既可以是原始的 Logo 命令,也可以是使用者自定义的过程。

- Logo 是交互式程序设计语言。任何 Logo 命令,不论是构成语言的,还是定义过程的,都能简单地通过键盘输入。

- Logo 的数据体包括数、字和表。表是一种将数值、词汇(字符串)等自由排列而形成的格式。过程能以表的形式进行

自我描述和处理。

● Logo 最重要的特点是：过程可以自我调用，构成递归。递归调用对编制结构化的高水平程序相当有效。

● Logo 引人注目的特色是：“海龟”在屏幕上作图。控制海龟作图不需要具有坐标系等数学知识，人们可凭直觉进行构图设计。

● Logo 语言有效地利用了人工智能的研究成果，十分重视符号处理，并借助于表处理以及启发式探索、开发情报检索、演绎推理和自然会话等系统。Logo 还具有相当强的数据处理能力。

目前，麻省理工学院计算机科学研究中心的教育电脑小组又在设计一种 Logo 的后继系统，以便这种语言适合于八十年代新一代的家用电子计算机。

可以预见，在今后的几年，计算机的发展和普及必然是令人兴奋的，它将同电视一样普及。因此，适合于非专业计算机人员使用的计算机语言的普及工作也就变得日益重要。

为了尽快向我国读者介绍和普及 Logo 语言，我们根据 Abelson, H. 所著的 Apple Logo 并参照其他有关 Logo 外文资料匆匆编译了这本《Logo 语言入门》。希望它能象一块抛入水中的石子，激起一些水波，对普及计算机语言工作起到一些推动。由于本册子中介绍的 Logo 是适用 APPLE 机的版本，因此，对 APPLE 机的应用开发也许能起到一定的作用。由于时间仓促，理解甚浅，错漏难免，恳请广大读者指正。

本册子初稿完成后承蒙上海师范大学张世正老师审阅，并提出不少宝贵意见，谨在此表示衷心的感谢。

译 者

一九八四年十月

目 录

前言

1. 初学 Logo	1
1.1 使用 Logo 之前	2
1.2 使用 Logo 命令	4
1.2.1 基本海龟命令	6
1.2.2 纠正打印错误	8
1.2.3 出错信息	10
1.2.4 用命令进行实践	11
1.3 过程入门	14
1.3.1 简单过程	14
1.3.2 用 Logo 屏幕编辑模式来定义过程	16
1.3.3 过程中的错误	22
1.4 其他绘图命令	23
1.4.1 彩色绘图	24
1.4.2 控制环绕方式	26
1.5 使用屏幕的方法	26
1.5.1 非作图模式	27
1.5.2 定义模式	27
1.5.3 编辑模式	27
1.5.4 绘图模式	27
2. 用过程来编程	29
2.1 带输入的过程	29
2.1.1 选择输入	31

2.1.2	作为专用名称的输入	32
2.1.3	弧过程	36
2.2	重复和递归	41
2.2.1	谈谈递归	43
2.2.2	条件命令和 STOP	45
2.2.3	较复杂的递归	47
2.2.4	画树	51
3.	海龟几何图形设计例选	56
4.	工作空间、文件编排和调试	88
4.1	管理工作空间	88
4.1.1	PO	89
4.1.2	ERASE	90
4.1.3	程序包	91
4.1.4	关于编辑命令的其他用法	92
4.2	Logo 文件系统	93
4.2.1	磁盘文件	93
4.2.2	文件系统的进一步使用	96
4.3	打印过程清单	97
4.4	辅助程序调试	97
5.	数、字和表	100
5.1	数和算术	100
5.1.1	指数记数法	102
5.1.2	取整操作	103
5.2	输出	104
5.3	字	108
5.4	表	111
5.5	命名	115
5.5.1	局部名称和全局名称	118

5.5.2	自由变量	119
5.6	条件表达式和判断	121
5.7	Logo 句法细节	126
5.7.1	Logo 如何将行分离成字	126
5.7.2	使用括弧	127
6.	运用数、字和表进行设计	131
6.1	算术测验程序	131
6.2	随机语句的产生	134
6.3	一种游戏程序	136
6.3.1	子目标计划	137
6.3.2	一个简单的记分员	139
6.3.3	机械游戏员	142
6.3.4	装饰与修改	145
6.3.5	有关 NIMPLAY 程序的清单	147
7.	写交互作用的程序	149
7.1	控制屏幕输出	149
7.2	键盘输入	151
7.2.1	适合小朋友的立即响应程序	152
7.2.2	对正在运行中程序的键盘控制	153
7.3	动态海龟程序	154
7.3.1	什么是动态海龟	154
7.3.2	用动态海龟进行的交互作用	156
7.3.3	改变动态海龟的行为	157
7.4	用游戏操纵杆进行输入	160
8.	输入、输出和递归	162
8.1	字和表的递归	164
8.2	使用表的递归程序	171
8.2.1	LENGTH 程序	171

8.2.2 PICK 程序	173
8.2.3 MEMBER?判断	175
8.3 数制转换	177
9. 表的进一步应用	181
9.1 分层结构	182
9.1.1 表操作	183
9.1.2 关联表	188
9.2 作为数据的程序	191
9.2.1 RUN 命令	191
9.2.2 DEFINE命令	195
9.2.3 TEXT 命令	200
9.2.4 增加新的程序设计的结构	201
9.3 使用表的更高级设计	203
9.3.1 DOCTOR 程序	204
9.3.2 ANIMAL 程序	208
10. Logo 基本命令汇编	221
10.1 绘图命令	221
10.2 数字操作	227
10.3 字和表的操作	230
10.4 定义和编辑过程	234
10.5 条件表达式	236
10.6 使用条件表达式的判断	238
10.7 控制过程的执行	239
10.8 输入和输出	240
10.9 命名	243
10.10 管理工作空间	245
10.11 辅助调试命令	248
10.12 编辑命令	249

10.13 其他控制字符	252
10.14 有关处理特性的原始命令	253
10.15 杂用命令和高级命令	254
10.16 出错信息	258
附录	262
I 软磁盘的格式化	262
II Apple Logo 与 TI Logo	264
III Terrapin Logo	269
参考书目	278

1 初 学 Logo

“Logo”一词源于希腊文，原意为“符号”、“文字”或“思考”、“想法”。设计者之所以用它来命名一种计算机语言，是有其深刻的寓意的。

六十年代对儿童进行的计算机辅助教学的实践，使人们认识到，计算机辅助教学的目的，不应该是仅仅让计算机来教育儿童，更重要的是要让计算机帮助儿童学会如何思考！使儿童通过程序设计来掌握有关计算机的各种概念，从而使儿童也具有应用计算机解决问题的能力。为此，从1967年起，美国麻省理工学院（MIT）佩帕特（Seymour Papert）教授指导下的一个研究小组开发了 Logo 语言。

佩帕特教授预见到，在不久的将来，计算机会同电视机一样普及，因此，他们设计的语言不仅可供儿童使用，也考虑到非计算机专业人员的需要。使用这种语言，即使缺乏数学物理知识的人，也能够凭感觉把握计算机，能够理解程序设计。1979年，麻省理工学院 Logo 研究小组又发展了 Apple Logo（可用于 Apple II 和 Apple II Plus 微型计算机）及 TI Logo（可用于 TI 99/4 微型计算机）等，这样，就使得 Logo 语言的进一步普及有了广泛的基础。

本章主要介绍使用 Logo 的一些基本技巧，包括怎样执行简单的命令以及定义和编辑简单的过程、怎样绘图等。为

了提高初学者,特别是广大青少年的学习兴趣,所有的例子都给出了用海龟几何方法在显示屏上作的图形。

1.1 使用 Logo 之前

如果你以前没有用过计算机,那么首先需要对照着英文打字机的键盘逐步熟悉一些计算机键盘的特性。需要注意的是,不要误将字母“o”的键当作数字“0”来按,同样,不要将字母“l”的键当作“1”。这些看起来似乎很相象,然而这些键对计算机来说是意味着完全不同的意义。当然,计算机键盘通常还包括一些普通英文打字机上所没有的键。在 Apple II 机上有一个标有 CTRL(control,英语中“控制”一词的缩写)字样的键,其用途有点象一个交替转移键用以键入所谓的控制字符。也就是说,为了键入“Control G”,你可以把 CTRL 键按下,并按下 G(也即同时按下 CTRL 和 G 键)。本书中,我们详细说明了使用前缀“CTRL”的控制字符,即“CTRL-G”。在 Apple 机上,其他一些有特殊功能的键是: ESC, REPT, RETURN, RESET 以及带箭头的键。Logo 也使用打开和关闭括弧符“[”和“]”,但它并不标记在某一键上,而是用键入“SHIFT-N”和“SHIFT-M”来表示。这里需要特别提出注意的是:决不要按 RESET 键,它会使 Logo 源程序毁坏,并退到 Apple 监控程序。因此,如有可能的话,在使用 Logo 语言时,RESET 键应被封住。图 1.1 是 Apple II 的键盘以及使用于 Logo 语言中的一些特殊键的说明。如要更详细地了解键盘的使用方法,可以参考 Apple Logo 技术手册。该技术手册也提供了如何在 Apple 机上引进和开工 Logo 系统的详细说明(关于这一点,4.2.1 节也作了简要介绍)。

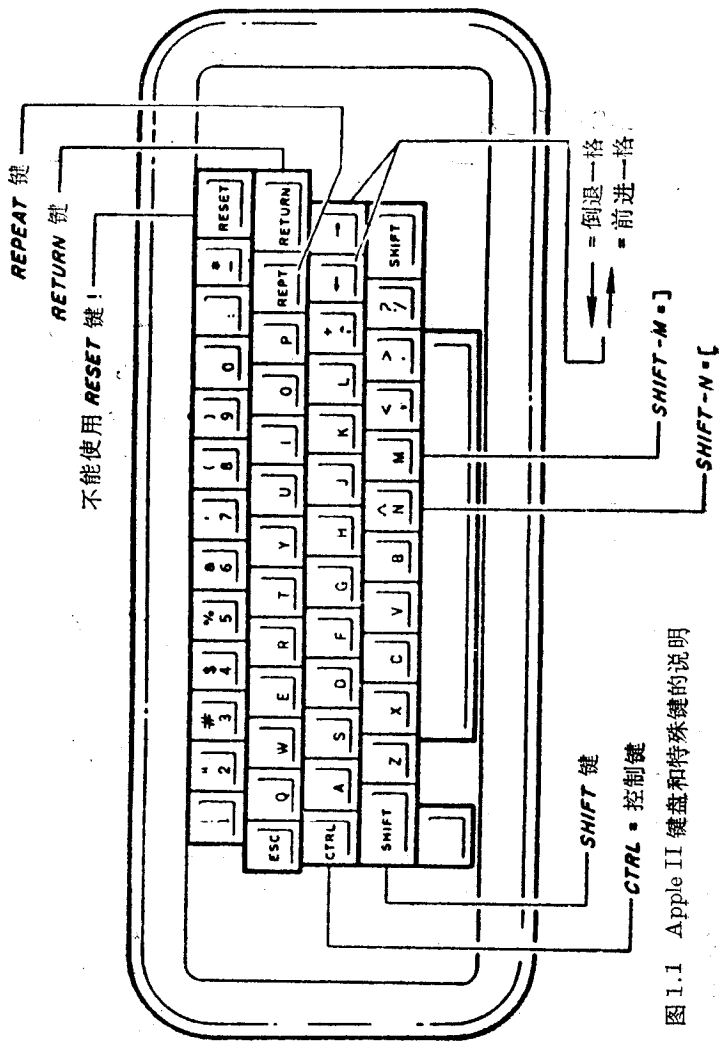


图 1.1 Apple II 键盘和特殊键的说明

1.2 使用 Logo 命令

图 1.2 是 Logo 系统开工后出现在 Apple 机显示屏上的图象。该系统打印出一条欢迎语句，紧接着的一行是以一个问号开始。这个问号也称作提示符，说明 Logo 等待你的命令。



图 1.2 在 Logo 初次开工后 Apple 显示屏上所出现的内容

要给予 Logo 一个命令，首先必须键入这个命令，然后按 **RETURN** 键。例如，你指示 Logo 打印 37 乘以 67 的乘积，你可以键入这样一行命令

```
PRINT 37 * 67
```

也就是说，你可以依次按 **P**、**R**、**I**、**N**、**T**、空格、**3**、**7**、空格、*****、空格、**6**、**7** 和 **RETURN** 键。然后，计算机就会打印出 2479。接着就转入带有问号提示符的新的一行，表示准备接受新的命令。须记住的是，当你键入一行命令，在未按 **RETURN** 键之前，它是并不执行的。要指示 Logo 打印“Logo 是一种计算机语言”，你可以键入这样一行命令

```
PRINT [LOGO IS A COMPUTER LANGUAGE]
```

随后按 **RETURN** 键。这个例子说明了在 Logo 语言中是如何运用方括弧来将“字”组成“表”的。在 Apple 机键盘上，打开和关闭方括弧只要分别按 **SHIFT-N** 和 **SHIFT-M** 键。这样，你可以运用表在显示屏上打印出各种信息。然而，在 Logo 语言中，表还有许多其他的用途，有关的详细内容我们

将在第5章和第9章中学习。

在一个命令行中，空格是很重要的，因为它在 Logo 语言中表示一行命令是如何被分成各个部分的。Logo 语句中，有一些关于“一行在什么地方分成各部分才是合理”的规定，甚至在不必要用空格键隔开的场合。例如，由 5 个字符 `37*67` 编成的字符串中，很明显地包括三个部分：数字 37、乘号 * 和数字 67。然而，在将一行命令分成各部分时，经常运用空格是一个很好的习惯，即使在完全不必要的时候。确定什么时候要用空格的规则将在 5.7 节中讨论。

如果你在键入上述第一行命令时，在 T 和 3 之间漏了一个空格，如下所示：

```
PRINT37 * 67
```

那么，Logo 就会认为你是想要它执行命令：`PRINT37`，并用抱怨它不知怎样做的出错信息来回答你：

```
I DON'T KNOW HOW TO PRINT37
```

图 1.3 显示出当你输入了上述 3 行命令后，在 Apple 机显示屏上出现的图象，以及计算机对每一行命令的回答。出现于每一行命令前的问号是由 Logo 打印出的提示符，行中其余的是用户键入的命令。在本书中，当我们想强调你键入的字符和 Logo 打印出的字符之间的差别时，我们往往将后者印成斜体。例如，图 1.3 中的第一行命令执行后将打印如下：

```
?PRINT 37 * 67
```

```
2479
```

在以后各章中，我们将了解怎样书写使用数字和文字的 Logo 程序。然而，我们是通过向称作海龟的“动物”发出命令来研究怎样使用计算机在显示屏上产生图象，从而开始我们有关 Logo 语言的学习的。为了建立能作图的屏幕，须键入



图 1.3 键入计算机的三行命令及系统的回答

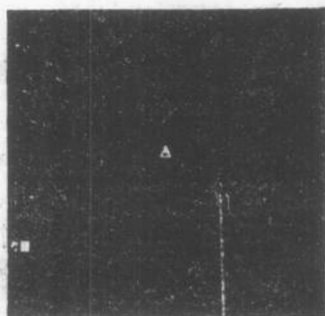


图 1.4 当 Logo 进入作图模式时显示屏上出现的图象

CLEARSCREEN 命令并按 **RETURN** 键。此时，显示屏上会出现如图 1.4 所示的情况，除了中央有一个小三角，也就是所谓海龟，靠近底边有一个问号提示符外，整个显示屏上一片空白。如前所述，这个问号提示符是指示出 Logo 准备接受命令。在作图时，Logo 保留显示屏底部 4 行以供显示你键入的命令和计算机作出的回答。显示屏的其余部分供作图用。当 Logo 用此法在显示屏上作图时，系统就被称作处于作图模式，而在图 1.2，1.3 所示情况下，显示屏是并不留出空间供作图用的，此时，称作非作图模式。当 Apple 机处于非作图方式时，可以有 24 行供打印显示。有关 Logo 系统操作的不同模式，将在 1.5 节中作比较完整的介绍。

1.2.1 基本海龟命令

海龟是 Logo 进入作图模式时出现在显示屏中央的三角形点阵，你可以通过让海龟移动并留下它的踪迹来作图。要使海龟移动，通常有 4 条基本命令。命令 FORWARD 和 BACK 使海龟作直线运动。每次你给出一个 FORWARD 或 BACK 命令，同时给出一个数值以告诉海龟应移动多远。命

令 RIGHT 和 LEFT 将使海龟旋转。RIGHT 和 LEFT 中的每一个命令都须给出旋转的度数。试键入以下一系列 Logo 命令：

```
RIGHT 45  
FORWARD 100  
LEFT 135  
FORWARD 200
```

以上命令会产生出图 1.5 所示的楔形图形。须记住：用 **RETURN** 键来结束每一行命令，并注意在命令字与数字之间留以空格。如果你打错一个字符，可以通过按 `|←|` 键来删除这个字符。要详细了解有关纠正打印错误的方法，请看 1.2.2 节。

命令后面的数字称作输入，FORWARD, BACK, LEFT 和 RIGHT 每个命令后

面都需要带一个输入项。不同的 Logo 命令中，可以有也可以没有输入项。CLEARSCREEN (清除屏幕) 就是一个没有输入项的命令的例子。接下来，我们来看一些带有一个输入项的命令的例子。

如果你想要移动海龟而不画出线条的话，则可用 PENUP (提笔) 命令。然后，用 FORWARD 和 BACK 命令就能不留踪迹地使海龟立即移动。若要恢复画线条，可用 PENDOWN (落笔) 命令。PENUP 和 PENDOWN 都不须附带输入。HIDETURTLE (藏海龟) 命令使海龟在显示屏上消失，

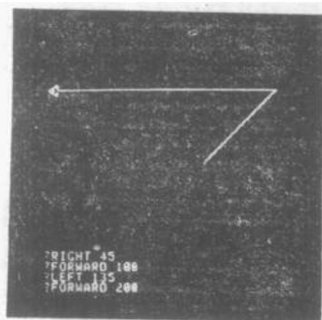


图 1.5 显示屏上显示出一个简单的海龟命令的结果