

结构化 TRUE BASIC 程序设计

〔美〕J.G.凯梅尼 T.E.库尔茨 著

秦笃烈 译 罗晓沛 校

北京大学出版社

结构化 TRUE BASIC 程 序 设 计

[美] J. G. 凯梅尼 著
T. E. 库尔茨

秦笃烈 译

罗晓沛 校

北 京 大 学 出 版 社

内 容 简 介

本书是 BASIC 语言的创始人凯梅尼(Kemeny)和库尔茨(Kurtz)于 1987 年的最新佳作。它是基于美国 1987 年国家标准程序语言(X3.11-1987. ANSI)的结构化版本所写成的。

本书系统地介绍了结构化程序设计。全书共分四部分。第一部分为结构化程序设计导论;第二部分为文本应用,涉及文本处理,文件应用,检索和排序,数据库系统;第三部分为数学应用,涉及初等数论,模型(蒙特卡洛法,资源最佳分配,突变遗传,随机模型等),计算机模型(包括生态,人口及学院管理等模型);第四部分为其他应用,涉及商业(预算方案,关键路径法,菜单驱动应用等),游戏和人工智能(九宫排列的各种解法,尼姆游戏,模式匹配游戏等),高级图形(图象,各种独特作图命令用法,两个图形库,递归,人工智能八皇后问题求解及图示,动画制作法等),声音和音乐(声音和音乐产生法,乐谱作图法,计算机为世界名曲配和声的方法等)。本书以程序表演为基础,列出全部可执行程序清单。内容新颖,趣味盎然,图文并茂,深入浅出,熔学术性,实用性,普及性及趣味性于一炉。无论对于初学者或程序设计专业工作者,都是一本难得的教材和参考书。可作为大学非计算专业程序设计课程的教材。

John G. Kemeny, Thomas E. Kurtz
Structured BASIC Programming
John Wiley & Sons, 1987

结构化 BASIC 程序设计

[美] J. G. 凯梅尼 著
T. E. 库尔茨

秦笃烈 译

责任编辑:邱淑清

*

北京大学出版社出版
(北京大学校内)

北京大学印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

787×1092 毫米 16 开本 24 印张 548 千字

1990 年 1 月第一版 1990 年 1 月第一次印刷

印数:0001—6000 册

ISBN 7—301—01001—X/TP·028

定价:11.00 元

序 言

学生学习程序设计的动机至少有三种。第一,也可能是最不重要的,他们中的一小部分要成为专业程序员。第二,计算机程序无疑是表达和研究涉及计算机科学,数学或其它任何学科的算法的最有效方法。第三,关于计算机程序的组织、设计和写作的训练问题的解决要靠个体化方式。写作大的计算机程序的经验可使学生增加解决跨学科问题的技巧和能力。

本书是针对有第二和第三种学习动机的人写的。在本书的前十一章介绍程序设计思想,在其后的部分是论述来自许多领域——模型、模拟、商业、数学、音乐等的算法和程序。由此学生可以了解来自这些领域的算法过程如何成功地转变为计算机程序。本书把大程序写作教学作为培养解决问题能力的重要方面对待,集中讨论子程序概念,把外部子程序集集成库的方法以及有时称为自顶而下的分层次设计方法。

对于非计算机专业的学生,可以说结构化程度很好的 BASIC 是适应这种教学目的的最佳语言。众所周知,它是世界上使用最广泛的语言。由于有了满足结构化程序设计需要的各种结构及外部子程序库,BASIC 成为能写非常小和非常大的应用程序的极佳的软件环境。

本书所用的 BASIC 版本是 True BASIC^①,它是基于美国国家标准程序设计语言全 BASIC(X3. 113—1987, ANSI, 1430 Broadway, New York, N. Y.)的结构化版本。True BASIC 提供 BASIC 的传统结构(诸如 GOTO 和 IF-THEN-GOTO 语句),但也提供许多结构化的构造(如 DO-LOOP 和 SELECT-CASE)。此外,它提供可带参数的内部和外部子程序。最后但并非不重要的一点是,BASIC 提供与象素无关的作图方法。

没有亲自实践经验要学会程序设计是不可能的。学生必须经常操作个人计算机和使用 True BASIC 语言。书中所列程序的磁盘可以在任何运行 True BASIC 的计算机上使用。

在学习每一章时,应该调用并试验书中的程序,通过修改它们加深体验并自己动手写类似的程序。每章之后均指定一些习题。它们是必须在计算机上写和试验的计算机习题。

本书的宗旨不是介绍硬件。需要的一点硬件知识仅仅是:如何装软盘,如何打入程序和进行修改以及如何运行程序。True BASIC 是一种独立于硬件之外的语言;同样的 BASIC 程序可以在各种计算机上运行。教太多的特定计算机品种的性能只能冲淡本书的宗旨。

全书内容分为四部分:

- 第一部分 程序设计入门;
- 第二部分 文本应用;
- 第三部分 数学应用;
- 第四部分 其它应用。

① True BASIC[®]是 True BASIC 公司的注册商标。

每一部分分为若干章。第一部分的设计目的是为了循序渐进地学习并保持内容的完整性。第二至第四部分互相独立,它们仅依赖于第一部分。

本书通篇强调子程序的重要性。可以访问若干外部子程序库,它们已包括在伴随的软盘中。也强调要有良好的结构和风格,然而要力求简洁,程序中的注解要少而精。

虽然本书不要求有 BASIC 的预备知识,但它不是 BASIC 的完备教程。只有用到的细节才加以讨论。对于没有讲到的一些内容,教师可以在讲授时加以补充或参见所用 BASIC 的参考手册。本书假定学生具有高中以上代数知识。

第一部分(“程序设计入门”)的十一章向读者介绍的是最初步的 BASIC 短小程序,在第十一章(“如何进行程序设计”)给出了如何对大型 BASIC 程序进行设计、编码、测试和调试的一般建议。在这些内容中,读者可以学会各种结构化结构的基础以及自定义函数和各种外部、内部子程序的用法,也介绍了数和字符串表和表格以及用极为方便的 MAT 指令进行输入和打印,并均附有实例。对文件也作了介绍并给出了像画点、线和文本这些简单作图方法。

在第一部分,在章的小结中偶而增加了一些事实,如合法的变量名规则,算术运算的括弧和优先级规则等,它们如在正文中叙述则会显得内容零乱。

第二部分处理文本应用,一共四章。第十二章以典型的应用更详细地说明字符串操作。第十三章讨论几种文件应用,其中有些是全部,有些是部分地处理字符串。第十四章讨论检索和排序算法,各给出一个简单的和一个高效率的算法,它们归结到如同为一本书建立索引那样的索引应用。第二部分的高潮是第十五章介绍的一个小型数据库检索系统。这个应用说明了许多设计好程序的原则而这本身也就是它的有用的应用。

第三部分涉及数学应用。第十六章处理三角学、方程求根和素数的基本数学思想以及强有力的矩阵运算 MAT 指令。第十七章涉及随机模拟模型。通过介绍简单的抛硬币、掷骰子和发扑克牌模拟归结到棒球队击球的实际模型以及个人计算机特殊的存贮管理系统的行为研究。第十八章介绍非随机计算机模型。在讨论一个生态学经典模型之后讨论基于分股的人口模型以及一个具有复杂反馈机制的小型大学模型。

第四部分以第十九章作为开始,集中讨论财经方面应用,诸如预算方案设计,复利计算以及关键路径分析法。第二十章讨论井字棋和尼姆棋这些博弈问题。第二十一章继续深入讨论第七章中已介绍的作图方法包括可作变换和动画的图象子程序。本书的最后一章是第二十二章,它处理音响问题,它介绍如何产生声音,如何利用第二十一章的作图方法画五线谱,并介绍了一个创作音乐和声的程序。

在此,我们谨向为开发结构化程度很高的 BASIC 版本作出贡献以及对本书的思想施惠甚多的同人表示谢意。我们特别是感谢以前的两位同事对本语言所作的重大贡献:A. W. 路尔曼在 70 年代初在达特蒙思学院设计了一种和设备无关的作图方法。S. J. 加兰德约在 1974 年在达特蒙思学院设计了 BASIC 第一个结构化版本并担任负责设计标准 BASIC 的 ANSI X3J2 委员会的付主席。

J. G. 凯梅尼

T. E. 库尔茨

1987 年 1 月

译者序

本书的译出基于如下的想法:

计算机科学如同一棵枝繁叶茂的大树。在这棵树的每一层次上迅速地冒出新的枝叶。知识更新的速度是惊人的。BASIC 语言在全世界流传,但是它的发明人在不断探索和改进。他们的努力结果是产生了 BASIC 美国国家标准。到目前为止,基于美国国家标准的 True BASIC 已从 85 年推出时的 1.0 版发展到 2.03 版。在语句、功能、结构化程度、与 DOS 接口以及不同语言或应用程序的 .COM 和 .EXE 程序在它环境下的自由调用等方面均有所突破。对广大 BASIC 语言爱好者软件升级和进一步开发无疑是好消息。对好的东西应该尽快引进、吸收,为我们所用。

译者 1988 年 1 月得见本书深感有很高应用价值,于 3 月下旬译出初稿并对书中程序作试运行。在翻译过程中对原书的某些错误作了订正,有个别程序中为使读者阅读更加方便,在不影响总效果的前提下作了技术性加工。本书程序众多、应用性强,为方便读者,特配有程序软盘并委托北京大学出版社发行。

对本书的出版我必须作如下致谢:

对中国科技大学研究生院罗晓沛教授仔细校阅全书并作了推荐,对此深表谢意。

感谢北京大学数学系李忠教授以及计算机科学技术系杨芙清教授的热忱支持。

秦笃烈

1988 年 10 月 29 日于首都医学院

1989 年 1 月 14 日修改

目 录

序言	I
译者序	II

第一部分 程序设计入门

第一章 BASIC 起步	3
1.1 简述	3
1.2 一个非常简单的例子	3
1.3 询问	4
1.4 进一步谈谈 PRINT	6
1.5 小结	7
习题	7
第二章 循环	9
2.1 简述	9
2.2 简单的循环	9
2.3 求平方根	10
2.4 嵌套的循环	13
2.5 小结	14
习题	14
第三章 选择	15
3.1 简述	15
3.2 IF 语句	15
3.3 二次方程	18
3.4 猜数游戏	20
3.5 退出循环	22
3.6 小结	23
习题	25
第四章 FOR 循环	27
4.1 简述	27
4.2 最简单的情况	27
4.3 最高级的用法	29
4.4 嵌套的循环	33
4.5 小结	35
习题	35
第五章 表和数据	37
5.1 简述	37

5.2	数表	37
5.3	READ 和 DATA 语句	41
5.4	更多的实例	42
5.5	小结	44
	习题	44
第六章	字符串	46
6.1	简述	46
6.2	字符串变量	46
6.3	字符串运算	48
6.4	字符串表	52
6.5	小结	55
	习题	56
第七章	图形	57
7.1	简述	57
7.2	点和线	57
7.3	坐标系的建立	58
7.4	窗口	60
7.5	彩色图形	63
7.6	多窗口作图	65
7.7	小结	67
	习题	68
第八章	文件和表格	70
8.1	简述	70
8.2	数据文件	70
8.3	表格	72
8.4	进一步谈谈文件	74
8.5	小结	77
	习题	77
第九章	函数	79
9.1	简述	79
9.2	内部数值函数	79
9.3	内部字符串函数	81
9.4	单行自定义函数	82
9.5	多行自定义函数	83
9.6	申明语句 DECLARE DEF	85
9.7	小结	86
	习题	86
第十章	子程序	88
10.1	简述	88
10.2	内部子程序	88
10.3	外部子程序	91

10.4 外部函数	96
10.5 小结	96
习题	97
第十一章 如何进行程序设计	98
11.1 简述	98
11.2 制订规范	99
11.3 程序设计	100
11.4 代码设计	102
11.5 结构化和测试	106
11.6 调试	107
11.7 小结	108

第二部分 文本应用

第十二章 文本处理	111
12.1 简述	111
12.2 字符串的基本运算	111
12.3 意外情况处置	114
12.4 一个文本处理实用程序	117
12.5 一个万年历程序	123
12.6 小结	127
习题	128
第十三章 文件应用	129
13.1 简述	129
13.2 一种简单的记分应用	129
13.3 一个字处理实用程序	134
13.4 一个主文件修改程序	139
13.5 小结	146
习题	147
第十四章 检索和排序	148
14.1 简述	148
14.2 简单的检索	148
14.3 效率更高的检索	149
14.4 排序	151
14.5 排序效率测量	153
14.6 分类合并	154
14.7 一种索引排序	159
14.8 一个索引程序	162
14.9 小结	167
习题	167

第十五章 一个数据库系统	169
15.1 简述	169
15.2 数据组织	169
15.3 程序的组织	171
15.4 主程序	172
15.5 命令程序	174
15.6 数据的检索	176
15.7 数据的操作	178
15.8 实用程序	179
15.9 几个低层子程序	182
15.10 运行样本	183
15.11 库	185
习题	188
进一步设计思考题	189

第三部分 数学应用

第十六章 初等数学	193
16.1 简述	193
16.2 三角学	193
16.3 方程的根	197
16.4 素数	200
16.5 MAT 指令	204
习题	208
进一步设计思考题	209
第十七章 模拟	210
17.1 简述	210
17.2 随机数	210
17.3 发牌	213
17.4 突变的传播	218
17.5 蒙特卡洛法	220
17.6 棒球模拟	222
17.7 资源分配	229
习题	231
进一步设计思考题	232
第十八章 计算机模型	233
18.1 简述	233
18.2 一个生态学模型	233
18.3 一个人口模型	235
18.4 一所小型学院的模型	241

进一步设计思考题	250
----------------	-----

第四部分 其它应用

第十九章 财经问题	253
19.1 简述	253
19.2 预算方案设计	253
19.3 菜单驱动程序	259
19.4 复利	261
19.5 关键路径法	269
习题	276
进一步设计思考题	277
第二十章 博弈问题	278
20.1 简述	278
20.2 井字棋:简单版本	278
20.3 井字棋:启发式版本	282
20.4 井字棋:前瞻式版本	288
20.5 尼姆博弈	292
20.6 一种模式匹配游戏	298
习题	305
第二十一章 高级图形	307
21.1 简述	307
21.2 图	307
21.3 BOX 命令	312
21.4 动画	314
21.5 ASK 命令	318
21.6 两个图形库	319
21.7 小结	324
习题	325
进一步设计思考题	326
第二十二章 声音和音乐	327
22.1 简述	327
22.2 声音	327
22.3 PLAY 语句	328
22.4 乐谱作图	329
22.5 乐曲的和声	338
习题	346
进一步设计思考题	347
附录	348
附录 1 ASC II 字符集	348

附录 2 PRINT USING 详介	352
附录 3 程序目录	356
关键词索引.....	365
中英文名词索引.....	367

第一部分

程 序 设 计 人 门

第一章 BASIC 起步

1.1 简述

一个程序是要求计算机完成特定任务的指令集合。有各种各样语言可用来写程序。BASIC是一种容易学会而又强有力的计算机程序设计语言。

用BASIC写的程序由语句组成,每个语句占一行。语句用一个关键词作为开始,向计算机说明它是哪一种语句。为了便于标识语句类型对这些关键词都使用大写(虽然 BASIC 并不在意使用大写或小写字母)。

在本章我们介绍五个最常用的关键词: LET, PRINT, END, REM 以及 INPUT。即便使用这样有限的词汇已经能写出数量惊人的程序。

这些概念是通过简单的例子进行解释的。关于变量如何命名以及程序中可出现哪些数的说明已包括在本章小结中。

1.2 一个非常简单的例子

我们以一个如此简单的程序开始,我们希望它是不言自明。确实,它是如此简单,在袖珍计算器上执行也许会更快一些。但它介绍了三类BASIC语句。我们给它一个名称:DIVIDE。

程序 DIVIDE

```
LET n = 147
LET d = 69
LET q = n/d
PRINT q
END
```

如果运行这个程序,可看到 147 和 69 的商是

2.13043

LET 语句是BASIC不可缺少的语句。头两行的作用是赋值:把数值赋给变量 n(作分子)和 d(作分母)。在第三行语句中 LET 是用来执行计算。它做除法并把答案赋给 q(作商)。注意除法的符号是“/”。LET 语句总是先对右边求值,然后把算出的值赋给左边的变量。因为右边可以包含一个长而复杂的公式,LET 可以为你做大量工作。

PRINT 语句是要计算机在它的屏幕上显示 q 的值(答案)的指令。如果这个语句省略(一种常见的程序设计错误)则计算机知道这个答案,但将不会告诉你。

END 语句有两个作用。它标志程序的物理结束同时当计算机达到 END 时就停止工作。在这个简单的程序中计算机开始时处在第一行,一次执行一行,最后停下来。为了利用现代计算机的威力,我们将需要更复杂的结构。

DIVIDE 是程序的名称,也就作为磁盘上包含这个程序的文件名。

变量名的选择权在程序员。可以用 numerator(分子)代替 n,用 denominator(分母)代替 d 以及用 quotient(商)代替 q。考虑如下的变异:

程序 DIVIDE2

```
LET cat = 147.  
LET dog = 69  
LET horse = cat/dog  
PRINT horse  
END
```

它看起来有点滑稽,但产生的结果和程序 DIVIDE 是完全一样的。计算机并不理解像 numerator 或 cat 这样的单词。它只是理解这是赋值为 147 的变量名。

甚至更简单的程序也能达到相同的目的。例如考虑:

```
LET q=147/69  
PRINT q  
END
```

或甚至:

```
PRINT 147/69  
END
```

这些程序的单调之处在于不管运行多少次,它们总是解这个同样的问题。让我们使这个程序更有趣一些。

1.3 询问

使程序更有用的一种方法是让使用者指定哪些数相除。可以说 PRINT 语句是“告诉我”语句而 INPUT 是“问我”语句。当 BASIC 遇到 INPUT 语句时,它印出一个问号等待你提供需要的信息。

程序 DIVIDE3

```
INPUT n, d  
LET q = n/d  
PRINT q  
END
```

我们要给出程序 DIVIDE3 的两次运行结果。在每次运行中,问号以后的数是由我们打入的。注意两个数必须要用逗号分开。第一个成为 n 的值而第二个是 d 的值。

run

? 147, 69
2.13043

run

? 1001, 32
31.2812

我们转向一个更有趣的例子,它是把摄氏温度变成华氏温度的程序。

程序 CELSIUS

REM Convert from Celsius to Fahrenheit

INPUT n, d

LET f = 9/5*c + 32

PRINT f

END

注意程序开始的 REM 语句。它使计算机执行了什么? 绝对没有! REM 代表“remark”(注解)或“reminder”(提醒),它的设置纯粹是为了程序员的方便,BASIC 对它不予理会。在程序开始用一个几个 REM 语句说明程序要做什么事是一种好作风。还可注意到几个空行,BASIC 也忽视它们。这里它们是为了使主程序分隔开来——很像在打一篇论文时在一段之后留出一空白行。

其它部分是不言自明的。至少你应该知道怎样把摄氏变华氏。如果不知道计算机就不能帮助你。你,或其它知道公式的人,在计算知道怎样作转换之前必须写这样的一个程序。

我们给出两次运行。第一次表明华氏 98.6°(人体正常体温)的原值——它的摄氏温度是一个整数。另一次仅表明对两种温度都相同的温度值。

run

? 37
98.6

run

? -40
-40