

TURBO PROLOG 程序设计

陈兆乾 潘金贵
谢俊元 编译
陈世福
袁 峰 审校

南京大学出版社

NANJING UNIVERSITY PRESS

TURBO PROLOG 程序设计

陈兆乾 潘金贵 谢俊元 编译

陈世福 袁 峰 审校

南京大学出版社

1991·南京

内 容 简 介

本书以 Turbo Prolog 最新版本为背景,详细阐述了 Turbo Prolog 的基础和程序设计技术。同时还介绍了域结构图和谓词结构图、数据流图和程序结构图等结构化程序设计工具与方法在 Turbo Prolog 程序设计中的应用。此外,以人工智能程序设计的四个热门领域:专家系统、数据库、自然语言处理、难题求解等介绍了 Turbo Prolog 程序的设计和实现。

本书写法新颖,每章开头的简短引言,点缀了内容提要,结尾的小结道出了各章精华,中间使用了大量完整的、有实用意义的小型程序来说明各种程序设计技术(这些程序都已灌制到一张 5 吋的软盘上,能极大地方便读者学习和练习)。此外,每章都提供了数量和难度都适中的练习,可供读者编制程序和上机操作。

全书叙述通俗易懂,是一本学习和讲授 Turbo Prolog 程序设计的合适的教材和教学参考书。适合高等学校的大学生、研究生、教师以及广大的计算机程序员和科技人员阅读、参考。

TURBO PROLOG 程序设计

陈兆乾 潘金贵 谢俊元 编译

陈世福 袁 峰 审校

南京大学出版社出版

(南 京 大 学 校 内)

江苏省新华书店发行 江苏省丹阳新华印刷厂印刷

1989 年 6 月第 1 版 1991 年 5 月第 3 次印刷

开本: 787×1092 1/16 印张: 26 字数: 643 千

印数 8501—16500

ISBN 7-305-00421-9

TP·20

定价: 7.90 元

责任编辑: 顾其兵

出版说明

人工智能是50年代兴起的一门新兴学科。30多年来,研究的进展和取得的成就都十分惊人。因此,它被誉为本世纪的重大科学技术成就之一。

人工智能是计算机科学的一个重要分支,它的研究领域十分广泛,涉及模式识别、物景分析、自然语言理解、数据库的智能检索、博弈、自动定理证明、自动程序设计、专家系统、机器翻译、机器学习、机器人等领域。随着人工智能的基本理论和技术的深入研究和广泛应用,它不仅正在深刻地影响着计算机科学的其他领域,而且对信息科学、控制科学、系统科学、数学、心理学、电子学、生物学、医学、语言学和哲学等学科产生愈来愈深刻的影响,并带来巨大的社会效益。

应广大科技人员、计算机工作者、研究生、大学生系统地学习人工智能的需要,我们在近期内组织出版一套具有特色的人工智能丛书:

《人工智能导论》

《PROLOG 程序设计教程》(即现《TURBO PROLOG 程序设计》)

《TURBO PROLOG 工具库》(已出版)

《知识工程语言与应用》(已出版)

前 言

一、Prolog & Turbo Prolog

Prolog 名字源于术语 **Programming in Logic**。1973 年, 法国马赛大学的 Alain Colmerauer 和 Groupe d'Intelligence Artificielle 的其他成员设计并开发了 Prolog。当时他们的主要目的是编制自然语言的翻译程序。从 1973 年起, 出现了 Prolog 的几种扩充和实现, 其中苏格兰爱丁堡大学小组的工作尤为突出。苏格兰人实现的 Prolog 语言有时称为 C&M Prolog, 这个名字同经典著作《Programming in Prolog》的作者 William F. Clocksin 和 Christopher S. Mellish 有关。尽管 Prolog 还没有统一的标准, 但通常 Clocksin 和 Mellish 的教科书中的定义被看作非官方标准。

自从日本的五代机计划选择 Prolog 作为其核心语言以来, Prolog 戏剧性地日益受到人们的青睐。把人工智能程序与传统的程序融合在一起使用, 这种需求稳步增长, 随着计算机应用的不断发展, 人们对开发 Prolog 应用程序的要求与日俱增。Turbo Prolog 正是为满足这些需要应运而生的。

Turbo Prolog 是一种高级的, 基于编译的人工智能语言, Turbo Prolog 尤其适合于开发专家系统、动态数据库、自然语言处理程序和一般问题求解应用。Turbo Prolog 也具有构造用户友善的应用程序的设施。这些设施包括多窗口, 彩色图形和交互式输入/输出设施。

Turbo Prolog 的内部谓词和扩充功能也使它适合于开发许多标准的事务处理程序, 包括记帐、工资处理和作图程序等。

除了 Turbo Prolog 之外, 在国外还有其他几种 Prolog 语言, 包括 Arity Prolog, Prolog II, Wisdom Prolog 和 Micro Prolog。Turbo Prolog 拥有良好的用户接口、精致的全屏幕编辑程序, 还有多窗口设施和交互式调试环境, 这些都为其他的 Prolog 语言所不及。它支持配有彩色/图形适配器 (CGA) 和扩充图形适配器 (EGA) 的 IBM PC 机的彩色图形功能。同时还支持图形谓词和海龟图形系统。并能处理顺序文件、随机存取文件和二进制文件。

Turbo Prolog 优于其他的 Prolog 语言的另一个原因是 Turbo Prolog 是一种编译型语言, 而其他的一些 Prolog 版本则是解释型的。解释型语言一个接一个地接受源代码指令并执行它们, 而编译程序则把整个源代码程序编译成机器代码。Turbo Prolog 编译源代码的速度是 IBM PC 机上可利用的所有 Prolog 语言中最快的。

Turbo Prolog 在某些方面同上面提到的 C&M Prolog 有所差别, Turbo Prolog 不具有 C&M Prolog 的一些成分, 但这些成分一般仅用于很高级和复杂的程序中。Turbo Prolog 和非官方的 C&M 标准的一个明显区别是 Turbo Prolog 把数据成分严格地类型化。这些区别中的许多有助于提高 Turbo Prolog 编译和执行速度。尽管一些程序员不赞成这些差别, 说 Turbo Prolog 不是一种“纯” Prolog, 然而这种语言却提供了许多扩充功能, 诸如窗口和图形, 而在其他的一些实现中是不具有这些功能的。不管 Turbo Prolog 是不是一种纯

Prolog, 毋庸置疑它是一种灵活的、实用的、具有很多特色的现代程序设计语言。

二、Turbo Prolog 和其他语言

诸如 Pascal, BASIC 和 C 这些语言都是过程式语言。用过程式语言编写的程序是由多个指令块构成的, 这些指令块说明了达到程序目的所采取的步骤。而 Prolog 同这些语言不同, 它是一种陈述式语言, 用陈述式语言编写的程序是由说明程序应达到的目标的逻辑说明组成的, 只需几行 Turbo Prolog 代码就能完成其他程序设计语言可能需要几页代码才能完成的任务。

Prolog 缺乏像 DOWHILE 和 IF... THEN 这些强制的控制结构。程序员说明达到程序目标的逻辑方法而不说明程序执行的步骤, 由功能很强的内部合一程序 (将在第二章讨论它的运行情况) 搜索程序目标的解。

Turbo Prolog 的功能在于它的搜索和模式匹配能力。内部合一程序在试图满足程序员设置的目标时, 它将一视同仁地检查所有可能的程序规则组合。因为 Prolog 是基于熟知的逻辑思想, 因此, 当读者学会如何使用它作程序设计时, 会对它爱不释手。

三、本书的适用对象

本书是为对 Prolog 程序设计感兴趣的任何人, 不管他是初学者还是专业程序员而编写的, 他们既不需要了解像 Pascal, BASIC 或 C 这些过程式语言的程序设计, 也无需了解像 Lisp 这样的一些陈述式语言的程序设计, 便可从本书中获益。本书为读者提供了所需的背景性概念和书写应用程序的方法。

即使读者尚未购置 Turbo Prolog 软件, 本书也将帮助读者了解什么是陈述式语言, 以及怎样按“Prolog 式”的思想来设计和开发程序。

如果读者已经获得了 Turbo Prolog 软件, 本书就更值得一读。本书揭示了如何设计和开发适合用户需要的应用程序, 对书中涉及的每一个问题都开发了完整的程序, 用以演示内部谓词和规则的用法。

本书是学习和讲授 Turbo Prolog 程序设计的一本较适合的教材, 而不是使用手册性的资料。对于 Turbo Prolog 程序员来说, 本书是很有价值的。这本书不仅综合探讨了 Turbo Prolog 谓词, 并向读者表明了诸如数据流图和结构图这些结构程序设计工具的法, 在文件处理程序、动态数据库和专家系统设计和实现中, 这些工具将对读者目前或今后的开发工作以有力的帮助。

本书的叙述和示例是以 Turbo Prolog 的 1.1 版本为背景的。

四、本书内容

本书由两大部分构成。第一部分共有八章, 阐述了 Turbo Prolog 基础和程序设计技术。在每一章中都用一些小型程序来说明这些技术。本书的第二部分以人工智能程序设计的四个主要领域: 专家系统、数据库、自然语言处理、游戏和难题求解, 介绍了 Turbo Prolog 程序的设计和实现。第二部分的每一章都以一个完整的应用程序作为结束, 为读者展示了如何有条不紊和高效地开发程序。另外, 每章都提供了有益的自助练习, 它鼓励读者亲手编制程序和上机操作。

第一章“Turbo Prolog使用入门”，提供了在硬盘和双软盘系统中装配和启动 Turbo Prolog 的完整的说明，并介绍了 Turbo Prolog 的用户接口。

第二章“Turbo Prolog 基本概念”介绍了 Turbo Prolog 的逻辑基础。本章叙述了诸如匹配、实例化和约束，以及回溯等基本概念，帮助读者理解 Turbo Prolog 的内部合一机制，了解为达到程序中确定的目标 Prolog 是如何进行工作的。

第三章“Turbo Prolog 程序设计基础”描述了 Turbo Prolog 程序的结构和语法。这章采用亲自实践的方法，提供了许多短小且具代表性的例子，阐明了基本的程序设计技术。另外，还介绍了两种适用的程序设计工具：域结构和谓词结构图。

第四章“重复和递归程序设计”描述了执行迭代性任务基本技术的两种重复方法：失败一回溯，截断—失败一回溯。三种递归方法：截断—失败、用户定义的循环和一般递归规则。

第五章“表程序设计”向读者展示了表的建立和操作，包括寻找表的一个成员，把一张表一分为二，两张表合二为一，并以一些短程序阐明这些概念和技术。

第六章“字符串程序设计”包括了 Turbo Prolog 的字符集、串和串处理操作。用有代表性的程序阐明了串的分裂和合并，检索串内的字符或子串以及其他的程序设计技术。

第七章“文件程序设计”首先描述了 IBM PC 机的可能的设备构成。接着结合例子程序，阐述了磁盘数据文件的建立和文件的读、写操作。

第八章“窗口、图形和声音程序设计”描述了 Turbo Prolog 的窗口设施，举例阐明了多窗口的建立和窗口的输入/输出。讨论了产生线形图、条形图和扇形图的程序，以及 Turbo Prolog 的海龟图形系统，用 Turbo Prolog 的声音谓词来谱写乐曲。

第九章“建立动态数据库”是涉及应用方面内容的第一章。这章利用前面各章学习过的许多技术来建立和操作关系数据库系统。在这一章中，读者将会看到 Turbo Prolog 强有力的内部匹配程序是如何发挥作用的。

第十章“基于 Turbo Prolog 的专家系统建造”描述了专家系统的组织和建立，讨论了用 Turbo Prolog 建立基于规则和基于逻辑的专家系统的知识库、推理机和用户接口系统。最后编译器结合自己的工作，介绍了用中西文 Turbo Prolog 实现的“勘探地下水专家系统”。本章给出的几个不同的专家系统程序，将使读者受到启迪。

第十一章“自然语言处理”描述了自然语言处理技术和流行的语言分析方法，这章通过例子演示了如何在简单的自然语言的用户接口程序中使用 Turbo Prolog 串和表操作的设施。透过这些例子程序读者有机会了解这些技术并可改造这些程序以适用于自己的目的。

第十二章“游戏和难题求解”提出了 Turbo Prolog 建立游戏和难题求解的一般方法。用一个猜数游戏，两种解法的 23 根火柴游戏，一个模式匹配游戏和基于著名的猴子和香蕉问题的游戏阐明了这些程序设计技术。

本书还包括四个附录，附录 A 是关于使用 Turbo Prolog 全屏幕交互式编辑程序的一个简略的指导。附录 B 讲述了如何使用在本书的其他地方没有讨论的 Turbo Prolog 系统特色和编译控制、跟踪调试等高级程序设计技术。附录 C 是一个简略的 Turbo Prolog 术语汇编。附录 D 列出编写本书的主要参考书目和进一步的读物。

与本书配套的软件盘片除配书实例外，还有最新版本的适合不同配置的 PC 机使用的 Turbo Prolog（含中西文、纯西文两种版本）系统软件和工具库软件。

本书由陈兆乾(第4,7,8,9,12章)、潘金贵(前言,第1,2,3,10,11章,附录C、D)和谢俊元(第5,6章,附录A、B)同志合作完成。

本书的编译工作是在陈世福副教授的主持下完成的,并由他主审。袁峰博士作了校对。谢琪同志,以及研究生姚威力、陈彬、王军等同志为书中大量例子程序的建立和调试等付出了辛勤的劳动,值此一并表示谢忱。

本书在正式出版前,因教学急需,曾先油印成讲义使用。此次出版时,又作了全面修订和若干补充,但限于水平和时间仓促,书中仍难免错误,恳请读者批评指正。

编译者 1988.2.
于南京大学计算机科学系

目 录

前 言

第一章 Turbo Prolog 使用入门	(1)
§ 1.1 安装 Turbo Prolog	(1)
1.1.1 在硬盘系统上安装 Turbo Prolog	(1)
1.1.2 在双软盘系统上安装 Turbo Prolog	(2)
§ 1.2 运行 Turbo Prolog	(3)
1.2.1 硬盘系统下的启动	(3)
1.2.2 软盘系统下的启动	(3)
§ 1.3 Turbo Prolog 用户接口	(5)
1.3.1 运行 Turbo Prolog 程序	(5)
1.3.2 保存程序文件	(6)
1.3.3 显示文件目录	(6)
1.3.4 装入和编辑程序文件	(6)
1.3.5 打印文件	(7)
小 结	(7)
练习一	(7)
第二章 Turbo Prolog 基本概念	(9)
§ 2.1 陈述式语言与过程式语言	(9)
2.1.1 谓词逻辑	(10)
2.1.2 Prolog 与谓词逻辑	(11)
§ 2.2 控制程序流	(12)
2.2.1 程序目标	(13)
2.2.2 求解问题	(13)
2.2.3 目标测试的输出	(14)
§ 2.3 连结词与其他符号	(15)
§ 2.4 目标与子目标	(15)
§ 2.5 Turbo Prolog 的内部合一程序	(16)
2.5.1 用事实与规则表示数据	(17)
2.5.2 内部合一中的变量与规则	(18)
2.5.3 回溯	(20)
小 结	(24)
练习二	(25)
第三章 Turbo Prolog 程序设计基础	(27)
§ 3.1 Turbo Prolog 程序的结构	(27)

3.1.1 域和谓词说明.....	(28)
3.1.2 Turbo Prolog 的项	(31)
§ 3.2 谓词和子句	(31)
3.2.1 内部目标和外部目标.....	(33)
3.2.2 多元谓词和子句.....	(34)
3.2.3 询问中使用规则.....	(39)
3.2.4 简单数据库.....	(41)
3.2.5 否定.....	(44)
3.2.6 复合对象及其使用.....	(46)
3.2.7 可选择域及其使用.....	(51)
§ 3.3 算术运算	(56)
小 结.....	(58)
练习三.....	(58)
 第四章 重复和递归程序设计.....	(60)
§ 4.1 重复性任务及其实现方法	(60)
§ 4.2 重复和回溯	(61)
§ 4.3 重复方法	(62)
4.3.1 BAF 方法	(62)
4.3.2 CAF 方法	(68)
4.3.3 UDR 方法	(73)
§ 4.4 递归方法	(76)
4.4.1 简单递归	(76)
4.4.2 GRR方法.....	(78)
小 结.....	(84)
练习四.....	(84)
 第五章 表程序设计.....	(86)
§ 5.1 表与 Turbo Prolog	(86)
5.1.1 表的属性	(86)
5.1.2 表的图示法	(87)
§ 5.2 表的建立和使用	(88)
§ 5.3 首-尾方法及其使用	(91)
§ 5.4 表处理技术	(93)
5.4.1 表的搜索	(93)
5.4.2 表的分解	(95)
5.4.3 表的拼接	(97)
5.4.4 表的排序	(99)
§ 5.5 表的数据采集	(102)
小 结.....	(104)
练习五.....	(105)

第六章 字符串程序设计	(107)
§ 6.1 ASCII 码表示	(107)
§ 6.2 字符串赋值	(109)
6.3 字符串的长度	(111)
6.4 字符串的拼接	(113)
6.5 建立子串	(115)
6.6 数据转换	(116)
6.7 用户定义的转换谓词	(117)
6.8 字符串前缀字符	(119)
6.9 字符串转换成字符表	(120)
6.10 Turbo Prolog 的名	(122)
6.11 由字符串形成词法符号	(123)
6.12 转变字符串成函子词法符号表	(125)
小 结	(128)
练习六	(128)
第七章 文件程序设计	(131)
§ 7.1 设备配置	(131)
7.1.1 物理和逻辑设备的分配	(131)
7.1.2 默认设备配置	(132)
7.1.3 扩充的设备配置	(135)
§ 7.2 Turbo Prolog 文件处理谓词	(135)
7.2.1 文件域说明	(136)
7.2.2 写数据文件	(136)
7.2.3 读数据文件	(137)
7.2.4 修改数据文件	(137)
7.2.5 扩充数据文件	(138)
§ 7.3 程序设计工具: 结构图和数据流图	(139)
§ 7.4 文件谓词的使用	(140)
7.4.1 创建新的磁盘文件	(140)
7.4.2 读出已存在的文件	(142)
7.4.3 由键盘输入创建一个磁盘文件	(144)
7.4.4 输出文件到屏幕和打印机	(146)
7.4.5 建立字符数据文件	(149)
7.4.6 从文件中读字符	(151)
§ 7.5 设计用户界面友善的文件处理程序	(153)
7.5.1 用户界面友善的写文件程序	(153)
7.5.2 用户界面友善的读文件程序	(158)
§ 7.6 添加数据到磁盘文件	(160)
§ 7.7 随机存取文件的读、写	(163)

7.7.1 写随机存取文件	(161)
7.7.2 读随机存取文件	(168)
小 结	(170)
练习七	(170)

第八章 窗口、图形和声音程序设计..... (172)

§ 8.1 窗口谓词及其使用	(172)
8.1.1 使用 makewindow 谓词设计窗口	(172)
8.1.2 其他窗口谓词及其使用	(175)
§ 8.2 使用窗口进行输入输出	(176)
§ 8.3 设计含有窗口的程序	(177)
8.3.1 在窗口内写正文	(177)
8.3.2 创建多窗口	(178)
8.3.3 创建重迭的窗口	(180)
8.3.4 利用窗口构造菜单	(182)
§ 8.4 设计产生图形的程序	(186)
8.4.1 分辨率和图形适配器模式	(186)
8.4.2 graphics 谓词	(187)
8.4.3 makewindow 谓词和图形模式	(188)
8.4.4 line 谓词和 dot 谓词	(190)
8.4.5 图形模式和参量值的总结	(190)
§ 8.5 用 line 和 dot 画图	(191)
8.5.1 画圆	(192)
8.5.2 绘制规则图形	(192)
§ 8.6 绘制统计图形	(195)
8.6.1 画线图	(195)
8.6.2 画简单条形图	(198)
8.6.3 画饼图	(201)
§ 8.7 海龟图形	(206)
8.7.1 海龟图形谓词	(206)
8.7.2 绘制图画	(208)
8.7.3 绘制移动的螺线	(210)
§ 8.8 产生声音和音乐	(211)
8.8.1 声音谓词	(211)
8.8.2 音乐标记	(212)
8.8.3 一个简单的带有音乐标记的CAI程序	(213)
小 结	(217)
练习八	(217)

第九章 建立动态数据库..... (220)

§ 9.1 数据库基础	(220)
9.1.1 数据库文件	(220)

9.1.2 关系数据库	(221)
9.1.3 Turbo Prolog 数据库	(222)
9.1.4 Turbo Prolog 数据库谓词	(223)
§ 9.2 建立基于内存的数据库	(226)
9.2.1 设计考虑	(226)
9.2.2 建立数据库	(227)
9.2.3 程序模块的实现	(229)
9.2.4 职业足球队数据库程序	(234)
§ 9.3 建立基于磁盘的数据库	(239)
9.3.1 设计考虑	(239)
9.3.2 建立数据库	(240)
9.3.3 大学足球队数据库程序	(246)
§ 9.4 数据库管理系统的改进	(251)
小 结	(252)
练习九	(252)
第十章 基于 Turbo Prolog 的专家系统建造	(254)
§ 10.1 专家系统的基本原理	(254)
10.1.1 专家系统的结构	(254)
10.1.2 知识表示	(256)
10.1.3 推理方法	(257)
10.1.4 用户接口	(257)
10.1.5 基于规则的专家系统	(258)
10.1.6 基于逻辑的专家系统	(259)
§ 10.2 基本的专家系统开发	(260)
10.2.1 一个分类专家系统的知识库	(261)
10.2.2 基于规则的专家系统的设计与实现	(263)
10.2.3 基于逻辑的专家系统的设计与实现	(269)
§ 10.3 基于逻辑的专家系统的扩展	(280)
10.3.1 设计考虑	(280)
10.3.2 建立医疗诊断专家系统	(281)
§ 10.4 勘探地下水专家系统的设计与实现	(288)
10.4.1 新构造分析方法简介	(288)
10.4.2 NCGW 系统结构概述	(289)
10.4.3 知识库结构与推理策略	(290)
10.4.4 知识获取子系统的功能及实现	(295)
10.4.5 人机接口与解释子系统的功能及实现	(298)
10.4.6 模块化程序设计技术	(302)
10.4.7 Turbo Prolog 编程经验点滴	(306)
小 结	(307)
练习十	(307)
第十一章 自然语言处理	(309)

§ 11.1	自然语言处理方法	(309)
§ 11.2	简单关键字识别程序	(310)
§ 11.3	表生成程序	(311)
§ 11.4	数据库关键字识别程序	(314)
§ 11.5	程序 Football 的 NLP 接口程序	(317)
§ 11.6	上下文无关文法分析	(325)
11.6.1	简单CFG分析模型	(325)
11.6.2	句子语法分析程序	(325)
11.6.3	上下文无关文法 (CFG) 句子分析程序	(331)
小 结		(337)
练习十一		(337)
第十二章	游戏和难题求解	(339)
§ 12.1	游戏和难题中的问题求解	(339)
§ 12.2	猜数游戏	(340)
12.2.1	设计考虑	(340)
12.2.2	一个游戏方案	(340)
12.2.3	猜数游戏程序	(341)
§ 12.3	23 根火柴游戏	(345)
12.3.1	设计考虑	(345)
12.3.2	简单的 23 根火柴游戏程序	(346)
12.3.3	一个聪明的 23 根火柴游戏程序	(350)
§ 12.4	猴子和香蕉问题	(355)
12.4.1	程序设计考虑	(355)
12.4.2	猴子和香蕉问题程序	(356)
§ 12.5	史密斯家庭聚会难题	(364)
12.5.1	问题的陈述	(364)
12.5.2	逻辑程序设计	(364)
12.5.3	家庭聚会程序	(366)
12.5.4	用户接口	(378)
小 结		(381)
练习十二		(382)
附录 A	使用 Turbo Prolog 编辑程序	(383)
A.1	基本编辑命令	(383)
A.2	其他编辑命令	(384)
附录 B	Turbo Prolog 的其他特征	(387)
B.1	Turbo Prolog 系统选择项	(387)
B.2	Turbo Prolog 程序的编译和连接	(389)
B.3	存储管理	(390)

B.4	编译程序控制	(391)
B.5	编译指令	(391)
B.6	使用跟踪设施	(392)
附录 C	名词术语汇编	(394)
附录 D	参考文献和进一步的读物	(397)

第一章 Turbo Prolog 使用入门

Turbo Prolog 是当今一种令人鼓舞的人工智能语言。几年之前, 还只有少数具有昂贵主机的计算机用户, 才能利用 AI 程序设计语言。由于 Turbo Prolog 高速度、高性能和低价格以及对 IBM PC 及兼容机的有效性, 使得众多的组织和个人能够以可承受的代价从 Turbo Prolog 应用程序的使用中获得裨益。

Turbo Prolog 在 IBM PC 及其兼容机上运行, 要有效地使用 Turbo Prolog, 需要对使用的计算机系统和使用的计算机磁盘操作系统(PC DOS 或 MS DOS)有所了解, 并熟悉经常使用的 DOS 命令。本书假定读者具有这方面的经验, 否则, 在学习本章之前最好阅读参考文献[6]的有关章节。

本章首先简略地介绍在硬盘和双软盘系统上安装和启动 Turbo Prolog 的方法, 然后描述 Turbo Prolog 的用户接口, 为读者提供了建立 Turbo Prolog 系统的基本步骤和最经常使用的系统命令的用法。

§ 1.1 安 装 Turbo Prolog

Turbo Prolog 编译程序包配有两张软磁盘, 一张称为程序盘, 另一张称为库和例子盘 (Library & Sample Program)。

Turbo Prolog 既可在硬盘系统上运行, 又可在双软盘系统上运行, 无论在哪个系统上运行, 它都至少需要 384 KB 的 RAM (随机存储器)。另外, 需要 PC DOS 2.0 版本 (或其更高版本) 的操作系统的支持。

在安装 Turbo Prolog 之前, 建议使用 DISKCOPY 命令来建立配给磁盘的工作副本。

1.1.1 在硬盘系统上安装 Turbo Prolog

为了在硬盘系统上安装 Turbo Prolog, 需要建立一个目录并把文件从配给软盘拷贝到硬盘上, 在硬盘上建立的目录应同配给磁盘相同。

假定硬盘驱动器为 C:, 因此系统提示符是 C>。安装步骤如下:

(1) 输入以下信息建立 Turbo Prolog 的文件目录。

```
MD\TBPROLOG
MD\TBPROLOG\EXAMPLES
MD\TBPROLOG\ANSWERS
MD\TBPROLOG\PROGRAMS
```

(2) 把文件从配给软盘拷贝到硬盘上的 Turbo Prolog 目录结构中。

① 把程序盘或其工作副本插入驱动器 A:。

② 输入下列信息, 把缺省目录改成目录\TBPROLOG。

CD\TBPROLOG

- ③ 使用下述命令, 把文件从驱动器A: 拷贝到目录 C:\TBPROLOG。

COPY A: *.* C: /V

- ④ 从驱动器 A: 中取出程序盘并插入库和例子盘。

- ⑤ 把 PROLOG.LIB 和 INIT.OBJ 两个系统文件拷贝到 TBPROLOG 目录中。这个命令是

COPY A: PROLOG.LIB C: /V

COPY A: INIT.OBJ C: /V

- ⑥ 把所有的文件从驱动器 A: 中的库和例子盘上的子目录拷贝到硬盘上的子目录。这些命令是

COPY A: \EXAMPLES*.* C: \TBPROLOG\EXAMPLES/V

COPY A: \ANSWERS*.* C: \TBPROLOG\ANSWERS/V

COPY A: \PROGRAMS*.* C: \TBPROLOG\PROGRAMS/V

现在已经把 Turbo Prolog 的所有文件拷贝到了硬盘系统且具有同配给磁盘一样的目录结构。现可以使用操作系统命令 CD 和 DIR 查看所有目录。

1.1.2 在双软盘系统上安装 Turbo Prolog

可以按照下述步骤在双软盘系统上安装 Turbo Prolog, 这里假定软盘驱动器盘符是 A: 和 B:。首先准备三张空磁盘。

- (1) 使用 FORMAT 命令格式化空磁盘。

- ① 把 DOS 盘放在驱动器 A:, 一张空盘放在驱动器 B:, 然后键入命令

FORMAT B: /S

/S 选择项使操作系统文件拷贝到那个空磁盘上, 以便使用它来启动计算机。当 FORMAT 程序问是否想要格式化另一张空磁盘时, 键入 N (如果键入 Y 表示继续格式化另外两张盘, 系统文件也被拷贝到这些盘上)。把那张空盘从驱动器 B: 中取出, 并标识为 Turbo Prolog 启动盘。

- ② 把另一张空盘放入驱动器 B: 并键入命令

FORMAT B:

(未带选择项/S)。这次, 当系统问是否要格式化另一张磁盘时回答 Y。

- ③ 把第三张盘插入驱动器 B:, 并按屏幕上出现的指令去做。当第三张盘进行格式化时, 可以将第二张盘 (已格式化的未带/S 选择项的那张盘) 标识为 Turbo Prolog 工作盘。第三张盘格式化好后, 回答 N, 把盘从驱动器 B: 中取出, 并且标识为 Turbo Prolog 库盘。

- (2) 把程序盘或其工作副本放入驱动器 A:, 标有 Turbo Prolog 启动盘的那张盘插入驱动器 B:, 然后键入命令

COPY A: PROLOG * B: /V

把 Prolog 系统文件拷贝到启动盘。

- (3) 现在把标有 Turbo Prolog 工作盘的磁盘放进驱动器 B: (把程序盘留在驱动器 A:), 并键入命令

COPY A: GEOBASE.* B: /V