

高等学校教材

# 人工智能与知识工程

北方交通大学 田盛丰 黄厚宽 编著  
清华大学 石纯一 主审

中国铁道出版社

1999年·北京

# (京)新登字 063 号

## 内 容 简 介

本书系统地介绍了人工智能学科的基本原理和应用技术,特别是各种类型的专家系统的构造。还涉及人工智能研究的一些前沿课题,如智能 Agent 系统、计算智能和从数据库发现知识等。

本书共分十一章,前五章介绍了人工智能的基本原理与方法,包括程序设计语言、知识表示、搜索策略和演绎推理,第六、七章介绍了建造专家系统的实用技术,第八、九、十章介绍了一些更深入的课题,包括机器学习、智能 Agent 和计算智能,第十一章详细地介绍了一个新型的专家系统工具。

本书注重实用性与先进性,并附有习题,可作为大学计算机及有关专业高年级学生和研究生的教材,也可供从事智能系统开发的科技人员参考。

## 图书在版编目(CIP)数据

人工智能与知识工程/田盛丰著. —北京:中国铁道出版社,1999  
高等学校教材  
ISBN 7-113-03406-3

I. 人… II. 田… III. 人工智能-高等学校-教材 IV. TP18

中国版本图书馆 CIP 数据核字(1999)第 45770 号

书 名:人工智能与知识工程

作 者:田盛丰 黄厚宽

出版发行:中国铁道出版社(100054,北京市宣武区右安门西街8号)

责任编辑:郭 宇

封面设计:马 利

印 刷:中国铁道出版社印刷厂

开 本:787×1092 1/16 印张:23.25 字数:579 千

版 本:1999 年 8 月第 1 版 1999 年 8 月第 1 次印刷

印 数:1~3000 册

书 号:ISBN7-113-03406-3/TP·378

定 价:29.40 元

版权所有 盗印必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社发行部调换。

# 前 言

人工智能是近 40 多年来迅速发展起来的一门新兴学科,人工智能技术已广泛应用于工业、农业、气象、地质、建筑、交通运输、数学、物理、化学、医学、军事等各个领域。在面向知识经济的今天,研究获取、表示和使用知识的人工智能学科就越来越受到人们的重视,新的思想、方法不断涌现。本书以实用性和先进性为目标,从人工智能的基本原理出发,详尽讲述建造专家系统的实用技术,并广泛介绍新的研究成果,如计算智能、机器学习和智能 Agent 系统等。

本书是在作者长期讲授本科生和研究生的人工智能课的教学实践的基础上,以及长期从事人工智能研究工作的基础上编写的。书中全面体现了“人工智能与知识工程”新的教学体系的内容,将“人工智能”、“专家系统”和“知识工程”三门人工智能学科的课程构成一个完整的体系。涉及基本原理的各章都附有习题,可作为大学计算机及有关专业高年级学生和研究生的教材,也可供从事智能系统开发的科技人员参考。

本书共分十一章。第一章为绪论。第二章介绍常用的人工智能程序设计语言。第三章至第五章介绍人工智能的基本原理和方法,包括知识表示、搜索方法和推理方法。第六章和第七章介绍建造专家系统的实用技术,包括推理技术和系统结构。第八章介绍知识获取和机器学习。第九章介绍计算智能,包括人工神经网络和演化算法。第十章介绍智能 Agent 和多 Agent 系统。第十一章介绍作者研制的一个专家系统工具,以供读者参考。

本书第一至九章及第十一章由田盛丰执笔,第十章由黄厚宽执笔并统稿全书。本书由清华大学计算机科学与技术系石纯一教授担任主审。

对于本书中的错误和不足之处,恳请各位专家和广大读者指正。

作 者

1999 年 1 月于北方交通大学

# 目 录

第一章 绪 论	1
1.1 人工智能的发展概况	1
1.1.1 什么是人工智能	1
1.1.2 人工智能的研究途径	1
1.1.3 人工智能学科的发展	2
1.2 人工智能的应用	3
1.2.1 人工智能的应用领域	3
1.2.2 专家系统	3
第二章 人工智能程序设计语言	6
2.1 LISP 语言	6
2.1.1 概 述	6
2.1.2 LISP 的基本功能	8
2.1.3 递归与迭代	15
2.1.4 输入输出功能	18
2.1.5 LISP 的其他功能	20
2.2 PROLOG 语言	22
2.2.1 概 述	22
2.2.2 重复与递归	28
2.2.3 表处理方法	33
2.2.4 字符串处理方法	35
2.2.5 输入输出功能	36
习 题	38
第三章 知识表示	41
3.1 概 述	41
3.1.1 知识与知识表示	41
3.1.2 知识表示的方法	41
3.2 逻辑表示法	43
3.2.1 一阶谓词逻辑	43
3.2.2 谓词逻辑用于知识表示	46
3.3 规则表示法	47
3.3.1 产生式规则与产生式系统	47

3.3.2	Markov 算法与 Rete 算法	49
3.3.3	控制策略的类型	51
3.4	语义网络表示法	52
3.4.1	语义网络的基本概念	52
3.4.2	语义网络的应用	54
3.5	框架表示法	56
3.5.1	框架的基本概念	56
3.5.2	框架表示的应用	57
3.6	概念从属与剧本表示法	58
3.6.1	概念从属	58
3.6.2	剧本	61
	习 题	62
第四章 基本的问题求解方法		64
4.1	状态空间搜索	64
4.1.1	概 述	64
4.1.2	回溯策略	67
4.1.3	图搜索策略	70
4.1.4	任一路径的图搜索	72
4.1.5	最佳路径的图搜索	73
4.1.6	与或图的搜索	79
4.2	博弈树搜索	82
4.2.1	概 述	82
4.2.2	极小极大过程	82
4.2.3	$\alpha$ - $\beta$ 过程	84
4.3	约束满足搜索	87
4.3.1	概 述	87
4.3.2	生长法	88
4.3.3	修改法	94
4.4	通用问题求解	98
4.4.1	手段目的分析	98
4.4.2	生成与测试(Generate-and-test)	100
	习 题	101
第五章 基本的推理方法		104
5.1	归结反演系统	104
5.1.1	谓词演算基础	104
5.1.2	归结反演(refutation)	108
5.1.3	归结反演的控制策略	109
5.1.4	从归结反演中提取解答	111

5.2	基于规则的演绎系统 .....	114
5.2.1	正向演绎系统 .....	114
5.2.2	逆向演绎系统 .....	119
5.3	规划生成系统 .....	121
5.3.1	机器人问题求解 .....	121
5.3.2	正向系统 .....	122
5.3.3	规划中表示 .....	123
5.3.4	逆向系统 .....	125
	习 题 .....	128
第六章 实用推理技术 .....		131
6.1	推理的类型 .....	131
6.1.1	从逻辑基础上的分类 .....	131
6.1.2	从推理方法上的分类 .....	133
6.2	非单调推理 .....	134
6.2.1	概 述 .....	134
6.2.2	非单调逻辑 .....	135
6.2.3	非单调系统 .....	137
6.3	不精确推理 .....	139
6.3.1	概 述 .....	139
6.3.2	概率方法 .....	144
6.3.3	可信度方法 .....	148
6.3.4	主观 Bayes 方法 .....	152
6.3.5	证据理论 .....	157
6.3.6	可能性理论 .....	164
6.4	基于模型的推理 .....	171
6.4.1	基本原理 .....	171
6.4.2	基于规则与模型的系统 .....	172
6.4.3	基于模型的故障诊断系统 .....	174
6.5	基于事例的推理 .....	181
6.5.1	基本概念 .....	181
6.5.2	基本方法 .....	182
6.5.3	与基于规则的系统比较 .....	183
6.5.4	实例系统 .....	183
	习 题 .....	186
第七章 专家系统 .....		187
7.1	基本结构 .....	187
7.2	元知识结构 .....	188
7.2.1	什么是元知识 .....	188

7.2.2	元知识的作用 .....	188
7.2.3	元知识在专家系统中的应用 .....	190
7.3	黑板系统结构 .....	193
7.3.1	黑板模型 .....	193
7.3.2	黑板结构 .....	194
7.3.3	知识源 .....	195
7.3.4	控制策略 .....	197
7.3.5	黑板模型的优越性 .....	197
7.4	黑板控制结构 .....	198
7.4.1	基本概念 .....	198
7.4.2	知识源表示 .....	198
7.4.3	控制黑板的组织 .....	199
7.4.4	调度机制 .....	201
7.4.5	黑板控制结构的优点与不足 .....	203
7.5	实 例 .....	203
7.5.1	MYCIN 系统 .....	203
7.5.2	AM 系统 .....	211
	习 题 .....	214
第八章 知识获取与机器学习 .....		215
8.1	概 述 .....	215
8.1.1	知识获取的基本过程 .....	215
8.1.2	知识获取的主要手段 .....	217
8.1.3	机器学习 .....	218
8.1.4	知识获取工具 .....	220
8.2	通过例子学习 .....	226
8.2.1	概 述 .....	226
8.2.2	学习单个概念 .....	229
8.2.3	学习多个概念 .....	237
8.2.4	学习执行多步任务 .....	245
8.3	通过类比学习 .....	250
8.3.1	概 述 .....	250
8.3.2	类比学习与推理系统 .....	252
8.3.3	转换类比与派生类比系统 .....	254
8.4	基于解释的学习 .....	259
8.4.1	概 述 .....	259
8.4.2	基于解释的抽象 .....	260
8.5	通过观察学习 .....	262
8.5.1	合取概念聚类系统 .....	262
8.5.2	结构对象的概念聚类 .....	267

8.6	从数据库学习 .....	270
8.6.1	数据库中的知识发现 .....	270
8.6.2	数据挖掘方法 .....	271
	习 题 .....	277
<b>第九章</b>	<b>计算智能 .....</b>	<b>278</b>
9.1	演化计算 .....	278
9.1.1	遗传算法概述 .....	278
9.1.2	遗传算法的理论基础 .....	280
9.1.3	演化策略与演化规划 .....	282
9.2	连接计算 .....	283
9.2.1	概 述 .....	283
9.2.2	感 知 机 .....	284
9.2.3	多层前向网络 .....	287
9.2.4	Hopfield 网络 .....	289
9.3	混合系统 .....	291
9.3.1	混合系统结构 .....	291
9.3.2	用神经网络表示符号知识 .....	292
<b>第十章</b>	<b>智能 Agent .....</b>	<b>294</b>
10.1	关于智能 Agent .....	294
10.1.1	基本概念 .....	294
10.1.2	Agent 理论 .....	295
10.1.3	Agent 系统结构 .....	296
10.1.4	Agent 程序设计 .....	299
10.1.5	Agent 的应用 .....	301
10.2	多 Agent 系统 .....	303
10.2.1	概 述 .....	303
10.2.2	多 Agent 系统的结构 .....	304
10.2.3	Agent 通信语言 .....	306
10.2.4	多 Agent 系统的协商机制 .....	313
<b>第十一章</b>	<b>分布式专家系统工具 DEST .....</b>	<b>320</b>
11.1	分布式系统结构 .....	320
11.2	面向对象的知识表示 .....	322
11.2.1	对象的表示 .....	322
11.2.2	规则的表示 .....	327
11.2.3	方法的表示 .....	330
11.3	问题求解机制 .....	335
11.3.1	概 述 .....	335

11.3.2	模糊规则推理 .....	336
11.3.3	模糊决策树推理 .....	341
11.3.4	神经网络的模拟 .....	345
11.3.5	实    例 .....	347
附录	DEST 系统函数 .....	351
参考文献	.....	355

# 第一章 绪 论

## 1.1 人工智能的发展概况

人工智能(Artificial Intelligence)简称 AI,它是计算机科学的一个重要的研究领域。近 40 年以来获得了迅速的发展,在很多领域都获得了广泛的应用。

### 1.1.1 什么是人工智能

斯坦福大学人工智能研究中心的 Nilsson 教授认为:“人工智能是关于知识的科学——怎样表示知识以及怎样获得知识并使用知识的科学。”MIT 的 Winston 教授指出:“人工智能就是研究如何使计算机去做过去只有人才能做的智能的工作。”这些定义反映了人工智能学科的基本思想和基本范围。事实上,一般认为用计算机模拟人的智能行为就属于人工智能的范畴。

### 1.1.2 人工智能的研究途径

从不同的途径认识和模拟人类的智能行为,产生了不同的人工智能的理论和技術。一般认为,模拟人类智能的途径有以下几种:

#### 1. 心理学途径

心理学的途径就是考察人类在解决各种问题时采取的方法,总结人们思维活动的规律,由此产生了人工智能的符号机制(symbolicism)。符号机制认为认识的基本元素是符号,人的认识过程就是一种符号处理过程。人类的思维过程可以用某种符号来描述,因此人工智能的核心就是“表示”问题。40 多年以来,人工智能的巨大进展主要就是在这种思想的指导下取得的,这也是本书介绍的重点。

#### 2. 生理学途径

这种途径直接模拟人和动物的大脑,由此产生了人工智能的连接机制(connectionism),即人工神经网络方法。连接机制认为认识的基本元素就是神经元本身,认识过程就是大量神经元的整体活动,本质上是并行分布的模式。因此,连接机制就是建立人工神经网络,模拟人类的智能行为。自从 80 年代以来,连接机制的研究获得了很大的发展。

#### 3. 生物演化途径

生物演化途径的研究者认为,人类的智能就是从生物的演化中得到的,从对生物演化的模拟产生了演化算法(evolutional programming)。从 70 年代开始,演化算法的研究也取得了很大的发展。

目前,将演化计算、连接计算、模糊计算(符号机制中的模糊逻辑方法)和最近发展起来的混沌计算统称为计算智能(computational intelligence)。计算智能是人工智能研究的一个新的方向,使人工智能从传统的符号计算向符号计算和数值计算的复合集成的方向发展。

#### 4. 社会学途径

社会学途径注重研究人在社会中的行为,将人模拟成由多种智能品质构成的有机的整

体——Agent,综合考察 Agent 本身及其在 Agent 环境中的行为,这就是智能 Agent 理论。近年来智能 Agent 理论也成为人工智能界最引人注目的研究课题之一。

本书将以符号机制为主,全面介绍人工智能的技术与方法。

### 1.1.3 人工智能学科的发展

“人工智能”这一术语,是在 1956 年由 McCarthy 和 Minsky 等人发起的关于用机器模拟智能的学术讨论会上提出的,这就标志着人工智能这一学科的诞生。1957 年 Rosenblatt 研制了感知机,这是一种将神经元用于识别的系统,它的学习功能引起了广泛的兴趣,推动了连接机制的研究。但是很快发现了感知机的局限性,不能解决复杂的识别问题,从而连接机制的研究进入低潮。从 60 年代到 70 年代,符号机制的研究获得了很大的发展。特别是 DENDRAL、MYCIN 和 PROSPECTOR 等有名的专家系统的研制成功,使人工智能开始走向实际的应用。70 年代 Holland 提出了遗传算法,标志着演化计算研究的开始。在 80 年代,连接机制的研究也有了突破性的进展,特别是 Hopfield 提出的网络模型和 Rumelhart 等人提出的多层网络学习算法,又把连接机制的研究推向高潮。在这一时期,符号机制和演化计算的研究也有了很大的进展。在 90 年代,智能 Agent 的研究开始引起人工智能界的广泛注意,并迅速成为一个热门的研究课题。目前,人工智能的研究在上述各个分支都取得了很大进展。

前面说到,人工智能是关于知识的科学,通常将实用的人工智能称为知识工程。知识工程的概念产生于 70 年代中期,1977 年美国斯坦福大学的 Feigenbaum 教授在第五届国际人工智能会议上提出了“知识工程”这一概念。他以“人工智能的艺术:知识工程的课题及实例研究”为题,对知识工程作了全面的论述。他认为:“知识工程是人工智能的一种技艺,它运用人工智能的原理和方法,对那些需要专家知识才能解决的应用难题提供求解的手段。恰当运用专家知识的获取、表达和推理过程的构成与解释,是设计基于知识的系统的重要技术问题。”知识工程这一概念的提出,标志着人工智能的研究进入了实际应用的阶段。

随着知识在人类社会的发展中占有越来越重要的位置,人工智能的研究和应用也就越来越受到重视。1962 年美国普林斯顿大学的马克卢普教授提出了“知识产业”的概念。所谓知识产业,就是生产和提供知识的产业。它是以知识为主体形成的,知识产业不同于知识密集型产业。知识密集型产业主要生产物质产品,而知识产业是以信息为中心的关联企业,主要产品是信息,其主要形式包括研究成果、软件、咨询报告、产品设计与工程设计等。

到了 90 年代,世界经济开始进入知识经济的时代。1996 年西方国家的经济合作发展组织(OECD)发出了一份年度报告,题目就是“以知识为基础的经济”。根据其定义,知识经济是以知识为基础的经济,这种经济直接依赖于知识和信息的生产、分配和使用。目前,西方知识经济是以信息产业为代表的迅速发展的高科技产业。世界首富、微软公司总裁比尔·盖茨被公认为知识经济时代的代表人物。因此,知识经济也称为信息经济。但严格来说,知识经济包含更多的内容。信息科学、遗传工程、新能源、新材料、空间技术、海洋科学、环境科学及现代管理等知识成为发达国家经济投入的重要资本,形成了一些以知识、智力等无形资产投入为主的企业。知识经济的出现,标志着人类社会正在步入一个以知识资源为主要依托的经济时代。因此,研究知识的表示、获取与应用的科学——人工智能,就变得越来越重要了。

## 1.2 人工智能的应用

### 1.2.1 人工智能的应用领域

目前,人工智能的主要应用领域包括:

- (1)专家系统。模拟人类专家的问题求解过程,解决那些只有专家才能解决的复杂问题。
- (2)知识库系统。对人类的知识进行存储、加工、管理,并根据需要对知识进行处理和应用。
- (3)决策支持系统。利用模型和知识,通过模拟和推理等手段,为人类的活动进行辅助决策。
- (4)自然语言理解。理解人类的自然语言,以实现人和计算机之间自然语言的直接通讯,从而推动计算机更广泛的应用。
- (5)智能机器人。具有感觉、识别和决策功能的机器人。
- (6)模式识别。模拟人类的听觉、视觉等感觉功能,对声音、图像、景物、文字等进行识别。
- (7)自动程序设计。由计算机完成程序的验证和综合,实现程序设计自动化。

当前,人工智能最主要的应用就是建造专家系统。

### 1.2.2 专家系统

专家系统(Expert System)简称 ES,是目前人工智能的应用方面最成熟的一个领域。

#### 1. 专家系统的产生

早在 60 年代初,人们就已开始研究用人工智能技术解决实际问题。通用问题求解器(GPS)就是一个典型的例子。但是人们很快发现,客观世界是相当复杂的,企图用一种普遍适用的模式去解决所有的问题是不可能的,因此开始转向比较狭窄的专门领域。1965 年,Feigenbaum 研制了第一个专家系统 DENDRAL,从而开创了专家系统的研究。目前,专家系统的应用领域已扩展到数学、物理、化学、医学、地质、气象、农业、法律、教育、交通运输、机械、艺术以及计算机科学本身,甚至渗透到政治、经济、军事等重大决策部门。

#### 2. 专家系统的特点

什么是专家系统? Feigenbaum 认为:“专家系统是一种智能的计算机程序,它运用知识和推理步骤来解决只有专家才能解决的复杂问题。”也就是说,专家系统提供了一种新型的程序设计方法,可以解决传统的程序设计方法难以解决的问题。专家系统的特点大致有以下三点:

(1)专家系统所要解决的是复杂而专门的问题。对这些问题,人们还没有精确的描述和严格的分析,因此还没有确定的算法来解决。解决这些问题需要专家的知识,包括理论知识和实际经验。

(2)专家系统突出了知识的价值。通常要推广和应用专家的知识,要通过培训的方法,这需要若干年的时间。专家系统则大大减少了知识传授和应用的代价,使专家的知识迅速转变成社会的财富。

(3)专家系统采用的是人工智能的原理和技术,包括符号智能和计算智能的各种方法。

总之,专家系统注重于知识本身而不是确定的算法。

#### 3. 专家系统的分类

专家系统从完成的任务分类,可包括诊断、预测、解释、调试、修理、规划、设计、监督、控制等多种类型。这些功能又可分成两大类:分析型和综合型。分析型专家系统所要解决的问题有

明确的、有限个数的解,系统的任务在于根据实际的情况选择其中的一种或几种解。这种专家系统的功能可包括分类、诊断、预测、修理以及部分设计、规划等方面。综合型专家系统的任务是根据实际的需要构造问题的解,包括设计、规划等类问题。另外,也可根据知识的特征和推理的类型对专家系统进行分类。

#### 4. 几个有名的专家系统

DENDRAL 系统是 1965 年研制的第一个专家系统。它用于根据有机化合物的分子式和质谱图判断分子结构。当给出一个有机化合物的分子式时,其可能的分子结构往往有几千种。对于给定的有机化合物的分子式和质谱图,系统根据化学家的知识和质谱图的知识,删掉不可能的结构,从几千种可能的分子结构中挑选出一个正确的分子结构。这个系统已经达到了化学博士的水平。

MYCIN 系统是一个医疗诊断系统,用于在不知道原始病原体的情况下,判断如何用抗菌素来处理血液细菌感染病患者。系统输入的原始数据是患者的症状、一般情况、病史和化验结果。系统根据专家的知识 and 输入数据判断出是什么病菌引起的感染,再提出抗菌素的配方。MYCIN 系统于 1974 年基本完成,是第一个功能较全面的专家系统。它不仅有很高的性能,而且具有解释功能和知识获取功能,它可以用简单的英语和用户对话,回答用户提出的问题,还可以学习专家的知识。

PROSPECTOR 系统是一个地质勘探专家系统,用于寻找矿藏。一开始它装入了三种矿藏的知识,到 1981 年已拥有 15 种矿藏的知识。该系统的特点是很好地协调了多个专家的多种矿藏的知识模型。它已存入了 20 多名第一流的经济地质专家的知识。该系统曾发现了美国华盛顿州的一处钼矿,据说其开采价值超过了一亿美元。

AM 系统是一个机器学习系统,用于模拟人类的归纳推理、抽象概念。它可以由一组有限的集合论方面的简单概念抽象出自然数、因子、质数、乘法等概念,并发现了一些定理和猜想,如质数分解唯一性定理,哥德巴赫猜想等等。此外,还归纳出了一些人类还不曾发现的数学概念。

XCON 系统是设计计算机总体配置的系统,也称为 R1 系统。它根据用户的需要,配置 DEC VAX 计算机系统,它可以发现用户要求中的错误之处,配置计算机系统的各个部件,规划各部件在地板上的位置等。这个系统每年可为数字设备公司(DEC)节省 2000 万美元的开支。

美国政府资助的一个调度与规划专家系统曾用于海湾战争。早在 90 年代初期,美国政府就意识到使用专家系统来辅助军需后勤规划的重要性,并资助了一个研究规划专家系统工具的计划。1991 年初,在沙漠风暴之战中,美国面临的一个严重问题就是如何尽快从美国本土和欧洲运送大量军队和物资装备到沙特阿拉伯,这包括 50 万军队和超过 6800 吨重的设备。短期内完成这样繁重的任务是非常困难的。受政府资助的小组在 6 个星期内迅速开发了一个规划专家系统,大量 SUN 工作站运到沙特阿拉伯,制定了运输规划,在很短的时间内完成了军事部署。

#### 5. 专家系统的建造

专家系统主要由知识库和推理机组成,知识库用于存放专家知识,推理机根据知识库中的知识进行推理,做出决策。因此,建造专家系统的工作量主要就集中在这两个方面。

建立知识库的过程包括获取领域知识,将其按专家系统要求的形式构成知识库,并在运行中不断调整和充实知识库,使之达到实用的程度。这一过程称为知识获取,是研制专家系统最

为困难的阶段,通常称为建造专家系统的瓶颈问题。为解决这一问题,已建立了一些知识获取工具和机器学习算法,但还未达到普遍应用的程度。这个问题是人工智能研究的重要课题。

建造推理机的过程就是程序设计的过程。为了加速这一过程,已经出现了象 LISP 和 PROLOG 那样的人工智能程序设计语言,用这些语言设计专家系统比用类似 C、PASCAL 那样的通用程序设计语言更为简洁、方便。更简便的途径是利用专家系统构造工具,像 EMYCIN、EXPERT 等骨架系统。利用骨架系统可以完全摆脱繁琐的程序设计,把注意力完全集中于知识库的建造。但是骨架系统通常具有通用性差的弊病,难以应用到所有的专家系统。一个折衷的方法是采用通用的知识表示语言,如 ROSIE、OPS、CLIPS 等。它们比骨架系统的限制少,适应面更广,但应用过程比骨架系统复杂。专家系统开发工具发展的最高阶段是专家系统开发环境,如我国研制的天马系统,是以一种或多种方法为核心,加上配套的各种辅助工具和界面环境的集成系统。综上所述,对于选择软件工具问题,建议采取如下方法:

(1)寻找适用的骨架系统或专家系统开发环境,如果有适用的系统,可以快速地实现一个高质量的专家系统。

(2)选用通用的知识表示语言,或者 LISP 与 PROLOG 类型的人工智能程序设计语言,可以较迅速地建立自己设计的专家系统。

(3)选用通用的程序设计语言如 C 语言,虽然工作量最大,但系统性能好,功能强,并易于修改。

本书从人工智能原理开始,为以各种途径建造专家系统的读者提供充实的材料。

## 第二章 人工智能程序设计语言

在人工智能领域,最具特色的程序设计语言大体包括三种:以 LISP 为代表的函数型语言,以 PROLOG 为代表的逻辑型语言,和以 Smalltalk 为代表的面向对象的语言。目前,采用 LISP 语言和 PROLOG 语言进行人工智能程序设计的比较多,因此本章对这两种语言进行具体的介绍。采用面向对象的程序设计方法是当前的一个趋势,因此本书也将对面向对象的思想进行介绍。

### 2.1 LISP 语言

#### 2.1.1 概 述

LISP 语言是 1959 年由 McCarthy 领导的 MIT 的人工智能小组创立的,在人工智能的程序设计中获得了广泛的应用。

LISP 语言有多种版本,包括 Mac Lisp、Zeta Lisp 及 Inter Lisp 等,1983 年在这些版本的基础上出现了一种新型的 LISP 语言 Common Lisp。Gold Hill Computer 公司在 IBM-PC 及其兼容机上实现了 Common Lisp,它和 Common Lisp 的核心部分兼容,称为 Golden Common Lisp,简称 GCLISP。本节就是介绍 GCLISP。

##### 1. LISP 语言的特点

LISP 语言是一种符号处理语言,也可以说是一种表处理语言。LISP 就是 List Processing language 的缩写。为什么 LISP 语言特别适合于人工智能呢?因为人工智能就是设法用计算机来模拟人的思维过程,而人的思维过程往往可以用语言来描述,而语言可以用符号来表示,因此 LISP 这种符号处理语言就特别适合于人工智能。具体地讲,LISP 语言有如下特点:

(1)LISP 语言是函数型语言,它的一切功能都由函数实现。因此,执行 LISP 程序主要就是执行一个函数,这个函数再调用其它函数。用 LISP 语言进行程序设计就是定义函数。

(2)LISP 语言的函数和数据的形式是一样的,都是 S-表达式一种形式。这样会给程序设计带来很大的方便。

(3)LISP 语言中程序的运行就是求值。LISP 的函数除完成一定的功能外,每个函数都有一个值,因此执行一个函数就可以理解成对函数求值。函数所完成的功能可以看成是它的副作用,在对函数求值的过程中实现函数的功能。

(4)LISP 语言的一个主要控制结构就是递归。递归的使用,使程序设计简单易懂。

##### 2. LISP 的基本数据类型

LISP 语言的程序和数据采用同一个数据类型 S-表达式(Symbolic expression),在 Common Lisp 中也称为对象(object)。S-表达式由原子和表两种类型组成。

(1)原子(atom) 原子可分为文字原子、数原子和串原子。

文字原子也称为符号(symbol),它是由字母开头的字母数字串,可用作变量名、函数名、

特性名等,如 a,ab2 等。

数原子也称为数(number)。数原子又分为整数(integer)和浮点数(float)。字符(character)是整数的子类,可表示为#\字符名。如字母 A 表示为#\A,在机内存储它的 ASCII 码 65。

串原子也称为串(string),是由双引号"开头和结尾的字符串。

(2)表(list) 表由括号及其中的若干元素所组成,元素可为任何数据类型。如(a b c)及((a)(bc) ((( )))都是表。

表由空表和非空表(cons)组成,空表就是(),也可表示为 nil,它既是表,也是文字原子。

LISP 的基本数据类型及其相互关系如图 2.1 所示。另外还有一些其它的数据类型如数组(Array)、函数(function)、流(stream)等,遇到时再加以说明。



图 2.1 LISP 的基本数据类型

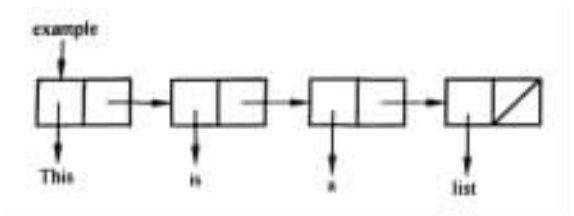


图 2.2 表(This is a list)的链表形式

(3)表在机内的存储 表在机内以链表的形式存储,链表是指一组内存单元,每个内存单元分成两部分,每个部分有一个地址,称为指针。右边的指针把各个单元串在一起,左边的指针指向表内各元素。

例如,若文字原子 example 的值是表(This is a list),表中共有 4 个元素,故链表由 4 个单元构成。每个单元的右边指针指向下一个单元,最后一个单元的右边指针为空(nil),每个单元的左边指针指向各元素如图 2.2 所示。

若文字原子 L 的值为表(4 (3 7) a (5) b c),表中有 6 个元素组成,因此链表应包含 6 个单元。但第 2、4 个元素又是表,它们又应分别由 2 个和 1 个单元组成,因此链表应如图 2.3 所示。

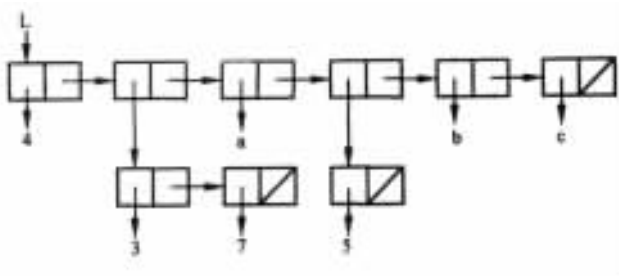


图 2.3 表(4 (3 7) a (5) b c)的链表结构

### 3. 读入-求值-显示循环

GCLISP 是一个解释执行的系统。进入系统后即进入了收听程序

(listener),完成读入-求值-显示循环。系统的提示符为\*。循环的操作为:由用户打入一个表达式,系统将其读入,对其求值,并将值显示在屏幕上。例如在提示符\*后打入求和函数(+ 1 2 3),则显示

\* (+ 1 2 3)

6

\*

系统显示了和 6 后,再显示提示符\*,等待下一个读入-求值-显示循环。

退出系统时打入(exit)

#### 4. 函数表示中的记号约定

为函数表示的简洁起见,特作如下约定:

&.rest:表示其后的自变量可以有 0 个或多个

&.key:表示其后为可选的关键词

&.optional:表示其后为可选的自变量

form:表示可求值表达式

form-result:表示 form 的值

last-form-result:表示最后一个 form 的值

[...]:表示括号内为可选项

{...}\*:表示括弧内的项可出现 0 次,1 次或多次

|:表示分割两个可选项

⇒:表示表达式的值

大写字母开头的自变量:表示不对其求值

小写字母开头的自变量:表示对其求值

### 2.1.2 LISP 的基本功能

#### 1. 算术函数

(1)(+ &.rest numbers)⇒sum

+表示函数名,本函数自变量的数目可以有 0 个或多个,number 表示自变量为数。函数的功能是,将各自变量求值,各自变量的值必为数,再将这些数相加,其和作为本函数的值。例如

(+ 1 2 3)⇒6

(+)⇒0

若文字原子 a 的值为 4,b 的值为 3,则

(+ a b)⇒7

注意:文字原子的值应预先赋予,而数的值就是该数本身。

(2)(- number &.rest more-numbers)⇒difference

本函数至少应有一个自变量,其功能是实现多个数相减。例如

(- 1 2)⇒-1

(- 1)⇒-1

(- 10 5 2 5)⇒-2

(- 10 5(+ 2 5))⇒-2

上述最后一个函数中,第三个自变量是一个+函数。对-函数的第三个自变量的求值就是对+函数求值,将其值作为-函数第三个自变量的值。

(3)(\* &.rest numbers)⇒product

功能是实现多个数相乘。例如

(\* 2 3)⇒6

(\* )⇒1

(4)(/ number &.rest more-numbers)⇒quotient

功能是实现多个数相除。例如