

第六章 基于知识的系统

- 第6.1节 知识系统的一般概念
- 第6.2节 知识系统的开发
- 第6.3节 知识系统的开发工具与环境介绍
- 第6.4节 知识系统的实例
- 第6.5节 问题求解的结构化组织

第6.1节 知识系统的一般概念

- 6.1.1 知识系统的特点
- 6.1.2 知识系统的组成
- 6.1.3 知识系统的工作方式和评价

6.1.1 知识系统的特点

- **知识系统的特点**主要如下：

- (1) **具有求解问题所需的专门知识**。专门知识分为两大类：

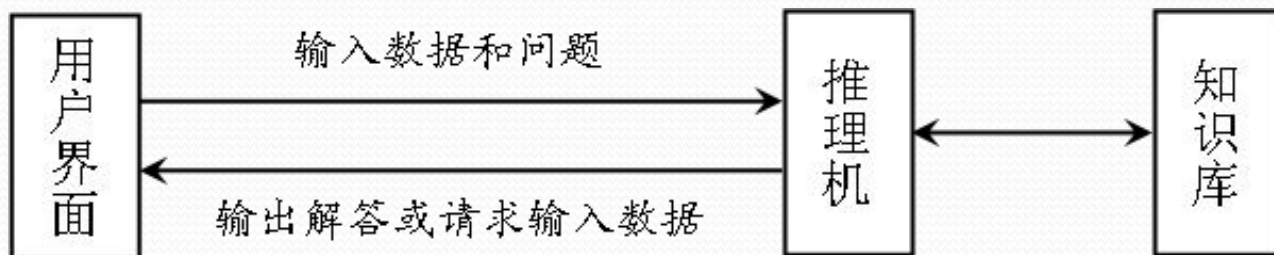
- i 应用领域的基本原理和常识，它是专门知识的主部，是求解问题的基础；
- ii 领域专家求解问题的经验知识，它是专家对使用前者求解问题的高度集中、抽象和浓缩的描述，是求解问题的性能关键。

- (2) **具有使用专门知识的符号推理能力**。符号推理能力主要包括知识表示和推理技术。

- i 表示知识的符号结构由适当的推理机制加以存取和使用；
- ii 推理机制的设计包括问题求解的组织和推理控制，主要取决于应用领域的特征和问题求解任务的要求。

6.1.2 知识系统的组成

- **知识系统的组成**主要包括下述三个部分：
 - (1) **知识库** 用于存储求解问题所需的以一定的符号结构表示的专门知识；
 - (2) **推理机** 根据输入的问题以及描述问题求解初始状态的数据，取用知识库中的知识进行推理，并输出最终解答；
 - (3) **用户界面** 作为人机交互的接口。用户界面不是知识系统的主要部分，但是对知识系统的推广应用具有重要影响。



6.1.3 知识系统的工作方式和评价

- 知识系统与一般的应用程序不同。
- 一般的应用程序把问题求解的知识隐含地编入程序，而知识系统则把其应用领域的问题求解知识单独组成一个实体，即为知识库。知识库的处理是通过与知识库分开的控制策略进行的。
- 更明确地说，一般应用程序把知识组织为两级：数据和程序；大多数**知识系统将知识组织成三级：数据、知识库和控制。**

- 知识系统执行的问题求解任务是知识密集型的。常见的任务主要有解释、诊断、监控、预测、规划和设计等。
 - (1) **解释** 分析测量或观测到的数据以确定其含义。
 - (2) **诊断** 从被诊断的故障现象确诊可能的故障根源。
 - (3) **监控** 连续观测和解释反应设备和系统现状的数据，在发现异常时，提出求解建议，给出异常原因，甚至对设备和系统进行自动控制。
 - (4) **预测** 从事物的现状和发展趋势，预言将来的状态和行为。
 - (5) **规划** 制定满足各种约束条件的行动方案。
 - (6) **设计** 完成或参与完成建筑、电子线路等设计任务。

- 知识系统执行问题求解任务时遇到的主要困难如下：
 - (1) 大型解空间；
 - (2) 试探性推理；
 - (3) 数据的不准确性。

- 知识系统面向复杂的结构不良问题，**解释**求解过程及结果的合理性是知识系统应具备的能力。
- 知识系统的**解释方式**主要依据**规则追踪**的方法，规则追踪就是把问题求解过程中激活使用的规则按激活的次序显示给用户，目前已为大多数知识系统所应用。更优雅的方法是将规则转变为自然语言的形式显示，以便用户理解。
- 规则追踪适合于审查推理过程的正确性，可以作为维护和扩充知识系统的调试手段，但是因为未按应用领域的基本原理和常识证实推理结果的正确性，推理结果往往不能令人信服。高级的解释功能应具有按领域基本原理和常识**重构解答**的能力。重构的推理路径未必是实际的推理路径但却能令人信服地验证解答的合理性，从而表示出高级的智能行为。

- **知识系统的评价**包括很多方面，下述三点是最重要的：
 - (1) **计算** 主要包括运行速度、存储空间、可扩展性、可移植性和是否易于与其它软件集成。
 - (2) **感观** 主要包括易用性、可理解性、自然性和是否提供联机帮助和解释。
 - (3) **性能** 主要包括能力范围、得出非正确解答的比率、时间和资金的节省等。
- 在实践中，知识系统与常规软件的评价使用同样的方式。通常用大量的实际案例进行经验性测试，直到用户信服系统的有效性才正式投入应用。

第6.2节 知识系统的开发

- 6.2.1 知识获取及其分类
- 6.2.2 知识系统的开发步骤

6.2.1 知识获取及其分类

- **知识获取的任务**是把已有的知识（包括经验、事实、规则等等）从人脑中或者书本中总结和抽取出来并转换成某种形式的表示。广义地，把从已获取的知识和实例以某种方式产生新知识的任务（即机器学习）也包括在知识获取之中。
- 知识获取研究的主要问题是：
 - (1) 已有知识的理解、认识、选择、抽取、汇集、分类和组织的方法；
 - (2) 从已有知识和实例中产生新知识，包括从外界学习新知识的机理和方法；
 - (3) 检查或保持已获取知识集合的一致性（或无矛盾性）和完全性约束的方法；
 - (4) 尽量保证已获取的知识集合无冗余的方法。

- 知识获取可以从不同的角度进行分类。
 - 从获取的主体角度来看，知识获取可以分成
 - (1) **主动式获取**，也称为直接获取。知识系统根据领域专家给出的数据和资料利用归纳程序之类的工具软件直接自动获取或产生知识并装入知识库中。
 - (2) **被动式获取**，也称为间接获取。通过一个中介人（如知识工程师或者用户）并采用知识编辑器之类的工具，把知识传授给知识系统。
 - 按知识系统获取知识的工作方式，可以分成
 - (1) **交互式获取**。
 - (2) **非交互式（自主式）获取**。

交互式知识获取在知识获取中不断与人交互，对用户或知识工程师有较大的透明度和控制能力，适合于从领域专家大脑中获取知识。

- 按知识获取的策略或机理，可以分成
- (1) 机械照搬式知识获取； (2) 条件反射式知识获取；
- (3) 教学式（传授式）知识获取； (4) 演绎式知识获取；
- (5) 归纳式知识获取； (6) 猜想证实式知识获取；
- (7) 反馈修正式知识获取； (8) 类比式知识获取；
- (9) 联想式知识获取； (10) 外延式知识获取；
- (11) 直觉发现式知识获取；
- (12) 灵感与偶发式知识获取，
- 等等。

6.2.2 知识系统的开发步骤

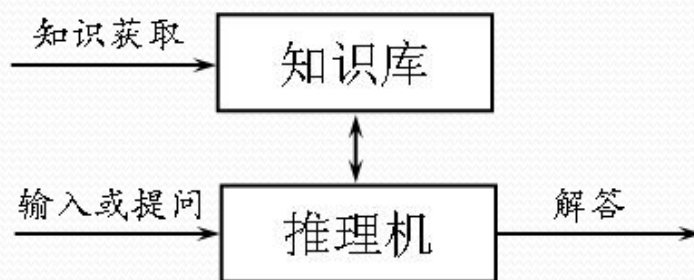
- 构造知识系统的关键在于
 - i 获取问题求解所需要的知识;
 - ii 设计知识系统的体系结构。
- 知识系统的体系结构在很大的程度上依赖于知识系统中的知识，特别是那些关于问题求解组织和推理控制的知识。
- 知识获取是一个十分困难而又耗时的过程，一直被认为是构造知识系统特别是大型知识系统的瓶颈。其主要原因是面向计算机实现的知识表示语言抽象级别太低，与在较高抽象级别观察问题的人的思维方式严重失配，无法提供有效的手段去系统化和结构化地描述问题求解的组织和推理机制。

- 在知识系统的建造过程中，必须对下述三点给予足够的重视：
 - (1) 知识的表示；
 - (2) 知识的使用；
 - (3) 决策的解释。
- 对于**知识的表示**，领域专家表达知识主要利用以下3个表达成分：观测或观察，假设或结论，推理或决策规则。
- 在领域专家看来，观测和假设之间是严格区分的。观测是观察或量测，它的值可以是“真”，“假”，数字或“不知道”等形式。假设是由系统推理得到的可能结论。通常假设附有不确定性的量度。

- **产生式规则**是决策规则最为常用的表示形式，可根据观测和假设之间的逻辑关系分成3类：
 - ① **从观测到观测的规则（FF规则）**，FF规则规定那些可从已确定的观测直接推导出来的观测的真值。因为通过把观测和假设相组合可以描述功能更强的产生式规则形式。
 - ② **从观测到假设的规则（FH规则）**，在许多用于分类的专家系统中，产生式规则可对产生式结论的可信程度进行量度。
 - ③ **从假设到假设的规则（HH规则）**，HH规则用来规定假设之间的推理。

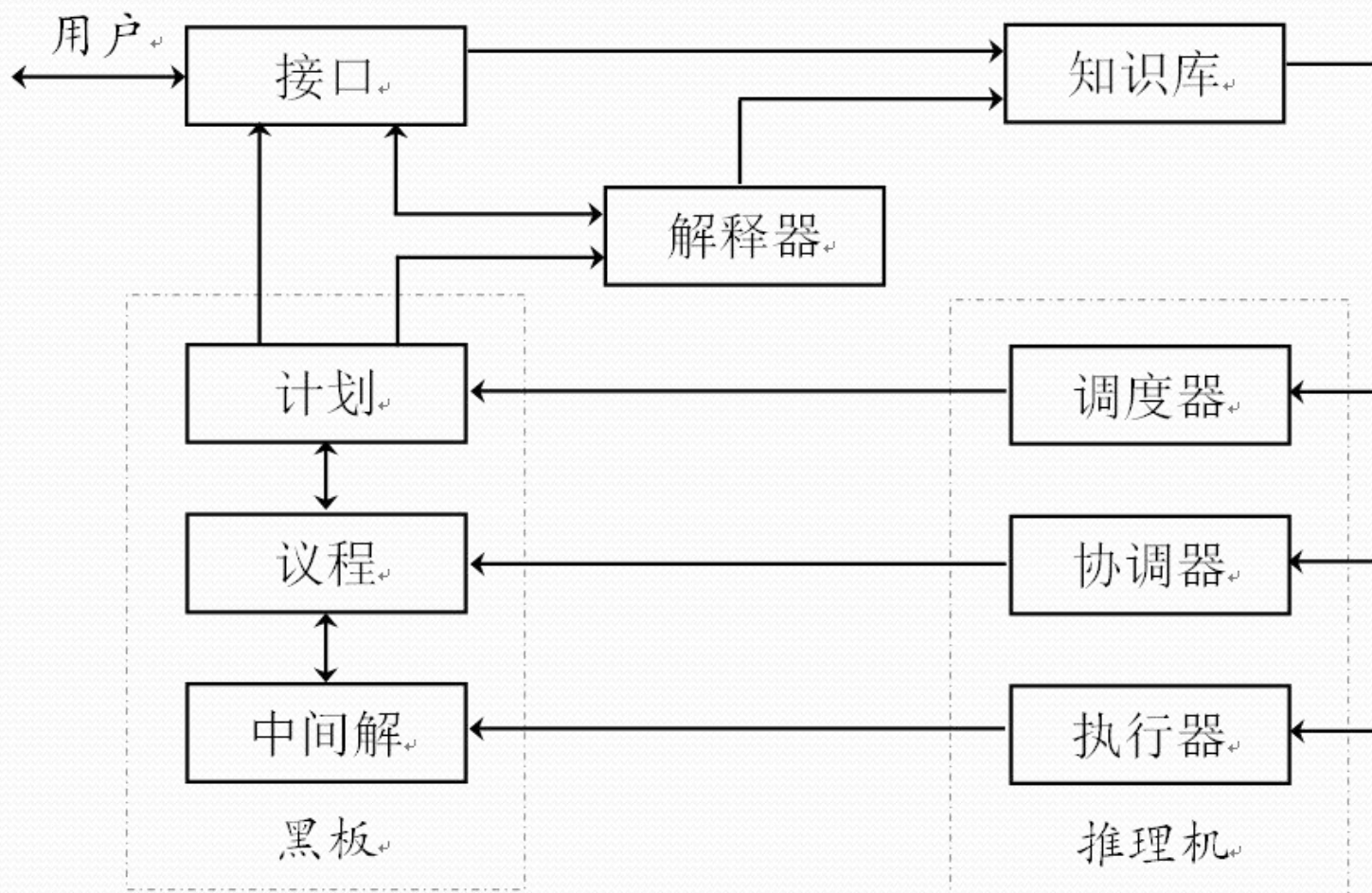
- 对于**知识的使用**，建立知识系统还不是一门精确的科学，因此必须力图抽取专家推理过程中的关键内容，并且尽可能准确而简洁地表示这些知识。
 - (1) 应根据专家的意见，在控制策略中确定规则的等级和规则选用方法；
 - (2) 清楚地组织问题，使问题包含尽可能多的结构，建立良好的询问策略。
- 对于**决策的解释**，系统的设计者和使用者都需要系统对它所做出的决策给予解释。但是他们对决策解释的要求又各不相同。不仅要关心对系统设计者的解释，也应关心对系统使用者的解释。
- 一种解释方法是用语句来说明结论。系统所用的假设可能是任何形式的包含说明和建议的语句。有时系统的设计者可以预先提出某些适合于给定假设的解释。

- **知识系统的结构**是指知识系统各组成部分的构造方法和组织形式。系统结构选择恰当与否，与知识系统的适用性和有效性密切相关，一般要根据系统的应用环境和所执行任务的特点而定。
- 简单地说，知识系统的结构如下：



知识系统的结构

- 理想的知识系统的结构如下图所示。



- 由于每个专家系统所需要完成的任务和特点不相同，其系统结构也不尽相同，一般只具有图中部分模块。
- (1) **接口**是人与系统进行信息交流的媒介，它为用户提供了直观方便的交互作用手段。
- (2) **黑板**是用来记录系统推理过程中用到的控制信息、中间假设和中间结果的数据库。它包括计划、议程和中间解三个部分。
- (3) **知识库**包括两部分内容。一部分是已知的同当前问题有关的数据信息；另一部分是进行推理时要用到的一般知识和领域知识。

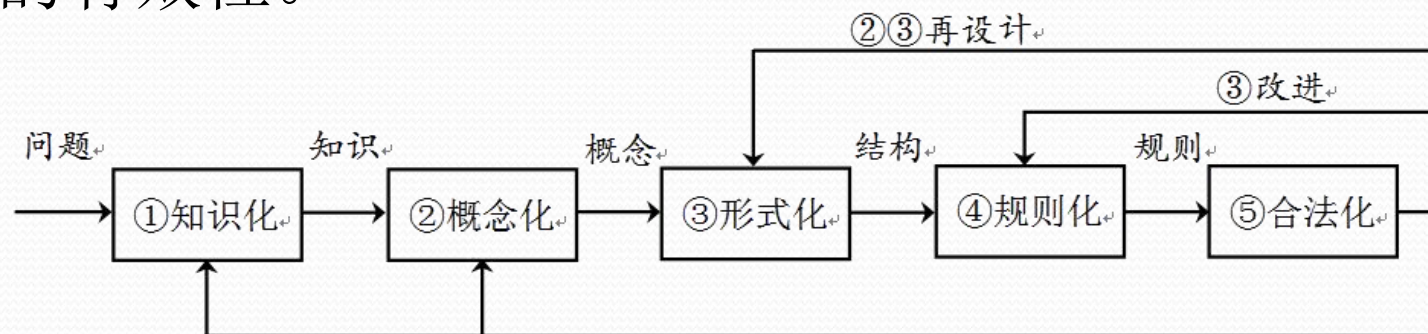
- (4) **调度器**按照系统建造者所给的控制知识，从议程中选择一个项作为系统下一步要执行的动作。**执行器**应用知识库中的及黑板中记录的信息，执行调度器所选定的动作。**协调器**的主要作用就是当得到新数据或新假设时，对已得到的结果进行修正，以保持结果前后的一致性。
- (5) **解释器**的功能是向用户解释系统的行为，包括解释结论的正确性及系统输出其它候选解的原因。

- 建立系统的一般步骤如下：

- (1) **设计初始知识库**，包括：

- ① 问题知识化，即辨别所研究问题的实质，如要解决的任务是什么，它是如何定义的，可否把它分解为子问题或子任务，它包含哪些典型数据等。
- ② 知识概念化，即概括知识表示所需要的关键概念及其关系，如数据类型、已知条件（状态）和目标（状态）、提出的假设以及控制策略等。
- ③ 概念形式化，即确定用来组织知识的数据结构形式，应用人工智能中各种知识表示方法把与概念化过程有关的关键概念、子问题及信息流特性等变换为比较正式的表达，它包括假设空间、过程模型和数据特性等。
- ④ 形式规则化，即编制规则、把形式化了的知识变换为由编程语言表示的可供计算机执行的语句和程序。

- ⑤ 规则合法化，即确认规则化了知识的合理性，检验规则的有效性。



□(2) 原型机的开发与试验

- 在选定知识表达方法之后，即可着手建立整个系统所需要的实验子集，它包括整个模型的典型知识，而且只涉及与试验有关的足够简单的任务和推理过程。

□(3) 知识库的改进与归纳

- 反复对知识库及推理规则进行改进试验，归纳出更完善的结果。经过相当长时间的努力，使系统在一定范围内达到人类专家的水平。

第6.3节 知识系统的开发工具与 环境介绍

- 6.3.1 知识系统的开发工具介绍
- 6.3.2 人工智能程序设计语言介绍

6.3.1 知识系统的开发工具介绍

- 知识系统开发工具主要有以下四种：

□(1) 骨架（外壳）型开发工具

- 专家系统一般都有推理机和知识库两部分，而规则集存于知识库内。在一个理想的专家系统中，推理机完全独立于求解问题领域。系统功能上的完善或改变，只依赖于规则集的完善和改变。由此，借用以前开发好的专家系统，将描述领域知识的规则从原系统中“挖掉”，只保留其独立于问题领域知识的推理机部分，这样形成的工具称为骨架型工具。这类工具因其控制策略是预先给定的，使用起来很方便，用户只须将具体领域的知识明确地表示成为一些规则就可以了。
- 因其程序的主要骨架是固定的，除了规则以外，用户不可改变任何东西，因而骨架型工具存在一些有待解决的问题，影响它的广泛应用。

□(2) 语言型开发工具

- 语言型工具提供给用户的是建立专家系统所需要的基本机制，其控制策略也不固定于一种或几种形式，用户可以通过一定手段来影响其控制策略。因此，语言型工具的结构变化范围广泛，表示灵活，所适应的范围要比骨架型工具广泛得多。

□(3) 构造辅助工具

- 系统构造辅助工具由一些程序模块组成，有些程序能帮助获得和表达领域专家的知识，有些程序能帮助设计正在构造的专家系统的结构。它主要分两类，一种是设计辅助工具，另一种是知识获取辅助工具。

□(4) 支撑环境

- 支撑设施是指帮助进行程序设计的工具，它常被作为知识工程语言的一部分。工具支撑环境仅是一个附带的软件包，以便使用户界面更友好。它包括四个典型组件：调试辅助工具、输入输出设施、解释设施和知识库编辑器。

- 尽管知识系统的设计取决于应用领域的特征和问题求解任务的要求，但是知识系统之间往往仍具有很多相同的共性。因而知识系统的开发工具仍然是必要的。
- 一般来说，对**知识系统开发工具的评价**只要考虑以下几个方面：
 - (1) 是否满足要开发的知识系统的应用领域的特征；
 - (2) 是否符合问题求解任务的要求；
 - (3) 是否能提高知识系统开发的效率和质量。

6.3.2 人工智能程序设计语言介绍

- 知识的处理虽然有时也包含不少对数据的处理，但主要的是符号的处理和逻辑的推理。正因如此，一些面向数据处理的高级程序设计语言在解决知识推理中无论在对象结构的描述还是在可施行的各种操作方面都不够方便。
- 在人工智能研究的初期人们就开始寻求适合解决人工智能问题的程序设计语言。在人工智能领域，LISP是一个先驱，PROLOG、SMALLTALK、OPS等也是在人工智能领域很有影响的通用程序设计语言。
- 这里，我们只对LISP语言和PROLOG语言做简单介绍。

- **LISP语言**是J. McCarthy等人在20世纪60年代根据Lambda 演算理论设计的一种函数式程序设计语言。
- **LISP语言的主要特点**是：
 - (1) 在LISP语言中，程序与其加工的数据都统一地用一种符号表达式（S-表达式）进行表示，因此LISP作为一种符号处理语言，广泛地应用于人工智能领域的程序设计中。
 - (2) LISP语言中最自然的控制结构是递归，也能够进行迭代，适用于许多问题求解任务。
 - (3) 大多数LISP系统可以用交互的方法运行。

- LISP语言最基本的语法单位是原子，并在原子的基础上定义表和S-表达式，S-表达式是LISP语言中唯一的数据结构。
- LISP语言的S-表达式用BNF范式定义如下：

$\langle S\text{-表达式} \rangle ::= \langle \text{原子} \rangle \mid \langle \text{表} \rangle$

$\langle \text{表} \rangle ::= (\langle S\text{-表达式} \rangle^*)$

$\langle \text{原子} \rangle ::= \langle \text{符号原子} \rangle \mid \langle \text{数原子} \rangle$

$\langle \text{符号原子} \rangle ::= \langle \text{字母} \rangle \mid \langle \text{符号原子} \rangle \langle \text{字母} \rangle \mid \langle \text{符号原子} \rangle \langle \text{数字} \rangle$

$\langle \text{数原子} \rangle ::= \langle \text{整数} \rangle \mid \langle \text{浮点数} \rangle$

...

- LISP语言中符号T和NIL被特别地定义为逻辑真和逻辑假。

- 在LISP语言中，一个程序也是S-表达式，其元素是一个或多个函数的定义。
- LISP程序被定义成一串函数定义和一串函数的计算和调用，各函数之间的关系只体现在函数的计算和调用之中。在其运行机理上与C程序的运行方式类似。

- **PROLOG语言**是法国Marseille大学的 Alain Colmerauer和他的助手在20世纪70年代发明的一种以一阶谓词逻辑中的Horn演算为基础的描述式语言。
- PROLOG语言的主要特点是：
 - (1) 具有极强的描述能力；
 - (2) 自动实现模式匹配和回溯；
 - (3) 具有递归功能。

- PROLOG只有三种基本的语句：**事实、规则和询问**。
它们在本质上都是Horn子句。PROLOG采用深度优先和回溯的方法自动进行搜索求解。
- (1) PROLOG中的事实的一般形式是
$$\langle \text{关系名} \rangle [(\text{对象1}, \text{对象2}, \dots, \text{对象}n)]$$
- (2) PROLOG中的规则的一般形式是
$$\langle \text{规则头} \rangle :- \langle \text{规则体} \rangle$$
- (3) PROLOG中的询问的一般形式是
$$?- \langle \text{目标} \rangle$$
- 为了控制上的目的，PROLOG还定义了两个内部谓词cut和fail，不同的PROLOG实现可能进一步提供了某些其它的数据类型、谓词和控制描述机制。

- 例 求两个正整数X和Y的最高公因 gcd(X,Y,Z)的 PROLOG程序。

解 谓词gcd(X,Y,Z)的定义是

$$\text{gcd}(X,Y,Z) = \begin{cases} 1, & \text{若} Z \text{ 是 } X \text{ 和 } Y \text{ 的最高公因;} \\ 0, & \text{否则。} \end{cases}$$

则求解谓词gcd(X,Y,Z)的PROLOG程序如下：

$\text{gcd}(X,0,X) : -!.$

$\text{gcd}(0,Y,Y) : -!.$

$\text{gcd}(X,Y,Z) : -X \geq Y, W \text{ is } X - Y, \text{gcd}(W,Y,Z).$

$\text{gcd}(X,Y,Z) : -\text{gcd}(Y,X,Z).$

如果询问?-gcd(133,301,X)., PROLOG能够求出X=7。

第6.4节 知识系统的实例

- 6.4.1 专家系统及其分类;
- 6.4.2 专家系统MYCIN概述

6.4.1 专家系统及其分类；

- 最常见的知识系统中是所谓的专家系统。**专家系统**是一个含有大量的某个领域专家水平的知识与经验智能计算机程序系统，能够利用人类专家的知识 and 解决问题的方法来处理该领域问题。简而言之，专家系统是一种模拟人类专家解决领域问题的计算机程序系统。
- **专家系统的主要特点**如下：
 - (1) **启发性**：专家系统能运用专家的知识与经验进行推理、判断和决策。
 - (2) **透明性**：专家系统能够解释本身的推理过程和回答用户提出的问题，以便让用户能够了解推理过程，提高对专家系统的信赖感。
 - (3) **灵活性**：专家系统能不断地增长知识，修改原有知识，不断更新。

● 专家系统的主要优点如下：

- (1) 专家系统能汇集多领域专家的知识和经验以及他们协作解决重大问题的能力。
- (2) 可以使专家的专长不受时间和空间的限制，以便推广珍贵和稀缺的专家知识与经验。
- (3) 专家系统能够高效、准确、周到、迅速和不知疲倦地进行工作，在解决实际问题时不受周围环境的影响，也不可能遗漏忘记。
- (4) 专家系统的研究不仅能提高领域研究的水平，也能够促进整个科学技术的发展，具有巨大的经济效益和社会效益。

- 专家系统按照不同的角度可以进行不同的分类。

□（一）如果按照专家系统所从事的工作任务，可以将专家系统分成**解释专家系统、预测专家系统、诊断专家系统、规划专家系统、监视专家系统、控制专家系统、调试专家系统、教学专家系统、修理专家系统**等等。

➤（1）**解释专家系统**

解释专家系统任务是通过通过对已知信息和数据的分析与解释，确定它们的涵义。其主要特点是数据量很大，常不准确、有错误、不完全能从不完全的信息中得出解释，并能对数据做出某些假设，推理过程可能很复杂和很长。

➤（2）**预测专家系统**

预测专家系统的任务是通过通过对过去和现在已知状况的分析，推断未来可能发生的情况。其主要特点是系统处理的数据随时间变化，且可能是不准确和不完全，系统需要有适应时间变化的动态模型。

➤ (3) 诊断专家系统

诊断专家系统的任务是根据观察到的情况(数据)来推断出某个对象机能失常(即故障)的原因。其主要特点是能够了解被诊断对象或客体各组成部分的特性以及它们之间的联系,能够区分一种现象及其所掩盖的另一种现象,能够向用户提出测量的数据,从不确切信息中得出尽可能正确的诊断。

➤ (4) 设计专家系统

设计专家系统的任务是寻找出某个能够达到给定目标的动作序列或步骤。其主要特点是从多种约束中得到符合要求的设计;系统需要检索较大的可能解空间;能试验性地构造出可能设计;易于修改;能够使用已有设计来解释当前新的设计。

➤ (5) 规划专家系统

规划专家系统的任务是寻找出某个能够达到给定目标的动作序列或步骤。其主要特点是所要规划的目标可能是动态的或静态的，需要对未来动作做出预测，所涉及的问题可能很复杂。

➤ (6) 监视专家系统

监视专家系统的任务是对系统、对象或过程的行为进行不断观察，并把观察到的行为与其应当具有的行为进行比较，以发现异常情况，发出警报。其主要特点是系统具有快速反应能力，发出的警报要有很高的准确性，能够动态地处理其输入信息。

➤ (7) 控制专家系统

控制专家系统的任务是自适应地管理一个受控对象或客体的全面行为，使之满足预期要求。其主要特点是控制专家系统具有解释、预报、诊断、规划和执行等多种功能。

➤ (8) 调试专家系统

调试专家系统的任务是对失灵的对象给出处理意见和方法。其主要特点是同时具有规划、设计、预报和诊断等专家系统的功能。

➤ (9) 教学专家系统

教学专家系统的任务是根据学生的特点、弱点和基础知识，以最适当的教案和教学方法对学生进行教学和辅导。其主要特点是需要同时具有诊断和调试等功能，并具有良好的的人机界面。

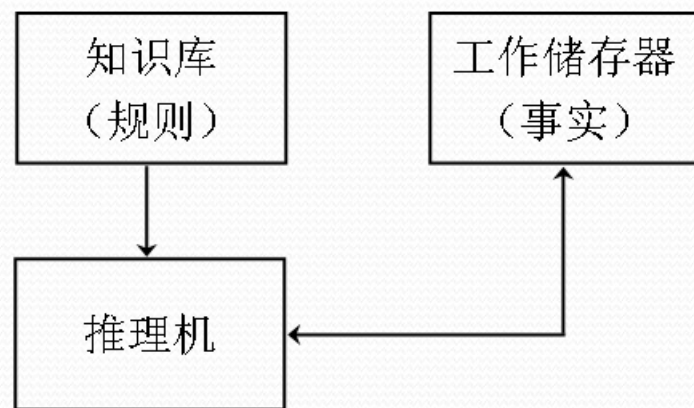
➤ (10) 修理专家系统

修理专家系统的任务是对发生故障的对象(系统或设备)进行处理，使其恢复正常工作。其主要特点是具有诊断、调试、计划和执行等功能。

□（二）如果按照专家系统的构建技术，可以将专家系统分成**基于规则的专家系统、基于框架的专家系统、基于模型的专家系统**等。

➤(1) **基于规则的专家系统**

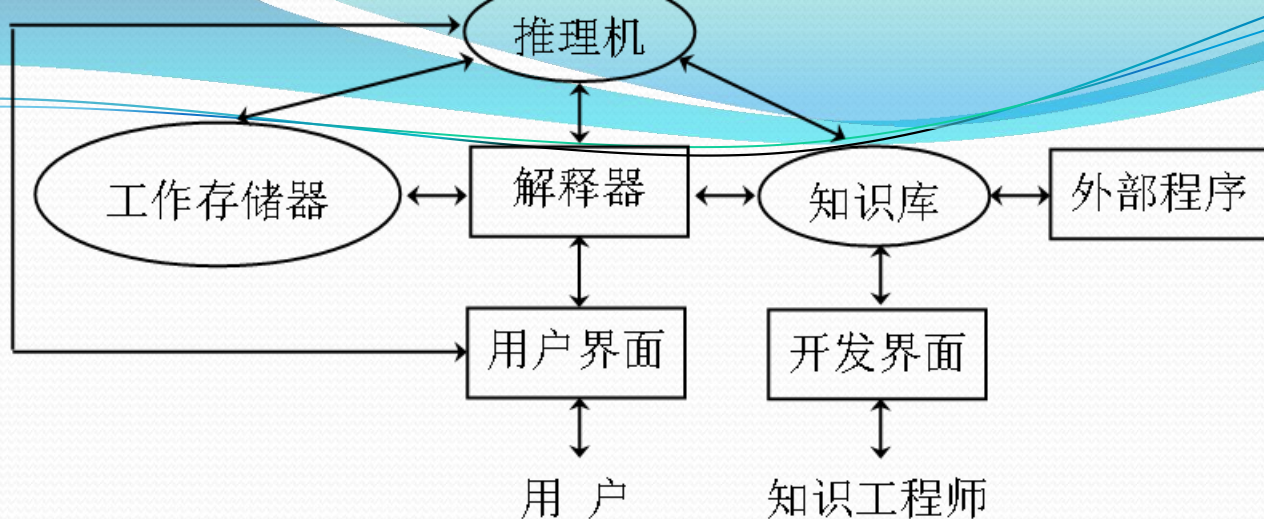
✓ 基于规则的专家系统是个计算机程序，该程序使用一套包含在知识库内的规则对工作存储器内的具体问题信息（事实）进行处理，通过推理机推断出新的信息。其工作模型如图所示。



基于规则专家系统的工作模型

✓ 基于规则的专家系统不需要一个人类问题求解的精确匹配，而能够通过计算机提供一个复制问题求解的合理模型。

- 一个基于规则专家系统的完整结构如图示。



基于规则专家系统的结构

- 其中，知识库、推理机和工作存储器是构成本专家系统的核心。
- 系统的主要部分是知识库和推理引擎。根据到目前为止讨论的推理系统，知识库由谓词演算事实和有关讨论主题的规则构成。推理引擎由所有操纵知识库来演绎用户要求的信息的过程构成——如归结、前向链或反向链。
- 用户接口可能包括某种自然语言处理系统，它允许用户用一个有限的自然语言形式与系统交互。也可用带有菜单的图形接口界面。
- 解释子系统分析被系统执行的推理结构，并把它解释给用户。

➤ (2) 基于框架的专家系统

- ✓ 基于框架的专家系统建立在框架的基础之上，采用面向目标编程技术，框架的设计和面向目标的编程共享许多特征。在设计基于框架系统时，专家系统的设计者们把目标叫做框架。
- ✓ 基于框架的专家系统是个计算机程序，该程序使用一组包含在知识库内的框架对工作存储器内的具体问题信息进行处理，通过推理机推断出新的信息。
- ✓ 基于框架专家系统的主要设计步骤与基于规则的专家系统相似。主要**差别**是在看待和使用知识采用框架而不是采用规则来表示知识。在设计基于框架的专家系统时，把整个问题和每件事想像为编织起来的事物。在辨识事物之后，寻找把这些事物组织起来的方法，其设计是高度交互的过程。

➤ (3) 基于模型的专家系统

- ✓ 对人工智能的研究内容有着各种不同的看法。有一种观点认为：人工智能是对各种定性模型的获得、表达及使用的计算方法进行研究的学问。基于该观点人们提出了基于模型的专家系统。在诸多模型中，人工神经网络模型的应用最为广泛。
- ✓ 神经网络模型与专家系统集成模式主要有以下三种：
 - ◆ ① **神经网络支持专家系统** 以传统的专家系统为主，以神经网络的有关技术为辅。
 - ◆ ② **专家系统支持神经网络** 以神经网络的有关技术为核心，建立相应领域的专家系统，采用专家系统的相关技术完成解释等方面的工作
 - ◆ ③ **协同式的神经网络专家系统** 针对大的复杂问题，将其分解为若干子问题，针对每个子问题的特点，选择用神经网络或专家系统加以实现，在神经网络和专家系统之间建立一种耦合关系。

- 神经网络模型从知识表示、推理机制到控制方式，与目前专家系统中的基于逻辑的心理模型有本质的区别。
- 神经网络专家系统的知识表示是一种隐式表示，自动获取模块输入、组织并存储专家提供的学习实例、选定神经网络的结构、调用神经网络的学习算法，为知识库实现知识自动获取。当新的学习实例输入后，知识获取模块通过对新实例的学习，自动获得新的网络权值分布，从而更新了知识库。
- 它的推理是个正向非线性数值计算过程，同时也是一种并行推理机制，神经网络各输出节点的输出是数值，因而需要一个解释器对输出模式进行解释。

6.4.2 专家系统MYCIN概述

- MYCIN是一个通过提供咨询服务来帮助诊治细菌感染性疾病的专家系统，于1972年开始研制，于1974年基本完成并投入实际应用。MYCIN是最有影响力的专家系统，对推动知识工程以及专家系统学科的建立和发展具有重要的影响。
- MYCIN是典型的产生式系统，由规则库、综合数据库和控制系统三个部分组成。MYCIN采用基于规则的逆向推理方式。
- MYCIN使用INTERLISP编程，运行于DEC PDP-10的操作系统TENEX下。MYCIN的执行代码约 50KB，规则库占 16KB，系统的咨询服务（包括提供解释）平均每次20分钟。

- **MYCIN的知识库**以前提-动作型产生式规则来表示诊断和治疗细菌感染性疾病的专家医学知识，以实现专家级诊断和治疗能力。
- MYCIN的规则形如：

$\langle \text{规则} \rangle ::= \text{RULE } \langle \text{规则号} \rangle$

$\text{PREMISE } \left(\$AND \{ \langle \text{条件} \rangle \}^+ \right)$

$\text{ACTION } \{ \langle \text{条件} \rangle \}^+$

$\langle \text{条件} \rangle ::= \langle \text{简单条件} \rangle \left| \left(\$OR \{ \langle \text{简单条件} \rangle \}^+ \right) \right|$

- MYCIN规则中的条件以MYCIN提供的4类简单函数来表示，其中最常用的是SAME函数，其BNF范式描述为：

$(SAME \langle \text{对象} \rangle \langle \text{属性} \rangle \langle \text{值} \rangle)$

- MYCIN规则中的动作也以简单函数表示，最常用的是CONCLUDE，其BNF范式描述为：

$(CONCLUDE \langle \text{对象} \rangle \langle \text{属性} \rangle \langle \text{值} \rangle TALLY \langle \text{结论} CF \rangle)$

- **MYCIN系统的推理机**基于可信度的不确定性推理，采用逆向推理和深度优先的搜索策略。
- 在咨询模块启动以后，首先在综合数据库中建立上下文对象patent-1作为上下文树（与或树）的根结点，以建立病人的治疗方案regimen为目标逆向形成推理规则链，直到推理规则链的末端规则的前提包含的条件都能直接由原始证据（医生提供的观测结果）证实为止。

第6.5节 问题求解的结构化组织

- 6.5.1 结构化组织的必要性
- 6.5.2 问题求解的事务表和黑板法组织
- 6.5.3 问题求解建模与新一代知识系统技术
- 6.5.4 新型知识系统介绍

6.5.1 结构化组织的必要性

- 知识系统要解决的问题往往是具有下述特点：
 - (1) **复杂性** 系统由大量部件构成，并且部件之间以非简单的方式交互作用；
 - (2) **结构不良** 目标不好严格定义，缺少可预先确定的从初始状态到目标状态的决策手段。
 - 由于产生式系统具有
 - (1) 难以扩展；
 - (2) 选择规则低效；
 - (3) 控制策略不灵活；
 - (4) 表示形式单一
- 的缺点，将求解复杂问题的知识按**模块化组织**可能比较合适。

6.5.2 问题求解的事务表和黑板法组织

- 知识的模块化划分可以是面向动作的，也可以是面向对象的。
- **事务表**也称为**任务表**，是一种面向动作的问题求解组织方式，事务表中的每个元素都是一个等待执行的任务，并需要记载相应的执行紧迫程度。事务表的意义在于它允许各个推理模块经由事务表相互通信，以合作的方式进行问题求解。
- **黑板法**也是一种面向动作的组织问题求解组织方式，黑板是一个公共数据区，可供所有的推理模块访问和修改，从而渐增性地形成对问题的解答。

6.5.3 问题求解建模与新一代知识系统技术

- 传统的关键将开发和维护知识系统的困难归因于“知识获取瓶颈”。现在人们逐渐地认识到，将知识获取视为建模活动有助于克服困难，也就是说，作为问题求解建模的组成部分，知识获取本身不再是困难的事情，真正困难的在于建模分析—分析问题的求解行为和应用领域世界的模型。
- 分析问题求解模型最重要的方面是**概念模型**。概念模型提供基本的术语和术语合成法则描述问题求解所涉及的实体、实体的属性和关系以及在问题求解中的作用。
- 概念模型往往隐含于推理机和知识获取工具（如交互式建模语言）中，以控制问题求解过程和指导知识获取。

- 概念模型可归纳为三个主要范畴：
 - (1) **基于表示的概念模型** 直接反映与推理机关键的符号级表示，是涉及符号级表示语言和推理机制的基础，由建模者完全负责知识级分析。其主要缺点是忽略了人的认识行为处于知识级这一特点。
 - (2) **基于方法的概念模型** 提供预先定义的方法，使用户建模的注意力集中在获取实现方法所需要的领域的特有知识而不是规则和框架等符号级表示结构，指导建模者做知识级分析。其主要缺点是容易与应用领域失配。
 - (3) **基于任务的概念模型** 面向特别种类的任务，直接刻画任务结构而不是执行任务的方法，以指导建模者进行知识级建模分析。

- 以问题求解建模为核心的知识结构化组织技术推动了人工智能应用研究向实用化和高性能的新一代知识系统的方向发展。
- 新一代知识系统技术除**问题求解建模**以外，还包括：
 - (1) **定性物理** 以开发定性演算的物理系统行为模型为目标，描述应用领域的基本原理和常识，用定性方程表示控制物理系统行为的法则，并通过定性演算和状态转换规则语言和仿真物理系统的定性行为。
 - (2) **基于模型的推理** 刻画物理系统的结构、功能、因果和行为模型。
 - (3) **深浅层推理的综合** 以浅层推理作为核心和主干，并将深层推理作为补充和后盾。增强系统的自适应性。
 - (4) **功能化体系结构** 以功能单元来构造基于知识的问题求解系统，而不是以预先固定的概念模型强加于系统。实现智能软件重用，加快知识系统的开发。
 - (5) **知识级问题求解建模** 研究知识级问题求解建模技术，避免概念模型与特定求解任务和领域特征失配或者基于任务的方法不够完备的问题。

6.5.4 新型知识系统介绍

- 随着计算机科学与技术的发展，在专家系统的构建中不断出现新的技术和方法，产生了新型的专家系统，如分布式专家系统、协同式专家系统等。
- 新型专家系统的主要特征是：
 - (1) **并行与分布处理**

基于各种并行算法，采用各种并行推理和执行技术，适合在多处理器的硬件环境中工作，即具有分布处理的功能。
 - (2) **多专家系统协同工作**

在这种系统中，有多个专家系统协同合作。
 - (3) **高级语言和知识语言描述**

专家系统生成系统就能自动或半自动地生成所要的专家系统。

➤ (4) 具有自学习功能

新型专家系统应提供高级的知识获取与学习功能。

➤ (5) 引入新的推理机制

在新型专家系统中，除演绎推理之外，还应有归纳推理，各种非标准逻辑推理，以及各种基于不完全知识和模糊知识的推理等等。

➤ (6) 具有自纠错和自完善能力

为了排错必须首先有识别错误的能力，为了完善必须首先有鉴别优劣的标准。

➤ (7) 先进的智能人机接口

理解自然语言，实现语声、文字、图形和图象的直接输入输出是如今人们对智能计算机提出的要求。

- **分布式专家系统**的主要目的是把一个专家系统的功能经分解以后分布到多个处理器上去并行地工作，从而在总体上提高系统的处理效率。分布式专家系统可以工作在紧耦合的多处理器系统环境中，也可工作在松耦合的计算机网络环境里，所以其总体结构在很大程度上依赖于其所在的硬件环境。
- 设计和实现分布式专家系统，需要解决的问题主要是：
 - (1) **功能分布** 把分解得到的系统各部分功能或任务合理均衡地分配到各处理节点上去
 - (2) **知识分布** 根据功能分布的情况把有关知识经合理划分以后分配到各处理节点上，
 - (3) **接口设计** 各部分间接口的设计目的是要达到各部分之间互相通讯和同步容易进行，在保证完成总的任务的前提下，要尽可能使各部分之间互相独立，部分之间联系越少越好。

- (4) **系统结构** 一方面依赖于应用的环境与性质，另一方面依赖于其所处的硬件环境。
- (5) **驱动方式** 可供选择的几种驱动方式。
 - ◆ i **控制驱动**，当需要某模块工作时，就直接将控制转到该模块，或将它作为一个过程直接调用它，使它立即工作。
 - ◆ ii **数据驱动**，一般一个系统的模块功能都是根据一定的输入，启动模块进行处理以后，给出相应的输出。
 - ◆ iii **需求驱动**，这种驱动方式亦称“目的驱动”，是一种自顶向下的驱动方式。与此同时又按数据驱动的原则让数据(或其他条件)具备的模块进行工作，输出相应的结果并送到各自该去的模块。
 - ◆ iv **事件驱动**，即当且仅当模块的相应事件集合中所有事件都已发生时，才能驱动该模块开始工作。否则只要其中有一个事件尚未发生，模块就要等待，即使模块的输入数据已经全部齐备也不行。

- 一般专家系统解题的领域面很窄单个专家系统的应用局限性很大，很难获得满意的应用。协同式多专家系统是克服一般专家系统的局限性的一个重要途径。
- **协同式专家系统**亦可称“群专家系统”，表示能综合若干个相近领域的或一个领域的多个方面的子专家系统互相协作共同解决一个更广领域问题的专家系统。
- 协同式专家系统更**强调子系统之间的协同合作**，而不着重处理的分布和知识的分布。

- 设计与建立一个协同式多专家系统，需要解决的问题主要是：
 - (1) **任务的分解** 根据领域知识，将确定的总任务分解成几个分任务，分别由几个分专家系统来完成。
 - (2) **公共知识的导出** 把解决各分任务所需知识的公共部分分离出来形成一个公共知识库，供各子专家系统共享。对解决各分任务专用的知识则分别存放在各子专家系统的专用知识库中。
 - (3) **讨论方式** 目前很多作者主张采用“黑板”作为各分系统进行讨论的“园地”。为了保证在多用户环境下黑板中数据或信息的一致性，需要采用管理数据库的一些手段来管理它，使用它，因此黑板有时也称作“中间数据库”。
 - (4) **裁决问题** 这个问题的解决办法往往十分依赖于问题本身的性质。
 - (5) **驱动方式** 这个问题是与分布数据库中要考虑的相应问题一致的。尽管协同式多专家系统、各子系统可能工作在一个处理机上，但仍然有以什么方式将各子系统根据总的要求激活执行的问题，即所谓驱动方式问题。