

# Chapter 8

## First-order Logic

王子磊 (Zilei Wang)

Email: [zlwang@ustc.edu.cn](mailto:zlwang@ustc.edu.cn)

<http://vim.ustc.edu.cn>

# 提纲

- ❖ 为什么需要一阶逻辑 (FOL)?
- ❖ FOL 的语法和语义
- ❖ 运用 FOL
- ❖ 用 FOL 处理 Wumpus 世界
- ❖ FOL 中的知识工程

# 命题逻辑的优劣分析

- ☺ 命题逻辑是陈述式的 (declarative)
- ☺ 命题逻辑采用 部分(partial)/析取(disjunctive)/否定(negated)的信息表达
  - 与大部分数据结构和数据库不同
- ☺ 命题逻辑是组合的 (compositional)
  - $B_{1,1} \wedge P_{1,2}$  的含义是从  $B_{1,1}$  和  $P_{1,2}$  推导出的
- ☺ 命题逻辑中的含义是上下文独立的 (context-independent)
  - 不同于自然语言 (含义依赖于上下文信息)
- ☹ 命题逻辑的表达能力非常有限 (不像自然语言那么强大)
  - E.g., cannot say "pits cause breezes in adjacent squares" except by writing one sentence for each square

# 一阶逻辑 (First-order logic)

- ❖ 命题逻辑假设世界包含事实 (facts)
- ❖ 一阶逻辑 (FOL) 像自然语言那样假设世界包含：
  - 对象 (Objects): people, houses, numbers, colors, baseball games, wars, ...
  - 关系 (Relations): red, round, prime, brother of, bigger than, part of, comes between, ...
  - 函数 (Functions): father of, best friend, one more than, plus, ...

# Logics in general

语言	本体论	认识论
命题逻辑	facts	true/false/unknown
一阶逻辑	facts, objects, relations	true/false/unknown
时序逻辑	facts, objects, relations, times	true/false/unknown
概率论	facts	degree of belief
模糊逻辑	facts + degree of truth	known interval value

# FOL 语法：基本要素

- ❖ 常量 (Constants) : *KingJohn, 2, NUS,...*
- ❖ 谓词 (Predicates): *Brother, >,...*
- ❖ 函数 (Functions): *Sqrt, LeftLegOf,...*
- ❖ 变量 (Variables): *x, y, a, b,...*
- ❖ 连接词 (Connectives):  $\neg, \Rightarrow, \wedge, \vee, \Leftrightarrow$
- ❖ 等词 (Equality):  $=$
- ❖ 量词 (Quantifiers):  $\forall, \exists$

# 原子语句 (Atomic sentence)

Atomic sentence = *predicate* ( $term_1, \dots, term_n$ )  
or  $term_1 = term_2$

Term = *function* ( $term_1, \dots, term_n$ )  
or *constant*  
or *variable*

❖ E.g., *Brother*(*KingJohn*, *RichardTheLionheart*)

> (*Length*(*LeftLegOf*(*Richard*)), *Length*(*LeftLegOf*(*KingJohn*)))

# 复合语句 (Complex sentences)

❖ 复合语句由原子语句通过逻辑连接词构成

$$\neg S, S_1 \wedge S_2, S_1 \vee S_2, S_1 \Rightarrow S_2, S_1 \Leftrightarrow S_2$$

E.g.  $Sibling(KingJohn, Richard) \Rightarrow Sibling(Richard, KingJohn)$

$$>(1,2) \vee \leq(1,2)$$

$$>(1,2) \wedge \neg >(1,2)$$

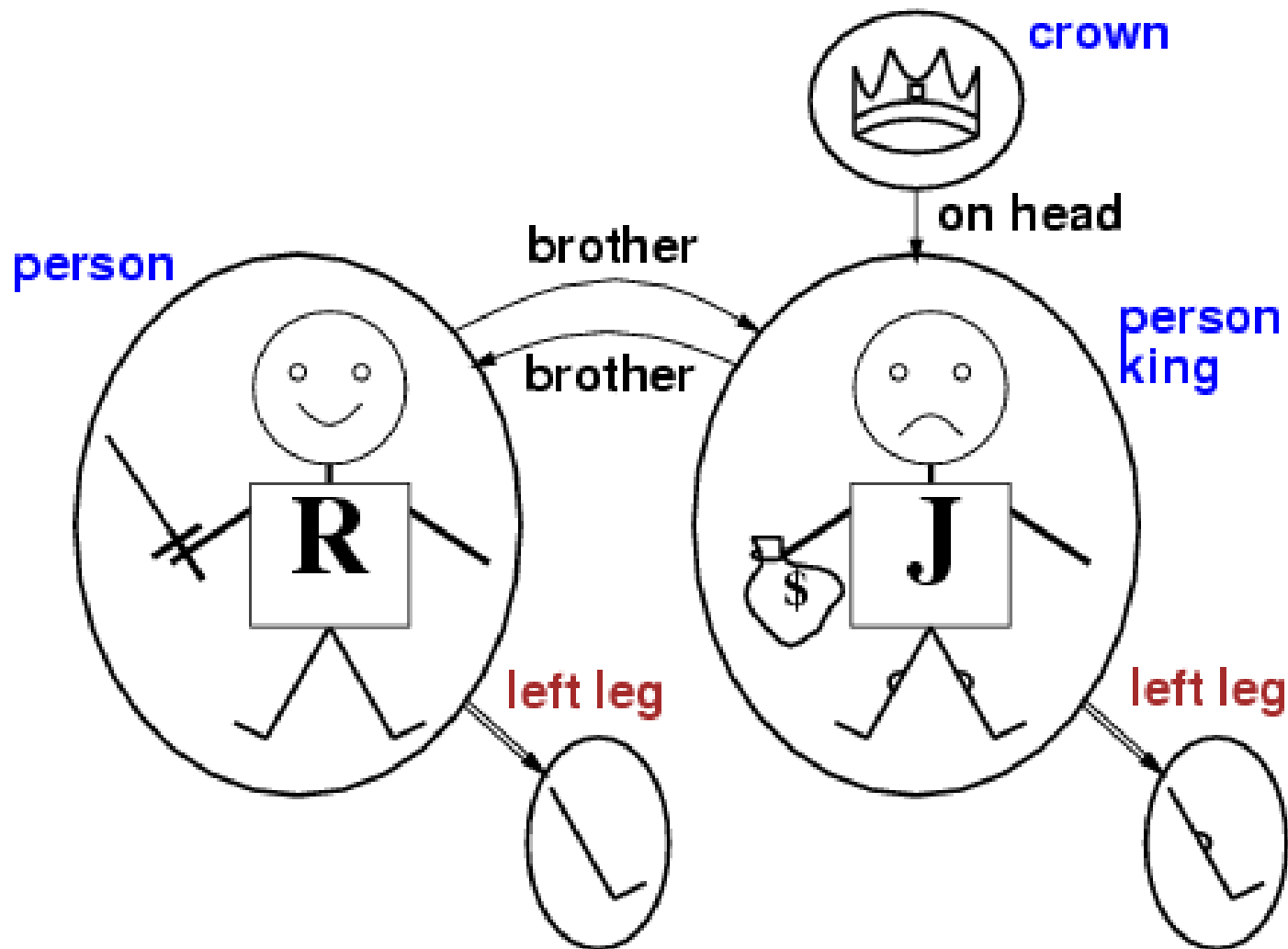


# FOL的真值

- ❖ 语句是在给定的模型和解释下为真
- ❖ 模型包含对象 (领域元素) 和它们之间的关系
- ❖ 解释指定了指示物:

constant symbols	→	objects
predicate symbols	→	relations
function symbols	→	functional relations
- ❖ 一个原子语句  $predicate(term_1, \dots, term_n)$  为真  
当且仅当  $term_1, \dots, term_n$  所表示的对象 (objects)  
具有  $predicate$  所表示的关系 (relation)

# FOL模型示例



# FOL模型

- ❖ 命题逻辑中的蕴涵能够通过模型枚举进行计算
- ❖ 对一个给定的  $KB$ ，我们也能够枚举 FOL 模型

For each number of domain elements  $n$  from 1 to  $\infty$   
For each  $k$ -ary predicate  $P_k$  in the vocabulary  
For each possible  $k$ -ary relation on  $n$  objects  
For each constant symbol  $C$  in the vocabulary  
For each choice of referent for  $C$  from  $n$  objects ...

- ❖ 通过 FOL 模型枚举进行蕴涵计算是高度复杂的！

# 全称量词 (Universal quantification)

## ❖ $\forall$ *<variables> <sentence>*

Everyone at USTC is smart:

$$\forall x \text{ At}(x, \text{USTC}) \Rightarrow \text{Smart}(x)$$

❖  $\forall x P$  在一个模型  $m$  中为真 当且仅当  $x$  被赋予  $m$  中任一可能对象时  $P$  都为真

❖ 粗略地讲，它等价于  $P$  所有实例 (instantiations) 的合取 (conjunction)

$$\text{At}(\text{KingJohn}, \text{USTC}) \Rightarrow \text{Smart}(\text{KingJohn})$$

$$\wedge \text{At}(\text{Richard}, \text{USTC}) \Rightarrow \text{Smart}(\text{Richard})$$

$$\wedge \text{At}(\text{USTC}, \text{USTC}) \Rightarrow \text{Smart}(\text{USTC})$$

$$\wedge \dots$$

# 需要避免的一个常见错误

- ❖ 典型地,  $\forall$  的主连接词是  $\Rightarrow$
- ❖ 常见错误: 用  $\wedge$  作为  $\forall$  的主连接词

$$\forall x \text{ } At(x, USTC) \wedge Smart(x)$$

表达了 “Everyone is at USTC and everyone is smart”

# 存在量词 (Existential quantification)

❖  $\exists \langle \text{variables} \rangle \langle \text{sentence} \rangle$

Someone at USTC is smart:

$$\exists x \text{ At}(x, \text{USTC}) \wedge \text{Smart}(x)$$

❖  $\exists x P$  在一个模型  $m$  中为真 当且仅当  $x$  被赋予  $m$  中某一对象时  $P$  为真

❖ 粗略地讲，它等价于  $P$  所有实例 (instantiations) 的析取 (disjunction)

$$\text{At}(\text{KingJohn}, \text{USTC}) \wedge \text{Smart}(\text{KingJohn})$$

$$\vee \text{ At}(\text{Richard}, \text{USTC}) \wedge \text{Smart}(\text{Richard})$$

$$\vee \text{ At}(\text{USTC}, \text{USTC}) \wedge \text{Smart}(\text{USTC})$$

$$\vee \dots$$

## 需要避免的另一个常见错误

- ❖ 典型地， $\exists$  的主连接词是  $\wedge$
- ❖ 常见错误：用  $\Rightarrow$  作为  $\exists$  的主连接词

$$\exists x \text{ } At(x, USTC) \Rightarrow Smart(x)$$

对于不在 USTC 的任何人将为真

# 量词特征

❖  $\forall x \forall y$  与  $\forall y \forall x$  相同

❖  $\exists x \exists y$  与  $\exists y \exists x$  相同

❖  $\exists x \forall y$  与  $\forall y \exists x$  不同

- $\exists x \forall y \text{ Loves}(x,y)$

- “There is a person who loves everyone in the world”

- $\forall y \exists x \text{ Loves}(x,y)$

- “Everyone in the world is loved by at least one person”

❖ 量词对偶性: 彼此能够相互转换

- $\forall x \text{ Likes}(x, \text{IceCream}) \quad \neg \exists x \neg \text{Likes}(x, \text{IceCream})$

- $\exists x \text{ Likes}(x, \text{Broccoli}) \quad \neg \forall x \neg \text{Likes}(x, \text{Broccoli})$



# 等词 (Equality)

❖  $term_1 = term_2$  在一个给定的解释下为真,  
当且仅当  $term_1$  和  $term_2$  指向同一对象

❖ E.g., 用 *Parent* 定义 *Sibling*:

$$\forall x, y \text{ Sibling}(x, y) \Leftrightarrow [\neg(x = y) \wedge \exists m, f \neg (m = f) \wedge \text{Parent}(m, x) \wedge \text{Parent}(f, x) \wedge \text{Parent}(m, y) \wedge \text{Parent}(f, y)]$$

# 运用 FOL

## 亲属关系论域：

### ❖ Brothers are siblings

$$\forall x,y \text{ Brother}(x,y) \Leftrightarrow \text{Sibling}(x,y)$$

### ❖ One's mother is one's female parent

$$\forall m,c \text{ Mother}(c) = m \Leftrightarrow (\text{Female}(m) \wedge \text{Parent}(m,c))$$

### ❖ “Sibling” is symmetric

$$\forall x,y \text{ Sibling}(x,y) \Leftrightarrow \text{Sibling}(y,x)$$

# 运用 FOL

## 集合论域:

- $\forall s \text{ Set}(s) \Leftrightarrow (s = \{\}) \vee (\exists x, s_2 \text{ Set}(s_2) \wedge s = \{x|s_2\})$
- $\neg \exists x, s \{x/s\} = \{\}$
- $\forall x, s \ x \in s \Leftrightarrow s = \{x/s\}$
- $\forall x, s \ x \in s \Leftrightarrow \exists y, s_2 (s = \{y/s_2\} \wedge (x = y \vee x \in s_2))$
- $\forall s_1, s_2 \ s_1 \subseteq s_2 \Leftrightarrow (\forall x \ x \in s_1 \Rightarrow x \in s_2)$
- $\forall s_1, s_2 \ (s_1 = s_2) \Leftrightarrow (s_1 \subseteq s_2 \wedge s_2 \subseteq s_1)$
- $\forall x, s_1, s_2 \ x \in (s_1 \cap s_2) \Leftrightarrow (x \in s_1 \wedge x \in s_2)$
- $\forall x, s_1, s_2 \ x \in (s_1 \cup s_2) \Leftrightarrow (x \in s_1 \vee x \in s_2)$

# 与FOL KB 交互

- ❖ 假设一个 wumpus-world agent 正在使用一个 FOL KB，并且感知到了 smell 和 breeze (but no glitter) at  $t=5$ :

$Tell(KB, Percept([Smell, Breeze, None], 5))$

$Ask(KB, \exists a \text{ BestAction}(a, 5))$

- ❖ i.e.,  $KB$  是否蕴涵了  $t=5$  的最佳行动?

- Answer:  $Yes, \{a/Shoot\}$        $\leftarrow$       substitution (binding list)

- ❖ 给定一个语句  $S$  和一个替换  $\sigma$ ， $S\sigma$  表示将  $\sigma$  插入到  $S$  的结果，e.g.,

$S = Smarter(x, y)$

$\sigma = \{x/Hillary, y/Bill\}$

$S\sigma = Smarter(Hillary, Bill)$

- ❖  $Ask(KB, S)$  返回部分/全部的  $\sigma$  使得  $KB \models S\sigma$

# Wumpus 世界的知识库

## ❖ Perception

- $\forall b, g, t \text{ Percept}([Smell, b, g], t) \Rightarrow Smell(t)$
- $\forall s, b, t \text{ Percept}([s, b, Glitter], t) \Rightarrow Glitter(t)$

## ❖ Reflex

- $\forall t \text{ Glitter}(t) \Rightarrow \text{BestAction}(\text{Grab}, t)$

## ❖ Reflex with internal state: 我们是否已经有 gold 了?

- $\forall t \text{ Glitter}(t) \wedge \neg \text{Holding}(\text{Gold}, t) \Rightarrow \text{BestAction}(\text{Grab}, t)$
- 这里,  $\text{Holding}(\text{Gold}, t)$  是不可观测的, 需要进行持续跟踪

# 推断隐命题 (hidden properties)

## ❖ 位置命题:

$$\forall x, t \text{ } At(Agent, x, t) \wedge Smelt(t) \Rightarrow Smelly(x)$$

$$\forall x, t \text{ } At(Agent, x, t) \wedge Breeze(t) \Rightarrow Breezy(x)$$

## ❖ 方格命题: Squares are breezy near a pit

- 诊断 (Diagnostic) 规则---从结果推断原因

$$\forall y \text{ } Breezy(y) \Rightarrow \exists x \text{ } Pit(x) \wedge Adjacent(x, y)$$

- 因果 (Causal) 规则---从原因推断结果

$$\forall x, y \text{ } Pit(x) \wedge Adjacent(x, y) \Rightarrow Breezy(y)$$

- 以上两个都是不完备的, 如: 因果规则没有说明远离 pit 的方格是否 *breezy*
- *Breezy* 的谓词定义为:

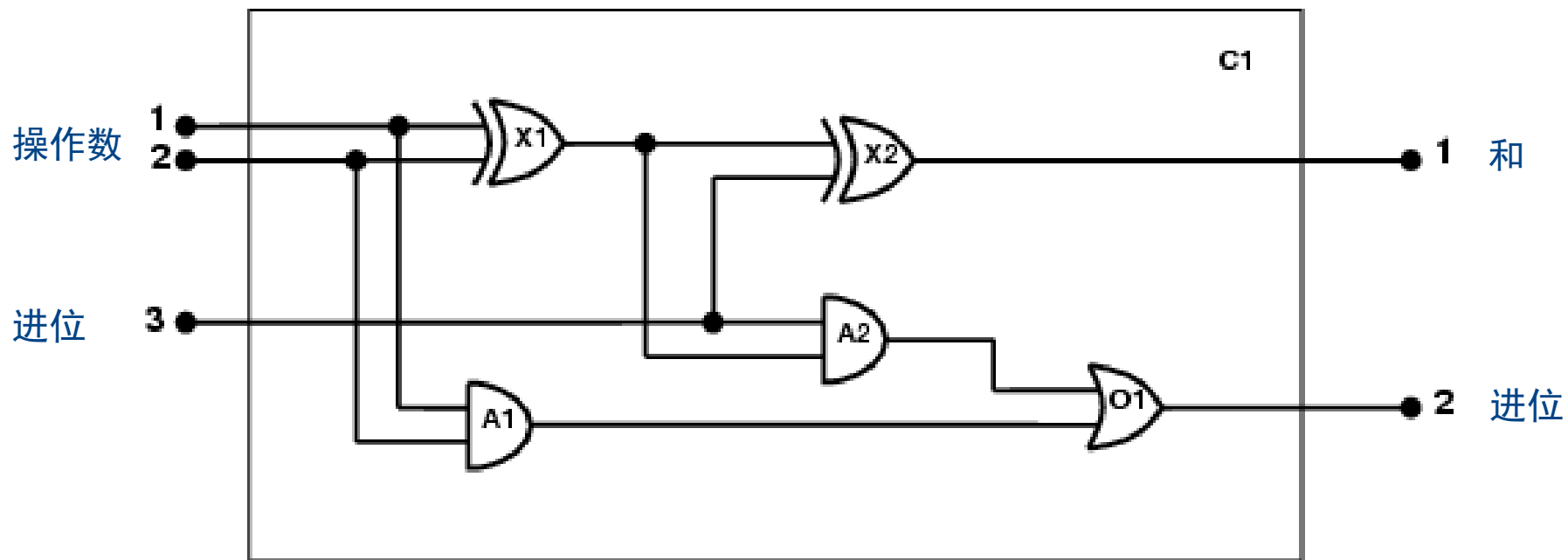
$$\forall y \text{ } Breezy(y) \Leftrightarrow [\exists x \text{ } Pit(x) \wedge Adjacent(x, y)]$$

# FOL 的知识工程

1. 确定任务
2. 搜集相关知识
3. 确定词汇表，包括谓词、函数和常量
4. 编码领域通用知识
5. 编码特定问题实例的描述
6. 将查询提交给推理过程并获得答案
7. 调试知识库

# 电路领域

## 一位全加器 (One-bit full adder)



- A —— 与门
- X —— 异或门
- O —— 或门



# 电路领域

## 1. 确定任务

- 该电路实际上能否正确的执行加法? (电路验证)

## 2. 搜集相关知识

- 由导线和门构成; 门的类型 (AND, OR, XOR, NOT)
- 不相关的: size, shape, color, cost of gates

## 3. 确定词汇表

- 一种含义的几种不同表示方法

$Type(X_1) = XOR$

$Type(X_1, XOR)$

$XOR(X_1)$

## 4. 编码领域通用知识

- $\forall t_1, t_2 \text{ Connected}(t_1, t_2) \Rightarrow \text{Signal}(t_1) = \text{Signal}(t_2)$
- $\forall t \text{ Signal}(t) = 1 \vee \text{Signal}(t) = 0$
- $1 \neq 0$
- $\forall t_1, t_2 \text{ Connected}(t_1, t_2) \Rightarrow \text{Connected}(t_2, t_1)$
- $\forall g \text{ Type}(g) = \text{OR} \Rightarrow \text{Signal}(\text{Out}(1, g)) = 1 \Leftrightarrow \exists n \text{ Signal}(\text{In}(n, g)) = 1$
- $\forall g \text{ Type}(g) = \text{AND} \Rightarrow \text{Signal}(\text{Out}(1, g)) = 0 \Leftrightarrow \exists n \text{ Signal}(\text{In}(n, g)) = 0$
- $\forall g \text{ Type}(g) = \text{XOR} \Rightarrow \text{Signal}(\text{Out}(1, g)) = 1 \Leftrightarrow \text{Signal}(\text{In}(1, g)) \neq \text{Signal}(\text{In}(2, g))$
- $\forall g \text{ Type}(g) = \text{NOT} \Rightarrow \text{Signal}(\text{Out}(1, g)) \neq \text{Signal}(\text{In}(1, g))$

## 5. 编码特定问题实例的描述

$Type(X_1) = XOR$

$Type(A_1) = AND$

$Type(O_1) = OR$

$Type(X_2) = XOR$

$Type(A_2) = AND$

$Connected(Out(1,X_1),In(1,X_2))$

$Connected(Out(1,X_1),In(2,A_2))$

$Connected(Out(1,A_2),In(1,O_1))$

$Connected(Out(1,A_1),In(2,O_1))$

$Connected(Out(1,X_2),Out(1,C_1))$

$Connected(Out(1,O_1),Out(2,C_1))$

$Connected(In(1,C_1),In(1,X_1))$

$Connected(In(1,C_1),In(1,A_1))$

$Connected(In(2,C_1),In(2,X_1))$

$Connected(In(2,C_1),In(2,A_1))$

$Connected(In(3,C_1),In(2,X_2))$

$Connected(In(3,C_1),In(1,A_2))$

## 6. 提交查询给推理过程

加法电路所有可能的终端数值集合有哪些？

$$\exists i_1, i_2, i_3, o_1, o_2 \text{ } \textit{Signal}(\textit{In}(1, C_1)) = i_1 \wedge \textit{Signal}(\textit{In}(2, C_1)) = i_2 \wedge \\ \textit{Signal}(\textit{In}(3, C_1)) = i_3 \wedge \textit{Signal}(\textit{Out}(1, C_1)) = o_1 \wedge \textit{Signal}(\textit{Out}(2, C_1)) = o_2$$

## 7. 调试知识库

- 了解知识库可能的错误行为

如：可能遗忘了  $1 \neq 0$

# 总结

## ❖ 一阶逻辑 (First-order logic, FOL):

- 对象 (objects) 和关系 (relations) 是语义原词
- 语法 (syntax): constants, functions, predicates, equality, quantifiers

## ❖ 增加了表达能力

- 足够用于定义 wumpus 世界

# 谢谢聆听！

