



# 教学安排（星期三，9:50~12:15，五教5102）

周次	日期	内容	作业	编程	项目	内容
1	9月14日	绪论				绪论。课程内容。Python
2	9月21日	无信息搜索		1		状态空间表示。宽度优先，一致代价，深度优先
3	9月28日	有信息搜索	1		1	算法复杂度分析。贪婪最佳优先搜索，A* 算法
4	10月 5日	约束满足	2			启发函数的设计，回溯算法，局部搜索
5	10月12日	对抗搜索		2		极小极大搜索，截断搜索与剪枝，蒙特卡洛树搜索
6	10月19日	命题逻辑	3			命题逻辑，演绎推理
7	10月26日	谓词逻辑	4			谓词逻辑，归结原理
8	11月 2日	回归问题	5			简单线性回归，多元回归，正则化
9	11月 9日	分类问题		3	2	Logistic 回归，Softmax 回归
10	11月16日	前馈神经网络	6			前馈神经网络。PyTorch
11	11月23日	卷积神经网络		4		卷积神经网络。TensorFlow
12	11月30日	马尔可夫决策过程	7			强化学习的数学基础
13	12月 7日	策略迭代与价值迭代		5		状态转移概率已知情况下的预测与控制
14	12月14日	蒙特卡洛与时序差分	8			状态转移概率未知情况下的预测与控制
15	12月21日	深度强化学习				价值函数的近似，深度强化学习

问题求解  
逻辑推理  
机器学习

# 人工智能基础

Fundamentals of Artificial Intelligence



# 命题逻辑 — 事实是否成立

- ▶ 假设: *The world consists of facts that either hold or do not hold*
- ▶ 优点
  - ▶ 陈述性语言: 语义从语句及模型得来
  - ▶ 有表达能力: 甚至能处理不完全信息
  - ▶ 上下文无关: 语法不依赖上下文语境
  - ▶ 语义合成性: 语义来自于语句各部分
  - ▶ 没有歧义性: 不存在似是而非的语义
- ▶ 缺点
  - ▶ 缺乏足够的表达能力, 难以简洁地描述多个事实
    - ▶ “清朝的皇帝比明朝的皇帝善于骑马” — 自然语言
    - ▶ 努尔哈赤比朱元璋善于骑马
    - ▶ 皇太极比朱元璋善于骑马
    - ▶ 福临比朱元璋善于骑马
    - ▶ ...



- ▶ 具有强大的表达能力
- ▶ 缺点

- ▶ 上下文相关：语义取决于语句的上下文
- ▶ 存在歧义性：同样的语句有不同的语义

浮海  
云水  
长朝  
长朝  
长朝  
长朝  
长朝  
长朝  
长朝  
消落

浮海  
云水  
涨潮，  
长朝  
长朝  
涨潮，  
长朝  
涨潮，  
长朝  
消落

浮海  
云水  
长朝  
涨潮，  
长朝  
长朝  
涨潮，  
长朝  
涨潮，  
长朝  
消落

浮海  
云水  
长朝  
涨潮，  
长朝  
长朝  
涨潮，  
长朝  
涨潮，  
长朝  
消落

I think **that that that that that**  
student wrote on the blackboard  
was wrong.



## 如何结合命题逻辑与自然语言的优势

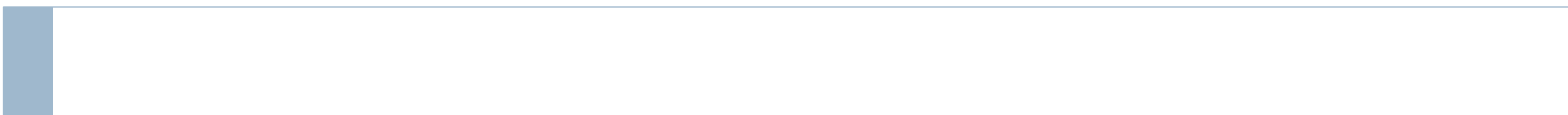
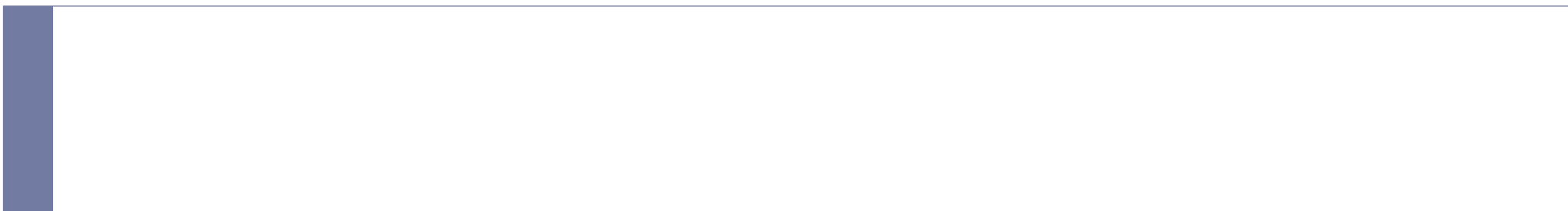
- ▶ 表达能力强
- ▶ 上下文无关
- ▶ 语义合成的
- ▶ 没有歧义的

# 一阶谓词逻辑

谓词逻辑 / Predicate logic

一阶逻辑 / First-order logic

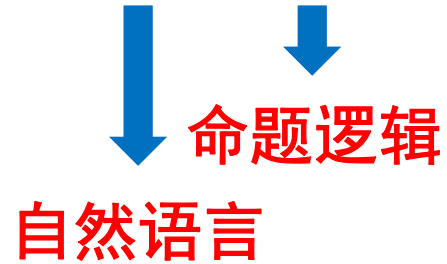
一阶谓词演算 / First-order predicate calculus



# 谓词逻辑 — 关系是否成立

- ▶ 假设: *The world consists of objects with certain relations among them that do or do not hold*

- ▶ 命题逻辑: 事实是否成立
- ▶ 谓词逻辑: 关系是否成立



- ▶ 整合命题逻辑与自然语言各自的优势
- ▶ 一阶谓词逻辑: 关系和函数本身是固定的
- ▶ 高阶谓词逻辑: 关系和函数本身也是对象

# 自然语言的关键元素

## ▶ 主谓宾

- ▶ 皇太极骑战马
- ▶ 特朗普吃牛排

## ▶ 主系表

- ▶ 煤是黑色      炭比煤更黑      这是我的头
- ▶ 我个子高      我比你更胖      这是我的手

## ▶ 名词：指代事物（对象，Object）

- ▶ 皇太极、特朗普、战马、牛排，人物、动物、数字、定理 ...

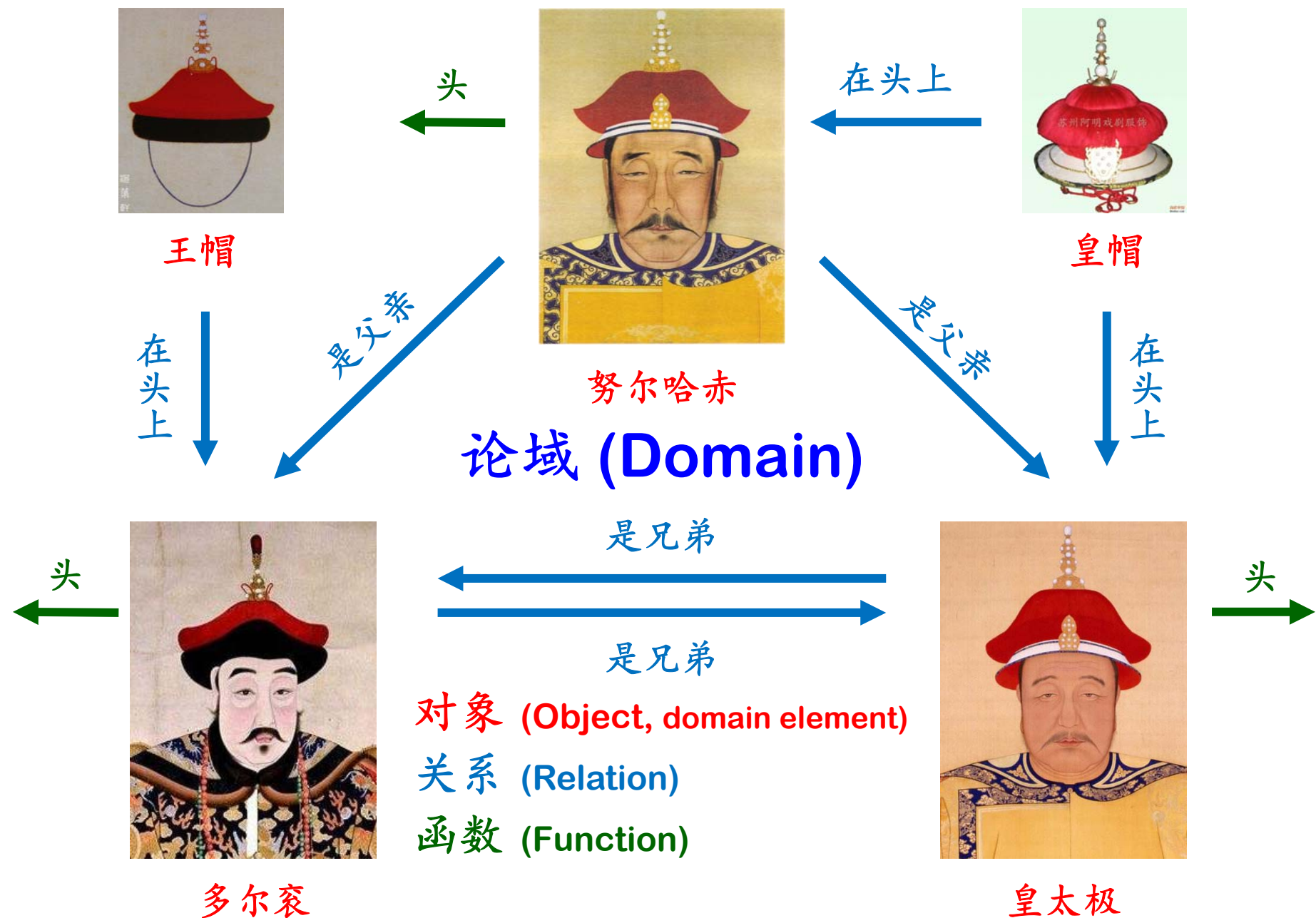
## ▶ 动词：表示对象之间的关系（Relation）

- ▶ 一元关系：颜色、高矮、胖瘦 ...      属性（Property）
- ▶ 二元关系：高于、长于、部分 ...      关系（Relation）
- ▶ 结果唯一：父亲、母亲、爷爷 ...      函数（Function）

## ▶ 数学语言

- ▶  $1 + 2 = 3$
- ▶ 对象：1, 2, 3,  $1+2$
- ▶ 关系：=
- ▶ 函数：+

# 关系





# 语法

## Grammar

--	--

--	--

- ▶ 表示论域中对象、关系、函数的字母或单词
- ▶ 常量符号：首字母大写的名词短语，努尔哈赤、皇太极、多尔袞
- ▶ 谓词符号：首字母大写的动词短语，在头上、是父亲、是兄弟
- ▶ 函数符号：首字母大写的名词短语，头、腿、手
- ▶ Constant symbols: *Nurhachi, Taiji, Dorgon ...*
- ▶ Predicate symbols: *OnHead, Father, Brother ...*
- ▶ Function symbols: *Head, Leg, Hand ...*

## ► 进行符号和对象、关系、函数之间映射的规则

- 将常量符号映射到对象
- 将谓词符号映射到关系
- 将函数符号映射到函数

- *Nurhachi*: 指代努尔哈赤
- *Taiji*: 指代皇太极
- *Dorgon*: 指代多尔袞
- *OnHead*: 指代在头上关系
- *Father*: 指代是父亲关系
- *Brother*: 指代是兄弟关系
- *Head*: 指代头函数
- *Hand*: 指代手函数
- *Leg*: 指代腿函数

- ▶ 指代对象的表达式称为项 (**Term**)
- ▶ 语法
  - ▶ 常量符号: *Nurhachi*, *Taiji*, *Dorgon*
  - ▶ 函数符号作用的结果: *Head(Nurhachi)* — 努尔哈赤的头  
函数的结果是唯一的, 因此不用再为 *Head of Nurhachi* 命名  
可以认为 ***Head(Nurhachi)*** 就是一个对象的名称
- ▶ 语义
  - ▶ 函数作用到对象, 得到另一个对象  
*LeftLeg(Taiji)* — 皇太极的左腿  
*RightHand(Gorgon)* — 多尔袞的右手

## ▶ 谓词符号作用于项的结果称为原子语句

### ▶ 语法

- ▶ *Emperor(Taiji)* — 皇太极是皇帝
- ▶ *Prince(Gorgon)* — 多尔衮是王爷
- ▶ *Brother(Taiji, Gorgon)* — 多尔衮是皇太极的兄弟
- ▶ *Married(Father(Taiji), Mother(Gorgon))* — 皇太极的父亲和多尔衮的母亲结婚了

### ▶ 语义

#### ▶ 原子语句是逻辑表达式

- ▶ 如果谓词所指代的关系在参数所指代的对象中成立，那么原子语句在给定模型、给定解释下为真

## ▶ 由逻辑连接符相连的语句

### ▶ 语法：同命题逻辑

- ▶  $\neg \text{Emperor}(\text{Gorgon})$  — 多尔袞不是皇帝
- ▶  $\text{Father}(\text{Taiji}) \wedge \text{Mother}(\text{Gorgon})$  — 皇太极的父亲与多尔袞的母亲
- ▶  $\text{Head}(\text{Taiji}) \vee \text{Head}(\text{Gorgon})$  — 皇太极的头或多尔袞的头
- ▶  $\text{Emperor}(\text{Taiji}) \Rightarrow \text{Prince}(\text{Gorgon})$  — 皇太极是皇帝隐含多尔袞是王爷
- ▶  $\text{Father}(\text{Taiji}) \Leftrightarrow \text{Father}(\text{Gorgon})$  — 皇太极的和多尔袞的父亲等价

### ▶ 语义：同命题逻辑

$P$	$Q$	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE

# 全称量词 ( $\forall$ )

# Universal quantifier

- 谓词逻辑通过**全称量词**指代论域中的所有对象

$$\forall x \text{ Emperor}(x) \Rightarrow \text{Person}(x)$$

- For all  $x$ , if  $x$  is an emperor, then  $x$  is a person
- 对于所有  $x$ , 如果  $x$  是皇帝, 那么  $x$  是人 (所有皇帝都是人)
- $x$ : **变量 (variable)**, 指代对象, 是一种项



# 全称量词隐含了合取关系

- ▶  $\forall x$ , 论域中的任意对象
  - ▶ 努尔哈赤、皇太极、多尔袞
  - ▶ 皇帽、王帽、努尔哈赤的头、皇太极的头、多尔袞的头 ...
- ▶ 扩展解释  $\forall x \text{ Emperor}(x) \Rightarrow \text{Person}(x)$ 
  - ▶  $\text{Emperor}(\text{Nurhachi}) \Rightarrow \text{Person}(\text{Nurhachi})$
  - ▶  $\text{Emperor}(\text{Taiji}) \Rightarrow \text{Person}(\text{Taiji})$
  - ▶  $\text{Emperor}(\text{Dorgon}) \Rightarrow \text{Person}(\text{Dorgon})$
  - ▶  $\text{Emperor}(\text{皇帽}) \Rightarrow \text{Person}(\text{皇帽})$
  - ▶  $\text{Emperor}(\text{Head}(\text{Nurhachi})) \Rightarrow \text{Person}(\text{Head}(\text{Nurhachi}))$
  - ▶ ... 所有这些语句是合取的关系
- ▶ 隐含关系什么时候为真？什么时候为假？
- ▶  $\forall x$ , 只用考虑前项为真的隐含关系
- ▶ 使用全称量词  $\forall$  时, 隐含关系  $\Rightarrow$  是完美的选择



# 存在量词 (Existential quantifier, $\exists$ )

- 谓词逻辑通过存在量词指代论域中的某个或某些对象

$$\exists x \text{ Crown}(x) \wedge \text{OnHead}(x, \text{Taiji})$$

- There exists an  $x$  such that  $x$  is a crown and  $x$  is on the head of Taiji
- 存在某个  $x$ , 它是皇帽并且戴在皇太极的头上  
(皇太极戴着皇帽)



# 存在量词隐含了析取关系

- ▶  $\exists x$ , 论域中的某个或某些对象（至少一个, at least）
  - ▶ 努尔哈赤、皇太极、多尔袞、皇帽、王帽、努尔哈赤的头、皇太极的头、多尔袞的头 ...
  - ▶ 其中有至少一个（一个或者多个）
- ▶ 扩展解释  $\exists x \text{ Crown}(x) \wedge \text{OnHead}(x, \text{Taiji})$ 
  - ▶  $\text{Crown}(\text{Nurhachi}) \wedge \text{OnHead}(\text{Nurhachi}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Taiji}) \wedge \text{OnHead}(\text{Taiji}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Dorgon}) \wedge \text{OnHead}(\text{Dorgon}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{皇帽}) \wedge \text{OnHead}(\text{皇帽}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Head}(\text{Nurhachi})) \wedge \text{OnHead}(\text{Head}(\text{Nurhachi}), \text{Taiji})$
  - ▶ ... ... 所有这些语句是析取的关系
- ▶ 只要这些语句有一个是真就行
- ▶ 使用存在量词  $\exists$  时, 合取关系  $\wedge$  是完美的选择

# $\forall \Rightarrow$ 在一起, $\exists \wedge$ 在一起

- ▶ 考虑  $\forall x \text{ Emperor}(x) \wedge \text{Person}(x)$ 
  - ▶  $\text{Emperor}(\text{Nurhachi}) \wedge \text{Person}(\text{Nurhachi})$
  - ▶  $\text{Emperor}(\text{Taiji}) \wedge \text{Person}(\text{Taiji})$
  - ▶  $\text{Emperor}(\text{Dorgon}) \wedge \text{Person}(\text{Dorgon})$
  - ▶  $\text{Emperor}(\text{皇帽}) \wedge \text{Person}(\text{皇帽})$
  - ▶  $\text{Emperor}(\text{Head}(\text{Nurhachi})) \wedge \text{Person}(\text{Head}(\text{Nurhachi}))$
- ▶ 合取项有一个为假, 语句为假, 结果是?

**上面的子句几乎必然会出现为假的!**

- ▶  $\forall$  作为限定词的情况下, 使用  $\wedge$  会使陈述过强

# $\forall \Rightarrow$ 在一起, $\exists \wedge$ 在一起

- ▶ 考虑  $\exists x \text{ Crown}(x) \Rightarrow \text{OnHead}(x, \text{Taiji})$ 
  - ▶  $\text{Crown}(\text{Nurhachi}) \Rightarrow \text{OnHead}(\text{Nurhachi}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Taiji}) \Rightarrow \text{OnHead}(\text{Taiji}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Dogon}) \Rightarrow \text{OnHead}(\text{Dogon}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{皇帽}) \Rightarrow \text{OnHead}(\text{皇帽}, \text{Taiji})$
  - ▶  $\text{Crown}(\text{Head}(\text{Nurhachi})) \Rightarrow \text{OnHead}(\text{Head}(\text{Nurhachi}), \text{Taiji})$
- ▶ 前项为假时, 隐含式为真, 结果是?

**上面的子句全都是真!**

- ▶  $\exists$  作为限定词的情况下, 使用  $\Rightarrow$  会使陈述过弱

# 量词嵌套

- 多个量词可以嵌套起来，表示更复杂的语句

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

- 同一种类型的多个量词可以写到一起

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

- 混合量词

$\forall x \exists y \text{ Loves}(x, y)$	每个人都会爱上某人
$\exists x \forall y \text{ Loves}(x, y)$	存在某个人爱所有人
$\exists y \forall x \text{ Loves}(x, y)$	存在某人被所有人爱
$\forall x (\exists y \text{ Loves}(x, y))$	每个人都会爱上某人

- 变量属于引用它的最内层量词，且不再属于其他任何量词**

$$\forall x (\text{Crown}(x) \vee (\exists x \text{ Brother}(\text{Taiji}, x)))$$

- 嵌套时使用不同的变量

$$\forall x (\text{Crown}(x) \vee (\exists y \text{ Brother}(\text{Taiji}, y)))$$

# $\forall$ 和 $\exists$ 的互反性

## ▶ 互反性

$$\forall x Likes(x, IceCream) \equiv \neg \exists x \neg Likes(x, IceCream)$$

每个人都喜欢吃冰淇淋  $\equiv$  没有人不喜欢吃冰淇淋

$$\begin{aligned} \forall x Likes(x, IceCream) & \quad \text{每个人都喜欢吃冰淇淋} \\ & \equiv Likes(a, IceCream) \wedge Likes(b, IceCream) \wedge Likes(c, IceCream) \wedge \dots \end{aligned}$$

$$\begin{aligned} \neg(\forall x Likes(x, IceCream)) & \quad \text{不是每个人都喜欢吃冰淇淋} \\ & \equiv \neg(Likes(a, IceCream) \wedge Likes(b, IceCream) \wedge Likes(c, IceCream) \wedge \dots) \\ & \equiv \neg Likes(a, IceCream) \vee \neg Likes(b, IceCream) \vee \neg Likes(c, IceCream) \vee \dots \\ & \equiv \exists x \neg Likes(x, IceCream) \quad \text{有的人不喜欢吃冰淇淋} \end{aligned}$$

$$\begin{aligned} \forall x Likes(x, IceCream) & \quad \text{每个人都喜欢吃冰淇淋} \\ & \equiv \neg(\neg(\forall x Likes(x, IceCream))) \quad \text{不是不是每个人都喜欢吃冰淇淋} \\ & \equiv \neg(\exists x \neg Likes(x, IceCream)) \quad \text{不是有的人不喜欢吃冰淇淋} \\ & \equiv \neg \exists x \neg Likes(x, IceCream) \quad \text{没有人不喜欢吃冰淇淋} \end{aligned}$$

# $\forall$ 和 $\exists$ 的互反性与摩根律

## ▶ 互反性

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

## ▶ 摩根律

$$\forall x P(x) \equiv \neg \exists x \neg P(x)$$

$$\exists x P(x) \equiv \neg \forall x \neg P(x)$$

$$\forall x \neg P(x) \equiv \neg \exists x P(x)$$

$$\neg \forall x P(x) \equiv \exists x \neg P(x)$$

$$\text{▶ } \forall x \neg P(x) \equiv \neg P(a) \wedge \neg P(b) \wedge \neg P(c) \dots \dots$$

$$\text{▶ } \neg(\forall x \neg P(x)) \equiv \neg(\neg P(a) \wedge \neg P(b) \wedge \neg P(c) \dots \dots)$$

$$\text{▶ } \equiv \neg(\neg P(a)) \vee \neg(\neg P(b)) \vee \neg(\neg P(c)) \dots \dots$$

$$\text{▶ } \exists x P(x) \equiv P(a) \vee P(b) \vee P(c) \dots \dots$$

$$\text{▶ } \forall x \neg P(x) \equiv \neg(\neg(\forall x \neg P(x))) \equiv \neg(\exists x P(x)) \equiv \neg \exists x P(x)$$

## ▶ 声明两个项指代同一个对象

$$Father(Taiji) = Nurhachi$$

## ▶ 皇太极至少有两个兄弟

- ▶  $x$  和  $y$  可以是同一个人

$$\exists x, y \text{ Brother}(Taiji, x) \wedge \text{Brother}(Taiji, y)$$

- ▶  $x$  和  $y$  不能是同一个人

$$\exists x, y \text{ Brother}(Taiji, x) \wedge \text{Brother}(Taiji, y) \wedge \neg(x=y)$$

$$\exists x, y \text{ Brother}(Taiji, x) \wedge \text{Brother}(Taiji, y) \wedge x \neq y$$



# 自然语言的形式化

- ▶ 谓词逻辑能够较好地描述自然语言

- ▶ 没有共产党就没有新中国

$$\forall x \neg CommunistParty(x) \Rightarrow \neg BuildNewChina(x)$$

$$\forall x BuildNewChina(x) \Rightarrow CommunistParty(x)$$

- ▶ 只有共产党才能救中国

$$\forall x \neg CommunistParty(x) \Rightarrow \neg SaveChina(x)$$

$$\forall x SaveChina(x) \Rightarrow CommunistParty(x)$$

- ▶ 四项基本原则是立国之本

$$\forall x FourCardinalPrinciples(x) \Rightarrow FoundationOfCountry(x)$$

$$\forall x \neg FoundationOfCountry(x) \Rightarrow \neg FourCardinalPrinciples(x)$$

- ▶ 我们要坚持思政教育

$$\forall x PoliticalEducation(x) \Rightarrow WeInsist(x)$$

# 自然语言的形式化

- ▶ 大家要好好学习的天天向上

$$\forall x \text{ StudyHard}(x) \wedge (\forall y \text{ MakeProgress}(x,y))$$

- ▶ 要玩得好也要学得好

$$\forall x \text{ PlayWell}(x) \wedge \text{StudyHard}(x)$$

- ▶ 双十一的东西又好又便宜

$$\forall x \text{ SaledInDoubleEleven}(x) \Rightarrow (\text{Good}(x) \wedge \text{Cheap}(x))$$

- ▶ 双十一的东西也有质量不好还挺贵的

$$\exists x \text{ SaledInDoubleEleven}(x) \wedge \text{LowQuality}(x) \wedge \text{Expensive}(x)$$

- ▶ 我们班有不用怎么学习就考高分的人

$$\exists x \text{ InOurClass}(x) \wedge \text{NotStudyMuch}(x) \wedge \text{ReceiveHighScore}(x)$$

# 谓词逻辑还能描述其他语言

- ▶ 所有有理数都是实数
  - ▶  $\forall x \text{RatNum}(x) \Rightarrow \text{RealNum}(x)$
  - ▶ *RatNum*: 谓词, 对有理数为真
  - ▶ *RealNum*: 谓词, 对实数为真
- ▶ 有的实数是有理数
  - ▶  $\exists x \text{RealNum}(x) \wedge \text{RatNum}(x)$
- ▶ 没有无理数是有理数
  - ▶  $\forall x \text{IrrNum}(x) \Rightarrow \neg \text{RatNum}(x)$
  - ▶  $\forall x \text{RatNum}(x) \Rightarrow \neg \text{IrrNum}(x)$
  - ▶  $\neg \exists x \text{RatNum}(x) \wedge \text{IrrNum}(x)$
  - ▶ *IrrNum*: 谓词, 无理数为真

# 定义自然数

- ▶  $NatNum$ : a predicate that will be true of natural numbers
- ▶  $0$ : a constant symbol
- ▶  $S$ : a function symbol, meaning successor

## 定义

1.  $NatNum(0)$
2.  $\forall n \text{ } NatNum(n) \Rightarrow NatNum(S(n))$
3.  $\forall n \text{ } 0 \neq S(n)$
4.  $\forall m, n \text{ } m \neq n \Rightarrow S(m) \neq S(n)$

Prefix:  $+(m, n)$

Infix:  $m+n$

$S(n)$ :  $n+1$

## 加法

1.  $\forall m \text{ } NatNum(m) \Rightarrow +(0, m) = m$
2.  $\forall m, n \text{ } NatNum(m) \wedge NatNum(n) \Rightarrow +(S(m), n) = S(+(m, n))$

# 语义

## Semantics

--	--

--	--

## 基本原理：命题逻辑的等值演算 + 对量词的处理

### 变量换名

$$\forall x P(x, \dots) \equiv \forall y P(y, \dots)$$

$$\exists x P(x, \dots) \equiv \exists y P(y, \dots)$$

### 量词消去

$$\forall x P(x) \equiv P(a) \wedge P(b) \wedge P(c) \wedge \dots$$

$$\exists x P(x) \equiv P(a) \vee P(b) \vee P(c) \vee \dots$$

### 量词否定

$$\forall x P(x) \equiv \neg \exists x \neg P(x)$$

$$\exists x P(x) \equiv \neg \forall x \neg P(x)$$

$$\forall x \neg P(x) \equiv \neg \exists x P(x)$$

$$\neg \forall x P(x) \equiv \exists x \neg P(x)$$

### ▶ 全称量词

$$\forall x (P(x) \wedge Q) \equiv (\forall x P(x)) \wedge Q$$

$$\forall x (P(x) \vee Q) \equiv (\forall x P(x)) \vee Q$$

$$\forall x (P(x) \Rightarrow Q) \equiv (\exists x P(x)) \Rightarrow Q$$

$$\forall x (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\forall x P(x))$$

$$\begin{aligned} \forall x (P(x) \wedge Q) &\equiv (P(a) \wedge Q) \wedge (P(b) \wedge Q) \wedge (P(c) \wedge Q) \wedge \dots \\ &\equiv (P(a) \wedge P(b) \wedge P(c) \wedge \dots) \wedge Q \\ &\equiv (\forall x P(x)) \wedge Q \end{aligned}$$

$$\begin{aligned} \forall x (P(x) \vee Q) &\equiv (P(a) \vee Q) \wedge (P(b) \vee Q) \wedge (P(c) \vee Q) \wedge \dots \\ &\equiv ((P(a) \wedge P(b)) \vee Q) \wedge (P(c) \vee Q) \wedge \dots \\ &\equiv ((P(a) \wedge P(b) \wedge P(c)) \vee Q) \wedge \dots \\ &\equiv (P(a) \wedge P(b) \wedge P(c) \wedge \dots) \vee Q \\ &\equiv (\forall x P(x)) \vee Q \end{aligned}$$

### ▶ 全称量词

$$\forall x (P(x) \wedge Q) \equiv (\forall x P(x)) \wedge Q$$

$$\forall x (P(x) \vee Q) \equiv (\forall x P(x)) \vee Q$$

$$\forall x (P(x) \Rightarrow Q) \equiv (\exists x P(x)) \Rightarrow Q$$

$$\forall x (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\forall x P(x))$$

$$\begin{aligned}\forall x (P(x) \Rightarrow Q) &\equiv \forall x (\neg P(x) \vee Q) \\ &\equiv \forall x (\neg P(x)) \vee Q \\ &\equiv \neg(\forall x (\neg P(x))) \Rightarrow Q \\ &\equiv \exists x P(x) \Rightarrow Q\end{aligned}$$

$$\begin{aligned}\forall x (Q \Rightarrow P(x)) &\equiv \forall x (\neg Q \vee P(x)) \\ &\equiv \forall x (P(x) \vee \neg Q) \\ &\equiv \forall x (P(x)) \vee \neg Q \\ &\equiv \neg Q \vee (\forall x P(x)) \\ &\equiv Q \Rightarrow (\forall x P(x))\end{aligned}$$



# 存在量词辖域收缩与扩张

$Q$  里不能出现  $x$

## ▶ 全称量词

$$\forall x (P(x) \wedge Q) \equiv (\forall x P(x)) \wedge Q$$

$$\forall x (P(x) \vee Q) \equiv (\forall x P(x)) \vee Q$$

$$\forall x (P(x) \Rightarrow Q) \equiv (\exists x P(x)) \Rightarrow Q$$

$$\forall x (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\forall x P(x))$$

## ▶ 存在量词

$$\exists x (P(x) \wedge Q) \equiv (\exists x P(x)) \wedge Q$$

$$\exists x (P(x) \vee Q) \equiv (\exists x P(x)) \vee Q$$

$$\exists x (P(x) \Rightarrow Q) \equiv (\forall x P(x)) \Rightarrow Q$$

$$\exists x (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\exists x P(x))$$

合取析取，直接扩张  
隐含后项，直接扩张  
隐含前项，量词变换

同理可证

# 全称量词的分配

- ▶ 全称量词可以对合取分配，不能对析取分配

$$\forall x (P(x) \wedge Q(x)) \equiv \forall x P(x) \wedge \forall x Q(x)$$

$$\forall x (P(x) \vee Q(x)) \equiv \forall x P(x) \vee \forall x Q(x) \text{ 错!}$$

$$\begin{aligned} \forall x (P(x) \wedge Q(x)) &\equiv (P(a) \wedge Q(a)) \wedge (P(b) \wedge Q(b)) \wedge (P(c) \wedge Q(c)) \wedge \dots \\ &\equiv (P(a) \wedge P(b) \wedge P(c) \wedge \dots) \wedge (Q(a) \wedge Q(b) \wedge Q(c) \wedge \dots) \\ &\equiv (\forall x P(x)) \wedge (\forall x Q(x)) \end{aligned}$$

- ▶ 反例

- ▶  $\forall x (P(x) \vee Q(x))$

每一名自动化系的同学，要么学习成绩好，要么社会工作好

- ▶  $\forall x P(x) \vee \forall x Q(x)$

每一名自动化系的同学都学习成绩好，或者  
每一名自动化系的同学都社会工作好

# 存在称量词的分配

- 存在量词可以对析取分配，不能对合取分配

$$\exists x (P(x) \vee Q(x)) \equiv \exists x P(x) \vee \exists x Q(x)$$

$$\exists x (P(x) \wedge Q(x)) \equiv \exists x P(x) \wedge \exists x Q(x) \text{ 错!}$$

$$\begin{aligned} \exists x (P(x) \vee Q(x)) &\equiv (P(a) \vee Q(a)) \vee (P(b) \vee Q(b)) \vee (P(c) \vee Q(c)) \vee \dots \\ &\equiv (P(a) \vee P(b) \vee P(c) \vee \dots) \vee (Q(a) \vee Q(b) \vee Q(c) \vee \dots) \\ &\equiv (\exists x P(x)) \vee (\exists x Q(x)) \end{aligned}$$

- 反例

- $\exists x (P(x) \wedge Q(x))$

自动化系有一些同学，学习成绩好并且社会工作好

- $\exists x P(x) \wedge \exists x Q(x)$

自动化系有一些同学学习成绩好，并且  
自动化系有一些同学社会工作好

# 量词分配

- ▶ 全称量词可以对合取分配，不能对析取分配
- ▶ 存在量词可以对析取分配，不能对合取分配

$$\forall x (P(x) \wedge Q(x)) \equiv \forall x P(x) \wedge \forall x Q(x)$$

$$\exists x (P(x) \vee Q(x)) \equiv \exists x P(x) \vee \exists x Q(x)$$

$$\forall x (P(x) \vee Q(x)) \equiv \forall x P(x) \vee \forall x Q(x) \text{ 错!}$$

$$\exists x (P(x) \wedge Q(x)) \equiv \exists x P(x) \wedge \exists x Q(x) \text{ 错!}$$

$$\begin{aligned}
 & (\forall x P(x, y) \Rightarrow \exists y Q(y)) \Rightarrow \forall x R(x, y) \\
 \equiv & (\forall x P(x, w) \Rightarrow \exists y Q(y)) \Rightarrow \forall z R(z, w) \\
 \equiv & \exists x (P(x, w) \Rightarrow \exists y Q(y)) \Rightarrow \forall z R(z, w) \\
 \equiv & \forall x ((P(x, w) \Rightarrow \exists y Q(y)) \Rightarrow \forall z R(z, w)) \\
 \equiv & \forall x (\exists y (P(x, w) \Rightarrow Q(y)) \Rightarrow \forall z R(z, w)) \\
 \equiv & \forall x (\forall y ((P(x, w) \Rightarrow Q(y)) \Rightarrow \forall z R(z, w))) \\
 \equiv & \forall x \forall y \forall z ((P(x, w) \Rightarrow Q(y)) \Rightarrow R(z, w))
 \end{aligned}$$

按命题逻辑处理

变量换名

量词辖域扩张

量词辖域扩张

量词辖域扩张

量词辖域扩张

量词辖域扩张

$$\begin{aligned}
 \forall x (P(x) \wedge Q) & \equiv (\forall x P(x)) \wedge Q \\
 \forall x (P(x) \vee Q) & \equiv (\forall x P(x)) \vee Q \\
 \forall x (P(x) \Rightarrow Q) & \equiv (\exists x P(x)) \Rightarrow Q \\
 \forall x (Q \Rightarrow P(x)) & \equiv Q \Rightarrow (\forall x P(x)) \\
 \exists x (P(x) \wedge Q) & \equiv (\exists x P(x)) \wedge Q \\
 \exists x (P(x) \vee Q) & \equiv (\exists x P(x)) \vee Q \\
 \exists x (P(x) \Rightarrow Q) & \equiv (\forall x P(x)) \Rightarrow Q \\
 \exists x (Q \Rightarrow P(x)) & \equiv Q \Rightarrow (\exists x P(x))
 \end{aligned}$$

有什么特点?

# 前束范式

- ▶ 语句中所有量词（不含否定词）都在最左边，并且这些量词的辖域都到公式的最末端
- ▶ 一阶逻辑中任何语句均可转换为与之等值的前束范式
- ▶ 前束范式不唯一

$$(\forall x P(x, y) \Rightarrow \exists y Q(y)) \Rightarrow \forall x R(x, y) \equiv$$

$$\forall x \forall y \forall z ((P(x, w) \Rightarrow Q(y)) \Rightarrow R(z, w))$$

前束范式

前面的全称量词存在量词怎么办？

# 全称量词的实例化

- ▶ 全称量词

$$\forall x \text{ Emperor}(x) \Rightarrow \text{CanFight}(x)$$

- ▶ 全称量词指代论域中的任意对象，因此
- ▶ 将变量替换成论域中的任意对象，逻辑表达式都是成立的
- ▶ 这种变量的替换称为**置换**
  - ▶ 置换： $\theta = \{x/\text{Taiji}\}$  把  $x$  换成  $\text{Taiji}$ ，有的书写为  $\theta = \{\text{Taiji}/x\}$
  - ▶ 形式： $\theta = \{v/g\}$   $v$ : variable,  $g$ : ground term
- ▶ 既然置换成任意对象都可以，
- ▶ **直接消去全称量词，待需要替换时再考虑置换为哪个对象**

# 存在量词的实例化

## 存在量词

$$\exists y \text{ Crown}(y) \wedge \text{OnHead}(y, \text{Nurhachi})$$

- 存在量词指代论域中的某个或某些对象，因此
- 将变量替换成论域中的某个具体的对象，逻辑表达式会成立
- 给这个特定对象起名字  $C_N$ ，这种起名也是一种**置换**
  - 置换:  $\theta = \{y/C_N\}$  给  $x$  起名  $C_N$ ,  $C_N$  不能在其他语句中出现
  - 形式:  $\theta = \{v/c\}$   $v$ : variable,  $c$ : Skolem constant
- 但是，如果该特定**对象被全称量词所管辖**

$$\forall x \exists y \text{ Crown}(y) \wedge \text{OnHead}(y, x)$$

- 则该对象的起名依赖于管辖它的变量，此时**置换应为函数**
  - 置换:  $\theta = \{y/F(x)\}$  给  $x$  起名  $F(x)$ ,  $F$  不能在其他语句中出现
  - 形式:  $\theta = \{v/c\}$   $v$ : variable,  $F(x)$ : Skolem function



# 合取范式

- ▶ 在前束范式的基础上，消去量词，转化为析取子句的合取形式
- ▶ 获取合取范式的步骤：
  1. 等价消去
  2. 隐含消去
  3. 否定深入
  4. 变量换名
  5. 量词前移（量词辖域扩张）
  6. 量词消去（先消去存在量词，再消去全称量词）
  7. 分配律

$$\forall x [(\forall y \text{ Person}(y) \Rightarrow \text{Loves}(x, y)) \Rightarrow (\exists y \text{ Loves}(y, x))]$$

▶ 等价消去:  $P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

▶ 隐含消去:  $P \Rightarrow Q \equiv \neg P \vee Q$

$$\forall x [\neg(\forall y \neg \text{Person}(y) \vee \text{Loves}(x, y)) \vee (\exists y \text{ Loves}(y, x))]$$

▶ 否定深入:  $\neg(\forall y P) \equiv \exists y \neg P$ ,  $\neg(\exists y P) \equiv \forall y \neg P$

$$\forall x [(\exists y \text{ Person}(y) \wedge \neg \text{Loves}(x, y)) \vee (\exists y \text{ Loves}(y, x))]$$

▶ 变量换名:

$$\forall x [(\exists y \text{ Person}(y) \wedge \neg \text{Loves}(x, y)) \vee (\exists z \text{ Loves}(z, x))]$$

▶ 消去存在量词: 斯柯林化

$$\forall x [(\text{Person}(F(x)) \wedge \neg \text{Loves}(x, F(x))) \vee \text{Loves}(G(x), x)]$$

▶ 消去全称量词: 直接消去

$$(\text{Person}(F(x)) \wedge \neg \text{Loves}(x, F(x))) \vee \text{Loves}(G(x), x)$$

▶ 分配律:

$$(\text{Person}(F(x)) \vee \text{Loves}(G(x), x)) \wedge (\neg \text{Loves}(x, F(x)) \vee \text{Loves}(G(x), x))$$

# 演绎推理

Deduction

--	--

--	--

# 前向链接 (Forward chaining)

正向推理

正向演绎

Forward reasoning

- 在限定子句上使用假言推理，从事实出发，匹配规则，产生事实，直到结论

事实

能见度差( $A$ ) 空气刺鼻( $B$ ) 咽喉干涩( $C$ ) 处于室内( $D$ )

规则

处于室外( $E$ ) 是雾霾天( $F$ ) 进行防护( $G$ ) 开净化器( $H$ ) 戴上口罩( $I$ )

如果 能见度差 并且 空气刺鼻 并且 咽喉干涩 那么 是雾霾天

$$A \wedge B \wedge C \Rightarrow F$$

如果 是雾霾天 那么 进行防护

$$F \Rightarrow G$$

如果 进行防护 并且 处于室内 那么 开净化器

$$G \wedge D \Rightarrow H$$

如果 进行防护 并且 处于室外 那么 戴上口罩

$$G \wedge E \Rightarrow I$$

推理（反复应用假言推理）

$$A \wedge B \wedge C \Rightarrow F$$

$$F \Rightarrow G$$

$G \wedge D \Rightarrow H$  应该开净化器

# 后向链接 (Backward chaining)

反向推理  
逆向演绎

Backward reasoning

- 在限定子句上使用假言推理，从查询出发，匹配规则，倒推事实，直到满足

事实

能见度差( $A$ ) 空气刺鼻( $B$ ) 咽喉干涩( $C$ ) 处于室内( $D$ )

查询

是否应该 开净化器 ( $H$ )

规则

处于室外( $E$ ) 是雾霾天( $F$ ) 进行防护( $G$ ) 开净化器 ( $H$ ) 带上口罩( $I$ )

如果 进行防护 并且 处于室内 那么 开净化器

如果 是雾霾天 那么 进行防护

如果 能见度差 并且 空气刺鼻 并且 咽喉干涩 那么 是雾霾天

如果 进行防护 并且 处于室外 那么 带上口罩

$$G \wedge D \Rightarrow H$$

$$F \Rightarrow G$$

$$A \wedge B \wedge C \Rightarrow F$$

$$G \wedge E \Rightarrow I$$

推理（反复应用假言推理）

$$G \wedge D \Rightarrow H$$

应该开净化器

$$F \Rightarrow G$$

$$A \wedge B \wedge C \Rightarrow F$$

# 谓词逻辑的推理

- ▶ 清朝皇帝能打仗，皇太极是清朝皇帝，所以皇太极能打仗

$$\forall x \text{Emperor}(x) \Rightarrow \text{CanFight}(x) \quad \text{Emperor}(\text{Taiji}) \quad \text{CanFight}(\text{Taiji})$$

$$((\forall x \text{Emperor}(x) \Rightarrow \text{CanFight}(x)) \wedge \text{Emperor}(\text{Taiji})) \Rightarrow \text{CanFight}(\text{Taiji})$$

- ▶ 要是命题逻辑的话，可以直接用假言推理

$$((\text{Emperor}(\text{Taiji}) \Rightarrow \text{CanFight}(\text{Taiji})) \wedge \text{Emperor}(\text{Taiji})) \Rightarrow \text{CanFight}(\text{Taiji})$$

- ▶ 谓词逻辑该怎么办，如何转化为命题逻辑？

- ▶ 置换： $\theta = \{x/\text{Taiji}\}$  把  $x$  换成  $\text{Taiji}$

- ▶ 形式： $\theta = \{v/g\}$   $v$ : variable,  $g$ : ground term

- ▶ 在置换  $\theta = \{x/\text{Taiji}\}$  下，谓词逻辑转化为了命题逻辑

$$((\text{Emperor}(x) \Rightarrow \text{CanFight}(x)) \wedge \text{Emperor}(\text{Taiji})) \Rightarrow \text{CanFight}(\text{Taiji})$$

$$((\text{Emperor}(\text{Taiji}) \Rightarrow \text{CanFight}(\text{Taiji})) \wedge \text{Emperor}(\text{Taiji})) \Rightarrow \text{CanFight}(\text{Taiji})$$

- ▶ 于是可以使用假言推理了

# 更一般的情况

- ▶ 贪婪的皇帝都是暴君，人都是贪婪的，皇太极是皇帝

$$\forall x \text{ Emperor}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Tyrant}(x)$$

$$\forall y \text{ Greedy}(y)$$

$$\text{Emperor}(\text{Nurhachi})$$

- ▶ 置换

$$\theta = \{x/\text{Nurhachi}, y/\text{Nurhachi}\}$$

$$\text{Emperor}(\text{Nurhachi}) \wedge \text{Greedy}(\text{Nurhachi}) \Rightarrow \text{Tyrant}(\text{Nurhachi})$$

$$\text{Greedy}(\text{Nurhachi})$$

$$\text{Emperor}(\text{Nurhachi})$$

- ▶ 结论:  $\text{Tyrant}(\text{Nurhachi})$
- ▶ 特点: 置换  $\theta$  使不同的逻辑表达式变得相同

## ► 使多个语句变得相同的置换称为合一

$Knows(John, x)$  和  $Knows(John, Jane)$ , 在  $\{x/Jane\}$  下合一

$Knows(John, x)$  和  $Knows(y, Bill)$ , 在  $\{x/Bill, y/John\}$  下合一

$Knows(John, x)$  和  $Knows(y, Brother(y))$ , 在  $\{y/John, x/Brother(John)\}$  下合一

## ► 变量分离


不同语句中的变量实际上是不同的, 应对变量重新命名

$Knows(John, x)$  和  $Knows(x, Elizabeth)$  看上去不能合一

$Knows(John, x)$  和  $Knows(y, Elizabeth)$  在  $\{y/John, x/Elizabeth\}$  下合一



- ▶ 假言推理：如果语句  $\alpha$  为真，且  $\alpha \Rightarrow \beta$ ，则  $\beta$  为真，即  $(\alpha \wedge (\alpha \Rightarrow \beta)) \models \beta$
- ▶ 命题逻辑：如果  $P_1, \dots, P_n$  为真，且  $P_1 \wedge \dots \wedge P_n \Rightarrow Q$ ，则  $Q$  为真
- ▶ 谓词逻辑：如果  $P_1, \dots, P_n$  为真， $P'_1 \wedge \dots \wedge P'_n \Rightarrow Q$ ，  
且存在合一置换  $\theta$ ，使得  $\text{SUBST}(\theta, P_i) = \text{SUBST}(\theta, P'_i)$   
则  $\text{SUBST}(\theta, Q)$  为真

$Emperor(Nurhachi)$	$\theta = \{x/Nurhachi, y/Nurhachi\}$	$Emperor(Nurhachi)$
$\forall y Greedy(y)$		$Greedy(Nurhachi)$
$\forall x Emperor(x) \wedge Greedy(x) \Rightarrow Tyrant(x)$		$Emperor(Nurhachi) \wedge Greedy(Nurhachi) \Rightarrow Tyrant(Nurhachi)$

基于广义假言推理即可进行限定性知识库的演绎推理

# 改变中国历史的神器

**红衣大炮**，16世纪初欧洲制造的前装重型滑膛炮，明代后期传入中国，也称为红夷大炮（荷兰人称为红夷）

天启六年（1626年），在宁远之战中发挥极大威力，八旗官兵血肉横飞，尸积如山，努尔哈赤本人也被击伤，后去世

崇祯四年（1631年），后金制成红衣大炮“**天佑助威大将军**”，还改进了铸造工艺，领先于明朝

崇祯十二年（1639年），清军拥有六十门自制的红衣大炮，连破明军据守的塔山、杏山二城

顺治元年（1645年）十二月，在入关战争中出击潼关，重创李自成的大顺军，李自成流窜至湖北通山县被害

**为什么清军的红衣大炮反而强于明军？**

# 有明朝叛徒相助

- ▶ 皇太极深知红衣大炮的厉害，在沈阳利用从明朝俘虏过来的工匠刘汉，成功仿制了红衣大炮 “天佑助威大将军”

## 刘汉有罪吗？

- ▶ 明朝：为敌对国制造武器的人有罪
- ▶ 清朝：皇太极时代已拥有红衣大炮
- ▶ 刘汉：为清朝制造大炮的明朝工匠
- ▶ 大炮：红衣大炮是威力极大的武器
- ▶ 敌对：处于交战中的国家是敌对国
- ▶ 交战：明朝和清朝当时正在交战中

- ▶ 明朝：为敌对国制造武器的人有罪

$$Ming(x) \wedge Weapon(y) \wedge Makes(x, y, z) \wedge Hostile(z) \Rightarrow Criminal(x)$$

- ▶ 清朝：皇太极时代已拥有红衣大炮

$$\exists x Hongyipo(x) \wedge Owns(Qing, x)$$

- ▶ 刘汉：为清朝制造大炮的明朝工匠

$$Hongyipo(x) \wedge Owns(Qing, x) \Rightarrow Makes(Liuhuan, x, Qing)$$

$$Ming(Liuhuan)$$

- ▶ 大炮：红衣大炮是威力极大的武器

$$Hongyipo(x) \Rightarrow Weapon(x)$$

- ▶ 敌对：处于交战中的国家是敌对国

$$Enemy(x, Ming) \Rightarrow Hostile(x)$$

- ▶ 交战：明朝和清朝当时正在交战中

$$Enemy(Qing, Ming)$$

# 前向链接

$\exists x \text{ Hongyipo}(x) \wedge \text{Owns}(\text{Qing}, x)$

$x/\text{Tianyou}$

**$\text{Hongyipo}(\text{Tianyou})$**   
 **$\text{Owns}(\text{Qing}, \text{Tianyou})$**

$\text{Hongyipo}(\text{Tianyou})$

$x/\text{Tianyou}$

$\text{Owns}(\text{Qing}, \text{Tianyou})$

**$\text{Makes}(\text{Liuhan}, \text{Tianyou}, \text{Qing})$**

$\text{Hongyipo}(x) \wedge \text{Owns}(\text{Qing}, x) \Rightarrow \text{Makes}(\text{Liuhan}, x, \text{Qing})$

$\text{Enemy}(\text{Qing}, \text{Ming})$

$x/\text{Qing}$

$\text{Enemy}(x, \text{Ming}) \Rightarrow \text{Hostile}(x)$

**$\text{Hostile}(\text{Qing})$**

$\text{Hongyipo}(\text{Tianyou})$

$x/\text{Tianyou}$

$\text{Hongyipo}(x) \Rightarrow \text{Weapon}(x)$

**$\text{Weapon}(\text{Tianyou})$**

$\text{Ming}(\text{Liuhan})$

$x/\text{Liuhan}, y/\text{Tianyou}, z/\text{Qing}$

$\text{Weapon}(\text{Tianyou})$

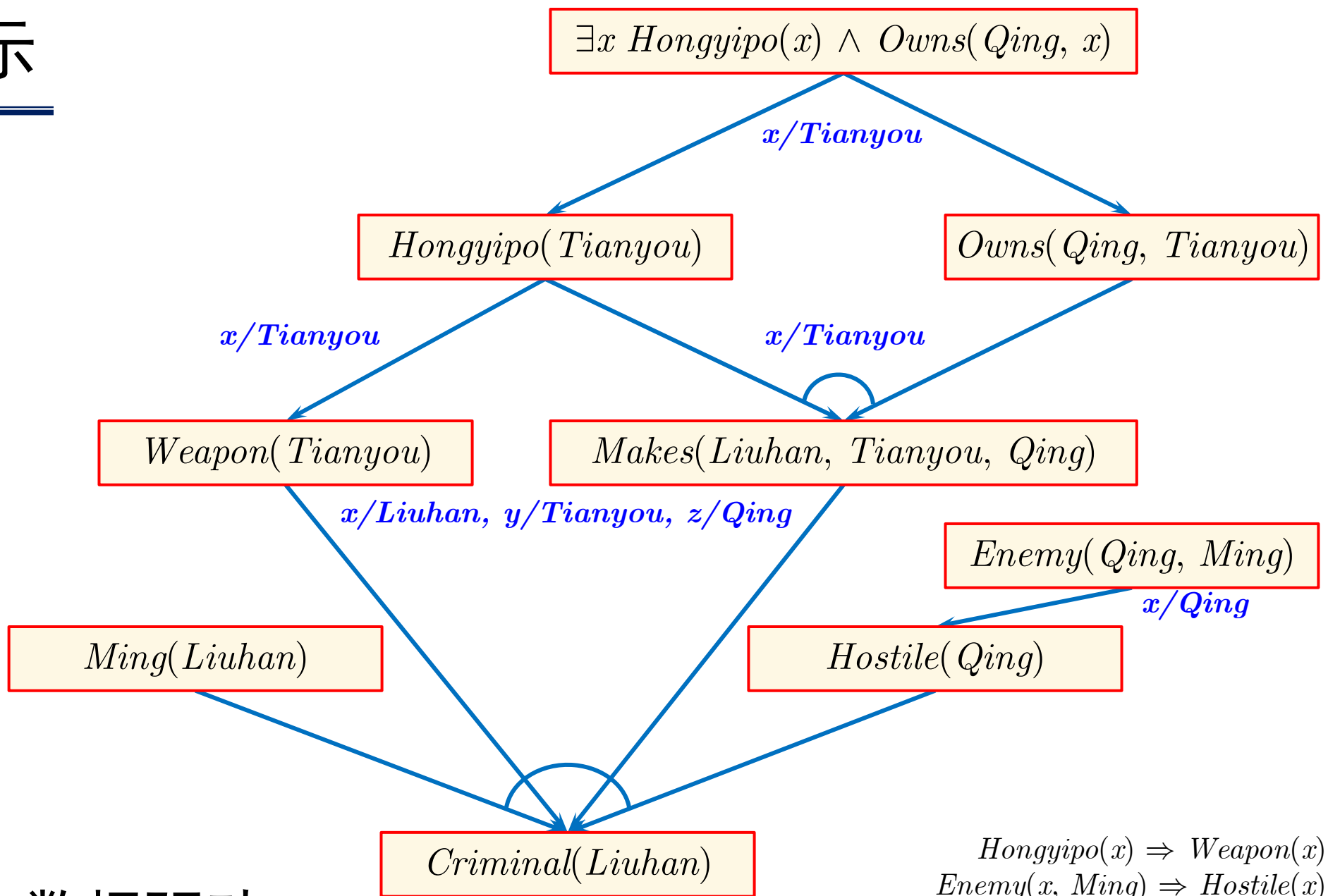
**$\text{Criminal}(\text{Liuhan})$**

$\text{Makes}(\text{Liuhan}, \text{Tianyou}, \text{Qing})$

$\text{Hostile}(\text{Qing})$

$\text{Ming}(x) \wedge \text{Weapon}(y) \wedge \text{Makes}(x, y, z) \wedge \text{Hostile}(z) \Rightarrow \text{Criminal}(x)$

# 与或图表示



数据驱动

$\text{Hongyipo}(x) \Rightarrow \text{Weapon}(x)$   
 $\text{Enemy}(x, \text{Ming}) \Rightarrow \text{Hostile}(x)$

$\text{Hongyipo}(x) \wedge \text{Owns}(\text{Qing}, x) \Rightarrow \text{Makes}(\text{Liuhan}, x, \text{Qing})$   
 $\text{Ming}(x) \wedge \text{Weapon}(y) \wedge \text{Makes}(x, y, z) \wedge \text{Hostile}(z) \Rightarrow \text{Criminal}(x)$

# 归结原理

Resolution rule

--	--

--	--

# 归结原理 (命题逻辑)

- 如果  $\alpha \vee \beta$  和  $\neg\alpha \vee \gamma$  均为真, 则  $\beta \vee \gamma$  亦为真

$$(\alpha \vee \beta) \wedge (\neg\alpha \vee \gamma) \models (\beta \vee \gamma)$$



消除了

$\alpha$	$\beta$	$\neg\alpha$	$\gamma$	$\alpha \vee \beta$	$\neg\alpha \vee \gamma$	$(\alpha \vee \beta) \wedge (\neg\alpha \vee \gamma)$	$\beta \vee \gamma$
F	F	T	F	F	T	F	F
F	F	T	T	F	T	F	T
F	T	T	F	T	T	T	T
F	T	T	T	T	T	T	T
T	F	F	F	T	F	F	F
T	F	F	T	T	T	T	T
T	T	F	F	T	F	F	T
T	T	F	T	T	T	T	T



# 归结原理 (谓词逻辑)

- 如果  $\alpha \vee \beta$  和  $\neg\alpha \vee \gamma$  均为真, 则  $\beta \vee \gamma$  亦为真

$$(\alpha \vee \beta) \wedge (\neg\alpha \vee \gamma) \models (\beta \vee \gamma)$$



消除了

如果存在合一置换使得两个子句包含互补的原子语句,  
则可通过消去该原子语句对这两个子句进行归结

$\neg \text{Loves}(y, \text{Taiji})$	$\theta = \{y / G(\text{Taiji})\}$	$\neg \text{Loves}(G(\text{Taiji}), \text{Taiji})$
		
$\text{Loves}(G(\text{Taiji}), \text{Taiji})$		$\text{Loves}(G(\text{Taiji}), \text{Taiji})$

# 是谁杀了褚英



仁厚之人为民所爱  
残暴之人为民所弃  
皇太极仁厚之人也  
褚英为其父弟所害  
诛英者其父耶弟耶

爱新觉罗·褚英是清太祖爱新觉罗·努尔哈赤的长子。努尔哈赤的胞弟爱新觉罗·舒尔哈齐死后，努尔哈赤开始逐渐让褚英带兵并主持一部分军政事务。

仁厚之人为民所爱 ^  
残暴之人为民所弃 ^  
皇太极仁厚之人也 ^  
褚英为其父弟所害 ^  
└ 诛英者努尔哈赤也

爱新觉罗·褚英是清太祖爱新觉罗·努尔哈赤的长子。努尔哈赤的胞弟爱新觉罗·舒尔哈齐死后，努尔哈赤开始逐渐让褚英带兵并主持一部分军政事务。

- ▶ 仁厚之人为民所爱（爱所有人的人，存在爱他的人）

$$\forall x [(\forall y \text{ Person}(y) \Rightarrow \text{Loves}(x, y)) \Rightarrow (\exists y \text{ Loves}(y, x))]$$

- ▶ 残暴之人为民所弃（杀人的人，没人爱他）

$$\forall x [(\exists y \text{ Person}(x) \wedge \text{Kills}(x, y)) \Rightarrow (\forall y \neg \text{Loves}(y, x))]$$

- ▶ 皇太极仁厚之人也

$$\forall x \text{ Person}(x) \Rightarrow \text{Loves}(\text{Taiji}, x)$$

- ▶ 褚英为其父弟所害

$$\text{Prince}(\text{Cuyen})$$

$$\forall x \text{ Prince}(x) \Rightarrow \text{Person}(x)$$

$$\text{Kills}(\text{Nurhachi}, \text{Cuyen}) \vee \text{Kills}(\text{Taiji}, \text{Cuyen})$$

- ▶ 诛英者努尔哈赤也

$$\text{Kills}(\text{Nurhachi}, \text{Cuyen})$$

# 转换为合取范式，获得子句集

- ▶ 仁厚之人为民所爱

$$Person(F(x)) \vee Loves(G(x), x)$$

$$\neg Loves(x, F(x)) \vee Loves(G(x), x)$$

- ▶ 残暴之人为民所弃

$$\neg Loves(y, x) \vee \neg Person(z) \vee \neg Kills(x, z)$$

- ▶ 皇太极仁厚之人也

$$\neg Person(x) \vee Loves(Taiji, x)$$

- ▶ 褚英为其父弟所害

$$Prince(Cuyen)$$

$$\neg Prince(x) \vee Person(x)$$

$$Kills(Nurhachi, Cuyen) \vee Kills(Taiji, Cuyen)$$

- ▶ 诛英者努尔哈赤也

$$\neg Kills(Nurhachi, Cuyen)$$

# 归结

$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\neg Loves(y, x) \vee \neg Person(z) \vee \neg Kills(x, z)$

$\neg Person(x) \vee Loves(Taiji, x)$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

$Prince(Cuyen)$

$\rightarrow Person(Cuyen)$

$\neg Prince(x) \vee Person(x)$

$Kills(Nurhachi, Cuyen) \vee Kills(Taiji, Cuyen)$

$\rightarrow Kills(Taiji, Cuyen)$

$\neg Kills(Nurhachi, Cuyen)$

# 归结

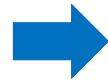
$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\neg Loves(y, x) \vee \neg Person(z) \vee \neg Kills(x, z)$

$\neg Person(x) \vee Loves(Taiji, x)$

$Person(Cuyen)$



$\neg Loves(y, Taiji) \vee$   
 $\neg Person(Cuyen)$

$Kills(Taiji, Cuyen)$

# 归结

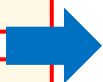
$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\neg Person(x) \vee Loves(Taiji, x)$

$Person(Cuyen)$

$\neg Loves(y, Taiji) \vee \neg Person(Cuyen)$



$\neg Loves(y, Taiji)$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结



$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\neg Person(x) \vee Loves(Taiji, x)$

$\neg Loves(y, Taiji)$



$Loves(G(x), x) \vee$   
 $Loves(Taiji, F(x))$

如果存在合一替换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

# 归结

$\neg \text{Loves}(x, F(x)) \vee \text{Loves}(G(x), x)$

$\text{Loves}(Taiji, F(x)) \vee \text{Loves}(G(x), x)$

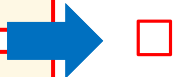


$\text{Loves}(G(Taiji), Taiji)$

$\neg \text{Loves}(y, Taiji)$

$\neg \text{Loves}(y, \text{Taiji})$

$\text{Loves}(G(\text{Taiji}), \text{Taiji})$



如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

# 从怪兽开始，以怪兽结束

$$(\forall x, y B(x, y) \Leftrightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \wedge \neg B(1, 1) \wedge P(1, 2)$$

## 等价消去

$$\begin{aligned} & (\forall x, y (B(x, y) \Rightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \\ & \quad \wedge (P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1) \Rightarrow B(x, y)) \\ & \quad \wedge \neg B(1, 1) \wedge P(1, 2)) \end{aligned}$$

## 隐含消去

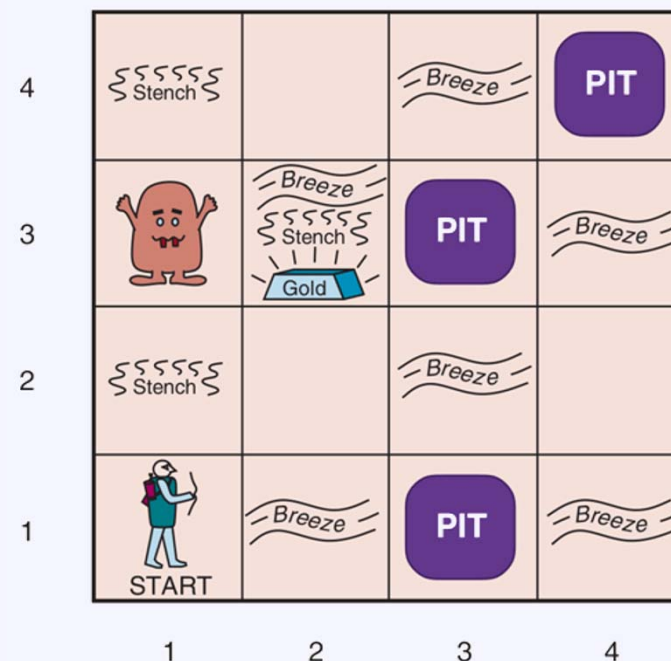
$$\begin{aligned} & (\forall x, y (\neg B(x, y) \vee P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \\ & \quad \wedge \neg ((P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \vee B(x, y)) \\ & \quad \wedge \neg B(1, 1) \wedge P(1, 2)) \end{aligned}$$

## 否定深入

$$\begin{aligned} & (\forall x, y (\neg B(x, y) \vee P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \\ & \quad \wedge (\neg P(x-1, y) \wedge \neg P(x+1, y) \wedge \neg P(x, y-1) \wedge \neg P(x, y+1)) \vee B(x, y)) \\ & \quad \wedge \neg B(1, 1) \wedge P(1, 2)) \end{aligned}$$

## 变量换名

## 存在量词消去（斯柯林化）



- ▶  $(x, y)$  有微风:  $B(x, y)$
- ▶  $(x, y)$  有陷阱:  $P(x, y)$
- ▶  $\neg B(1, 1)$
- ▶  $\forall x, y B(x, y) \Leftrightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)$

$\neg P(1, 2) ?$

# 从怪兽开始，以怪兽结束

$(\forall x, y B(x, y) \Leftrightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)) \wedge \neg B(1, 1) \wedge P(1, 2)$

## 全称量词消去

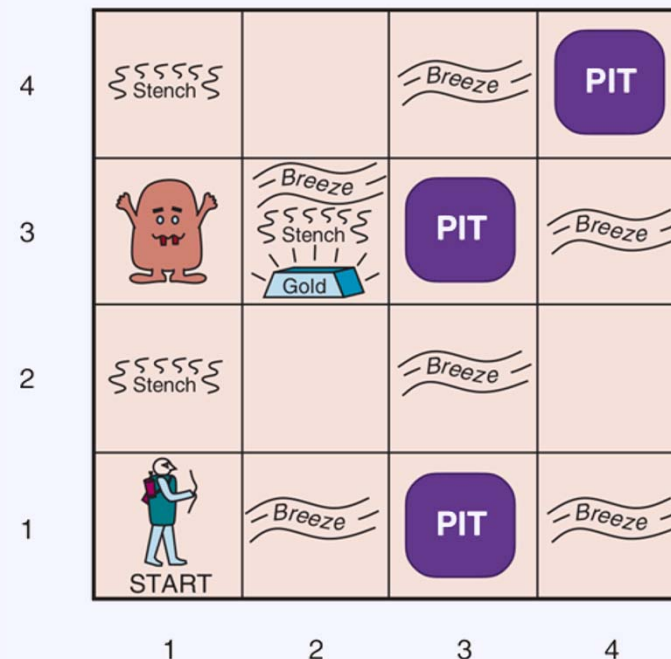
$(\neg B(x, y) \vee P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1))$   
 $\wedge ((\neg P(x-1, y) \wedge \neg P(x+1, y) \wedge \neg P(x, y-1) \wedge \neg P(x, y+1)) \vee B(x, y))$   
 $\wedge \neg B(1, 1) \wedge P(1, 2)$

## 分配律

$(\neg B(x, y) \vee P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1))$   
 $\wedge (\neg P(x-1, y) \vee B(x, y)) \wedge (\neg P(x+1, y) \vee B(x, y)) \wedge (\neg P(x, y-1) \vee B(x, y))$   
 $\wedge (\neg P(x, y+1) \vee B(x, y))$   
 $\wedge \neg B(1, 1) \wedge P(1, 2)$

## 归结

$\neg P(x, y+1) \vee B(x, y)$   
 $\neg B(1, 1)$   $\xrightarrow{\theta = \{x/1, y/1\}}$   $\neg P(1, 2)$   
 $P(1, 2) \xrightarrow{\quad} \square$



- ▶  $(x, y)$  有微风:  $B(x, y)$
- ▶  $(x, y)$  有陷阱:  $P(x, y)$
- ▶  $\neg B(1, 1)$
- ▶  $\forall x, y B(x, y) \Leftrightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)$

$\neg P(1, 2) ?$

**Thank you very much!**