

Homework6

8.24 用一个相容的词汇表（需要你自己定义）在一阶逻辑中表示下列语句：

- 某些学生在2001年春季学期上法语课
- 上法语课的每个学生都通过了考试
- 只有一个学生在2001年春季学期上希腊语课
- 希腊语课的最好成绩总是比法语课的最好成绩高
- 每个买保险的人都是聪明的
- 没有人会买昂贵的保险
- 有一个代理，他只卖保险给那些没有投保的人
- 镇上有一个理发师，他给所有不自己刮胡子的人刮胡子
- 在英国出生的人，如果其双亲都是英国公民或永久居住者，那么此人生来就是一个英国公民
- 在英国以外的地方出生的人，如果其双亲生来就是英国公民，那么此人血统上是一个英国公民
- 政治家可以一直愚弄某些人，也可以在某个时候愚弄所有人，但是他们无法一直愚弄所有的人

设计如下的词汇表：

- $Student(x)$ 表示 x 是学生
- $Select(x, c, s)$ 表示学生 x 在 s 学期上 c 课
- $Pass(x, c, s)$ 表示在 s 学期上 c 课的学生 x 通过了考试
- $Grade(x, c, s)$ 表示在 s 学期上 c 课的学生 x 的成绩
- $Person(x)$ 表示 x 是正常人
- $Smart(x)$ 表示 x 是聪明的
- $Policy(x)$ 表示 x 是保险
- $Expensive(x)$ 表示 x 是昂贵的
- $Agent(x)$ 表示 x 是代理
- $Insured(x)$ 表示 x 已被投保
- $Buy(x, y, g)$ 表示 x 向 y 购买了 g
- $Sell(x, y, g)$ 表示 x 把 g 卖给了 y
- $Barber(x)$ 表示 x 是理发师
- $Shave(x, y)$ 表示 x 给 y 刮胡子
- $Born(x, c)$ 表示 x 在国家 c 出生
- $Parent(x, y)$ 表示 x 是 y 的双亲之一
- $Citizen(x, c, b)$ 表示 x 基于 b 是国家 c 的公民
- $Resident(x, c)$ 表示 x 是国家 c 的永久居住者
- $Politician(x)$ 表示 x 是政治家
- $Fool(x, y, t)$ 表示 x 在 t 时刻愚弄了 y

给出上述语句的表示

- $\exists x \quad Student(x) \wedge Select(x, French, 2001Spring)$
- $\forall x, s \quad Student(x) \wedge Select(x, French, s) \Rightarrow Pass(x, French, s)$

- $\exists x \text{ Student}(x) \wedge \text{Select}(x, \text{Greek}, s) \wedge (\forall y \ y \neq x \Rightarrow \neg \text{Select}(y, \text{Greek}, 2001\text{Spring}))$
- $\forall s \exists x \forall y \ \text{Grade}(x, \text{Greek}, s) > \text{Grade}(y, \text{French}, s)$
- $\forall x \ \text{Person}(x) \wedge (\exists a, p \ \text{Policy}(p) \wedge \text{Agent}(a) \wedge \text{Buy}(x, a, p)) \Rightarrow \text{Smart}(x)$
- $\forall x, p, a \ \text{Person}(x) \wedge \text{Policy}(p) \wedge \text{Expensive}(p) \Rightarrow \neg \text{Buy}(x, a, p)$
- $\exists a \ \text{Agent}(a) \wedge (\forall x, p \ (\text{Policy}(p) \wedge \text{Sell}(a, x, p)) \Rightarrow (\text{Person}(x) \wedge \neg \text{Insured}(x)))$
- $\exists x \ \text{Barber}(x) \wedge (\forall y \ \text{Person}(y) \wedge \neg \text{Shave}(y, y) \Rightarrow \text{Shave}(x, y))$
- $\forall x \ \text{Person}(x) \wedge \text{Born}(x, \text{UK}) \wedge \left(\forall y \ \text{Parent}(y, x) \wedge \left((\exists b \ \text{Citizen}(y, \text{UK}, b)) \vee \text{Resident}(y, \text{UK}) \right) \right) \Rightarrow \text{Citizen}(x, \text{UK}, \text{"Birth"})$
- $\forall x \ \text{Person}(x) \wedge \neg \text{Born}(x, \text{UK}) \wedge \left(\forall y \ \text{Parent}(y, x) \wedge (\exists b \ \text{Citizen}(y, \text{UK}, b)) \right) \Rightarrow \text{Citizen}(x, \text{UK}, \text{"Descent"})$
- $\forall x \ \text{Politician}(x) \Rightarrow (\exists y \forall t \ \text{Person}(y) \wedge \text{Fool}(x, y, t)) \wedge (\exists t \forall y \ \text{Person}(y) \wedge \text{Fool}(x, y, t)) \wedge \neg (\forall t \forall y \ \text{Person}(y) \wedge \text{Fool}(x, y, t))$

8.17

解释下面给出的 Wumpus 世界中相邻方格的定义存在什么问题：

$$\forall x, y \ \text{Adjacent}([x, y], [x + 1, y]) \wedge \text{Adjacent}([x, y], [x, y + 1])$$

Answer:

这样的定义仅考虑了坐标轴正方向上的相邻，但实际上相邻关系是无向的（即对称的）；另外定义没有考虑到世界边界的作用关系，可能会给出在实际问题中不存在的相邻关系。

一个更合理的定义是

$$\begin{aligned} & \forall 1 \leq x < 4, 1 \leq y < 4 \\ & \text{Adjacent}([x, y], [x + 1, y]) \wedge \text{Adjacent}([x + 1, y], [x, y]) \\ & \wedge \text{Adjacent}([x, y], [x, y + 1]) \wedge \text{Adjacent}([x, y + 1], [x, y]) \end{aligned}$$

9.3

假定知识库中只包括一条语句： $\exists x \text{ AsHighAs}(x, \text{Everest})$ ，下列哪个语句是应用存在量词实例化以后的合法结果？

1. $\text{AsHighAs}(\text{Everest}, \text{Everest})$
2. $\text{AsHighAs}(\text{Kilimanjaro}, \text{Everest})$
3. $\text{AsHighAs}(\text{Kilimanjaro}, \text{Everest}) \wedge \text{AsHighAs}(\text{BenNevis}, \text{Everest})$

Answer:

语句1不合法，替换变元的应当是从未在知识库中出现过的常量符号，而不是已经出现过的 *Everest*

语句2是合法

语句3不合法，使用了两次实例化，而 \exists 只要一个

9.4

对于下列每对原子语句，如果存在，请给出最一般合一置换：

1. $P(A, B, B), P(x, y, z)$
2. $Q(y, G(A, B)), Q(G(x, x), y)$

3. $Older(Father(y), y), Older(Father(x), John)$

4. $Knows(Father(y), y), Knows(x, x)$

Answer

1. $\{A/x, B/y, B/z\}$

2. 合一失败, 如果 $y = G(A, B), y = G(x, x)$ 那么 x 取 A 和 B , 矛盾。两个语句对不同对象使用相同变量名则不能合一, 要通过重命名解决

3. $\{y/John, x/John\}$

4. 合一失败, 发生检验中 y 和 $Father(y)$ 无法合一

9.6

写出下列语句的逻辑表示, 使得它们适用一般化假言推理规则:

- 马、奶牛和猪都是哺乳动物。
- 一匹马的后代是马。
- Bluebeard 是一匹马。
- Bluebeard 是 Charlie 的家长。
- 后代和家长是逆关系。
- 每个哺乳动物都有一个家长。

Answer

定义

- $Horse(x)$ 表示 x 是马。
- $Cow(x)$ 表示 x 是奶牛。
- $Pig(x)$ 表示 x 是猪。
- $Mammal(x)$ 表示 x 是哺乳动物。
- $Descendant(x, y)$ 表示 x 是 y 的后代。
- $Parent(x, y)$ 表示 x 是 y 的家长。

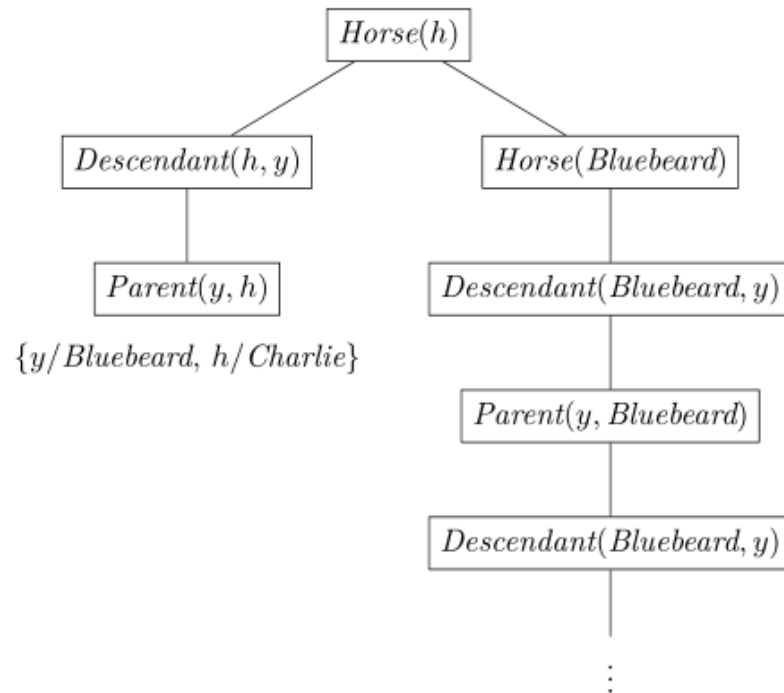
逻辑表示

- $Horse(x) \Rightarrow Mammal(x)$
 $Cow(x) \Rightarrow Mammal(x)$
 $Pig(x) \Rightarrow Mammal(x)$
 - $Descendant(x, y) \wedge Horse(y) \Rightarrow Horse(x)$
 - $Horse(Bluebeard)$
 - $Parent(Bluebeard, Charlie)$
 - $Descendant(x, y) \Rightarrow Parent(y, x)$ $Parent(x, y) \Rightarrow Descendant(y, x)$
 - $Mammal(x) \Rightarrow Parent(Gen(x), x)$, 其中 $Gen(x)$ 是一个 Skolem 范式
-

9.13

用到你在习题9.6中写出的语句, 运用反向链接算法来回答问题

a. 画出用穷举反向链接算法为查询 $\exists h horse(h)$ 生成的证明树, 其中子句按照给定的顺序进行匹配



如图所示，会有无限递归的现象

b. 对于本邻域，你注意到了什么？

注意到树中出现的无限延伸，这实际上是由于规则子句的顺序引起的，可以通过在规则 $Descendant(x, y) \wedge Horse(y) \Rightarrow Horse(x)$ 之前指定匹配顺序来得到解，但是如果要求穷举所有的解，那与子句顺序无关，循环一定会发生。

c. 实际上从你的语句中得出了多少个 h 的解？

实际上得到了 *Bluebeard* 和 *Charlie* 两个解。