CSC242: Homework 2.5
AIMA Chapter 9

1. One of the early successes of AI was using rule-based systems (a.k.a. production systems or "expert" systems) for clinical diagnosis. These systems did not do logical inference, strictly speaking, but they did apply rules to derive conclusions. Let's take a look at a small example based on the real-life system MYCIN (Wikipedia) that diagnosed bacterial infections.

   MYCIN used diagnostic rules like the following:

$$\forall x \; \mathit{Gram}(x, \mathit{Negative}) \land \mathit{Morph}(x, \mathit{Rod}) \land \mathit{Anaerobic}(x) \Rightarrow \mathit{Identity}(x, \mathit{Bacteroides}, 0.6) \tag{1}$$

   That is, if an organism is "gram-negative" (a staining test), has a "rod" morphology (shape), and is anerobic (does not use oxygen), then with 60% confidence it is *Bacteroides*.

   MYCIN used its own inference procedure to draw conclusions. We will do first-order inference, but with only the following inference rules:

   UI: $\dfrac{\forall v \; \alpha}{\text{Subst}(\{v/g\}, \alpha)}$ for any variable v and ground term g

   AI: $\dfrac{\alpha, \beta}{\alpha \land \beta}$

   MP: $\dfrac{\alpha \Rightarrow \beta, \alpha}{\beta}$

   The expression Subst(Θ, α) denotes the result of applying subtitution Θ to sentence α, yielding a new sentence.

   Now suppose that you have the following information about an infectious organism denoted by the constant symbol $\mathit{ORG1}$:

$$\mathit{Gram}(\mathit{ORG1}, \mathit{Negative}) \tag{2}$$
$$\mathit{Morph}(\mathit{ORG1}, \mathit{Rod}) \tag{3}$$
$$\mathit{Anaerobic}(\mathit{ORG1}) \tag{4}$$

   Give a proof of the organism's identity using *only* the given inference rules. Be sure to show which rule is being used at each step and any substitutions.

2. Suppose we introduce a new quantifier, $\Box$, and the following inference rule:

$$\text{BX:} \quad \frac{\Box v\ \alpha}{\mathsf{Subst}(\{v/n\}, \alpha)} \quad \text{for any variable } v \text{ and natural number } n$$

   Now suppose we have the following knowledge base:

$$\Box v\ Num(v) \tag{1}$$
$$\forall n, m\ Num(n) \land Num(m) \Rightarrow Num(n+m) \tag{2}$$

   Use the BX inference rule with the inference rules from the previous question to prove that $Num(3+4)$.

3. Do you think the following first-order knowledge base is consistent?

$$\exists x\ P(x) \tag{1}$$
$$\forall x\ \neg P(x) \tag{2}$$

   Use the UI inference rule from before and the following inference rule to prove that it's not, by deriving a contradiction.

$$\text{EI:} \quad \frac{\exists v\ \alpha}{\mathsf{Subst}(\{v/k\}, \alpha)} \quad \text{where } k \text{ is a new constant symbol not already occurring in the knowledge base}$$

4. For each pair of atomic sentences, give the most general unifier if one exists:

   (a) $P(A, B, B)$ and $P(x, y, z)$

   (b) $Q(y, g(A, B))$ and $Q(g(x, x), y)$

   (c) $Older(Father(y), y)$ and $Older(Father(x), John)$

   (d) $Knows(Father(y), y)$ and $Knows(x, x)$

5. From "Horses are animals," it follows that "The head of a horse is the head of an animal." How would you demonstrate that this inference is valid? Think about it. . . then read my suggestion for how to proceed:

   (a) Translate both the premise and the conclusion into first-order logic using the predicates $Horse(x)$ ("x is a horse"), $Animal(x)$ ("x is an animal"), and $HeadOf(h, x)$ ("h is the head of x").

   (b) Negate the conclusion, and convert the premise and the negated conclusion into conjunctive normal form.

   (c) Use resolution in the appropriate way to show that the conclusion logically follows from the premise.

6. Suppose a knowledge base contains just the following first-order Horn clauses:

$$Ancestor(Mother(x), x)$$
$$Ancestor(x, y) \wedge Ancestor(y, z) \Rightarrow Ancestor(x, z)$$

Consider a forward-chaining algorithm that, on the jth iteration, terminates if the KB contains a sentence that unifies with the query, and otherwise adds to the KB every atomic sentence that can be inferred from the sentences already in the KB after iteration $j - 1$.

(a) For each of the following queries, say whether the algorithm will (1) give an answer (if so, give that answer); or (2) terminate with no answer; or (3) not terminate.

i. $Ancestor(Mother(y), John)$
ii. $Ancestor(Mother(Mother(y)), John)$
iii. $Ancestor(Mother(Mother(Mother(y))), y)$
iv. $Ancestor(Mother(John), Mother(Mother(John)))$

(b) Can a resolution algorithm prove the sentence $\neg Ancestor(John, John)$?