National University of Singapore
School of Computing
CS3243: Foundations of Artificial Intelligence
Solutions for Tutorial 7

**Readings:** AIMA Chapter 9

1. (Question 9.4 from AIMA) For the following pairs of atomic sentences, give the most general unifier if one exists:

   (a) $P(A, B, B)$, $P(x, y, z)$
   $\{x/A, y/B, z/B\}$

   (b) $Q(y, G(A, B))$, $Q(G(x, x), y)$
   *Fail*

   (c) $Older(Father(y), y)$, $Older(Father(x), John)$
   $\{x/John, y/John\}$

   (d) $Knows(Father(y), y)$, $Knows(x, x)$
   *Fail*

2. What is the problem in each the following first order logic statements? Suggest how these statements can be corrected.

   *(a) This statement really means: everything under the universe is a boy and is tall. It excludes the possibility that something in the universe need not be a boy, and that something that is not a boy need not be tall.*

   *Corrected statement:* $\forall x\ Boy(x) \Rightarrow Tall(x)$ *Note that this statement holds true even if* $x$ *is not a boy and is not tall. Thus, the correct way to express such "for all" statements is usually via an implication.*

   *(b) This statement is vacuously true as long as there is something in the universe that is not a boy. This is because the implication* $Boy(x) \Rightarrow Tall(x)$ *is satisfied whenever* $x$ *is not a boy.*

   *Corrected statement:* $\exists x\ Boy(x) \wedge Tall(x)$ *Note that this statement requires that there is something that is both a boy and is tall. Thus, the correct way to express such "there exists" statements is usually via a conjunction.*

3. An atheist asked two knowledge engineers to write a rule to say that Nothing is divine! The first engineer wrote $\neg\exists x\ Divine(x)$ and transformed it into the following clause:

$$\neg Divine(G1)$$

   where $G1$ is a Skolem constant. The second engineer wrote $\forall x\ \neg Divine(x)$ and transformed it into the following clause:

$$\neg Divine(x)$$

   Why did they produce two different versions? Which version is correct?

   *The second version is correct and the first version is wrong. The first engineer made a mistake by skolemizing the sentence before moving* $\neg$ *inwards.*

4. Two English sentences "Anyone who takes an AI course is smart" and "Any course that teaches an AI topic is an AI course" have been represented in first-order logic:

$$\forall x\ (\exists y\ AI\_course(y) \wedge Takes(x,y)) \Rightarrow Smart(x)$$

$$\forall x\ (\exists y\ AI\_topic(y) \wedge Teaches(x,y)) \Rightarrow AI\_course(x)$$

It is also known that John takes CS3243 and CS3243 teaches Inference which is an AI topic. Represent these facts as first-order logic sentences. Now convert all first-order logic sentences into conjunctive normal form and use resolution to prove that "John is smart."

*Convert 1st sentence to CNF:*

$$\forall x\ \neg(\exists y\ AI\_course(y) \wedge Takes(x,y)) \vee Smart(x)$$

$$\forall x\ \forall y\ \neg AI\_course(y) \vee \neg Takes(x,y) \vee Smart(x)$$

$$(1)\quad \neg AI\_course(y) \vee \neg Takes(x,y) \vee Smart(x)$$

*Convert 2nd sentence to CNF:*

$$\forall x\ \neg(\exists y\ AI\_topic(y) \wedge Teaches(x,y)) \vee AI\_course(x)$$

$$\forall x\ \forall y\ \neg AI\_topic(y) \vee \neg Teaches(x,y)) \vee AI\_course(x)$$

$$\neg AI\_topic(y) \vee \neg Teaches(x,y)) \vee AI\_course(x)$$

*Standardize variables apart from the 1st sentence:*

$$(2)\quad \neg AI\_topic(u) \vee \neg Teaches(v,u)) \vee AI\_course(v)$$

*Facts:*
(3) $Takes(John, CS3243)$
(4) $Teaches(CS3243, Inf)$
(5) $AI\_topic(Inf)$

*Added negated goal:*
(6) $\neg Smart(John)$

*Resolution:*
(7) $\neg AI\_course(y) \vee \neg Takes(John, y)$ *[From (1) & (6), $\theta = \{x/John\}$]*
(8) $\neg AI\_course(CS3243)$ *[From (3) & (7), $\theta = \{y/CS3243\}$]*
(9) $\neg AI\_topic(u) \vee \neg Teaches(CS3243, u))$ *[From (2) & (8), $\theta = \{v/CS3243\}$]*
(10) $\neg Teaches(CS3243, Inf))$ *[From (9) & (5), $\theta = \{u/Inf\}$]*
(11) [] *[From (10) & (4)]*

5. (Slightly modified from Question 9.19 of AIMA) Here are two sentences in the language of first-order logic:
$(A) : \forall x\ \exists y\ (x \geq y)$
$(B) : \exists y\ \forall x\ (x \geq y)$

   (a) Assume that the variables range over all the natural numbers $0, 1, 2, \cdots$ and that the "$>$" predicate means "is greater than or equal to." Under this interpretation, translate (A) and (B) into English.

   *(A) is translated as "For every natural number x, there is a natural number y that is less than or equal to x." (B) is translated as "There is a natural number y that is less than or equal to all natural numbers."*

(b) Is (A) true under this interpretation? Is (B) true under this interpretation? *Yes. Yes.*

(c) Does (A) logically entail (B)? Does (B) logically entail (A)? Justify your answers.

*(A) does not logically entail (B). Consider the domain of integers* $\{\cdots, -2, -1, 0, 1, 2, \cdots\}$, *and the interpretation where "$\geq$" maps to "is greater than or equal to." Then (A) is true but (B) is false.*

*(B) logically entails (A)*
*We will prove by resolution.*
*First convert (B) to CNF:*
$\exists\, y\ \forall\, x\ (x \geq y)$
$\forall\, x\ (x \geq C)$ *(skolemization)*
$x \geq C$
*Next convert* $\neg(A)$ *to CNF:*
$\neg(\forall\, x\ \exists\, y\ (x \geq y))$
$\exists\, x\ \forall\, y\ \neg(x \geq y)$
$\forall\, y\ \neg(D \geq y)$ *(skolemization)*
$\neg(D \geq y)$
*The two clauses unify with mgu* $\{x/D, y/C\}$ *to derive the empty clause.*

6. PSA would like to implement its tax system on ships and cargo for its Brani port as part of a first order logic system. You have been hired as a knowledge engineer to convert the following predicates into FOL representation. You may use any predicate that you create in previous parts for subsequent parts. You may also define new constants and predicates.

   Note that variables are in lowercase, whereas constant, predicate and function symbols start with uppercase. Given the functions:

   $Arrival\_Time(ship)$
   $Departure\_Time(ship)$

   And the predicates:

   $Unload\_From\_Ship(cargo, ship, arrival\_time)$
   $Load\_Onto\_Ship(cargo, ship, departure\_time)$
   $Weapons(cargo)$

   (a) In the PSA system, can ship objects (n.b., not constant symbols) arrive at the Brani port multiple times? Justify your answer.

   *Yes, they can, but the same ship object will have to be mapped different constant symbols in the interpretation. This is necessary as the arrival and departure times are specified as functions, which always return the same object.*

   (b) Write a FOL predicate of a simplified tax law for cargos entering and not departing Singapore. Note that ships unload and load at their arrival and departure times.

   $$\forall\, c\ (\exists\, s_1\ Unload\_From\_Ship(c, s_1, Arrival\_Time(s_1))) \wedge$$
   $$(\neg\exists\, s_2\ Load\_Onto\_Ship(c, s_2, Departure\_Time(s_2))) \Rightarrow TaxableCargo(c)$$

   (c) Aside from cargo, ships are also taxed. A ship is taxable upon entry to Singapore unless all the ship's cargo are weapons (slated for the armed forces).

   $$\forall\, s\ (\exists\, c\ Unload\_From\_Ship(c, s, Arrival\_Time(s) \wedge \neg Weapons(c)) \Rightarrow Taxable(s)$$

   (d) State your answer from Part (c) in Conjunctive Normal Form.

   $$\neg Unload\_From\_Ship(c, s, Arrival_Time(s) \vee Weapons(c) \vee Taxable(s)$$

(e) Using the following observations, use resolution by refutation to answer the query $Taxable(Storm\_King)$:

$$Unload\_From\_Ship(Torpedos, Storm\_King, Arrival\_Time(Storm\_King))$$

$$Weapons(Torpedo)$$

$$Unload\_From\_Ship(Laser\_Parts, Storm\_King, Arrival\_Time(Storm\_King))$$

$$\neg Weapons(Laser\_Parts)$$

*Resolution: Add* $\neg Taxable(Storm_King)$ *to* $KB$
*Resolves with sentence in part (d) above with substitution* $\theta = \{s/Storm\_King\}$ *to yield:*

$$\neg Unload\_From\_Ship(c, Storm\_King, Arrival_Time(Storm\_King) \vee Weapons(c)$$

*Resolves with* $Unload\_From\_Ship(Laser\_Parts, Storm\_King, Arrival\_Time(Storm\_King))$ *with substitution* $\theta = \{c/Laser\_Parts\}$ *to yield:*

$$Weapons(Laser\_Parts)$$

*Resolves with* $\neg Weapons(Laser\_Parts)$ *to yield false.*