6.034 Quiz 3 Solutions, Spring 2007

1 Predicate Logic (25 points)

Hint: Make sure you understand the definition of **clause** in propositional logic.

1. (3 pts) The truth-table method is a:

   (a) sound $\surd$

   (b) complete $\surd$

   (c) efficient

   procedure for deciding entailment for propositional logic. Circle all the true choices.

2. (3 pts) **True** or **False**: The only unsatisfiable **clause** in propositional logic is the empty **clause**.

   True. *Any clause with one or more variables can be satisfied by assigning True to one of the variables. There is no interpretation that makes the empty clause True, therefore the empty clause is unsatisfiable.*

3. (3 pts) **True** or **False**: $(A \vee \neg B) \wedge (\neg A \vee B)$ is satisfiable.

   True. *We only need to find an assignment that makes the sentence True, such as* $A = B = True$.

4. (3 pts) **True** or **False**: $A \leftrightarrow B$ entails $A \wedge B$

   False. *When* $A = B = false$, $A \leftrightarrow B$ *is True but* $A \wedge B$ *is not. Therefore,* $A \leftrightarrow B$ *doesn't entail* $A \wedge B$.

5. (3 pts) **True** or **False**: $A \leftrightarrow B$ entails $A \vee \neg B$

   True. *From* $A \leftrightarrow B \equiv (A \vee \neg B) \wedge (\neg A \vee B)$, *we know that every interpretation that satisfies* $A \leftrightarrow B$ *also satisfies* $A \vee \neg B$. *Therefore,* $A \leftrightarrow B$ *entails* $A \vee \neg B$. *Alternatively, we can also use the truth table to show the entailment.*

6. (4 pts) **True** or **False**: If C_1 and C_2 are **clause**s in propositional logic and all the literals in C_1 are contained in C_2 then C_1 entails C_2

   True. *If there are* $k \geq 0$ *additional literals in* C_2 *that are not in* C_1, *we can write* $C_2 = C_1 \vee L_1 \vee \ldots \vee L_k$. *Therefore, any interpretation that satisfies* C_1 *also satisfies* C_2 *regardless of truth values of* $L_1, \ldots, L_k$.

7. (3 pts) **True** or **False**: If we can successfully apply resolution to a propositional **clause** C and a copy of itself, it must be logically equivalent to $True$.

   True *Two clauses can be resolved if one contains a literal, for example, A and the other contains its negation $\neg A$. Since we consider resolving two copies of C, it should be that C contains both A and $\neg A$. Therefore, $C = A \vee \neg A \vee \ldots = True \vee \ldots = True$.*

8. (3 pts) **True** or **False**: Proof by resolution refutation in propositional logic cannot, in general, decide whether a sentence is **not** entailed by a KB.

   False*. Proof by resolution refutation is complete and decidable. That is, if the sentence is entailed, resolution refutation always finds a finite length of correct proof for the entailment. If the sentence is not entailed, it either proves false or cannot be applied any more after a finite number of steps.*

2 First-Order Logic (40 points)

Hint: Make sure you understand the definition of **clause** in first-order logic.

1. (6 pts) "Every cat loves its mother or father" can be translated as:

   (a) $\forall x.\neg Cat(x) \vee Loves(x, mother(x)) \vee Loves(x, father(x))$

   (b) $\forall x.Cat(x) \rightarrow Loves(x, mother(x) \vee father(x))$

   (c) $\forall x.Cat(x) \wedge (Loves(x, mother(x)) \vee Loves(x, father(x)))$

   Circle all the true choices.

   Only (a) is true.

2. (6 pts) "Every dog who loves one of its brothers is happy." can be translated as:

   (a) $\forall x.Dog(x) \wedge (\exists y.Brother(y, x) \wedge Loves(x, y)) \rightarrow Happy(x)$

   (b) $\forall x.(\exists y.Dog(x) \wedge Brother(y, x) \wedge Loves(x, y)) \rightarrow Happy(x)$

   (c) $\forall y.\forall x.Dog(x) \wedge Brother(y, x) \wedge Loves(x, y) \rightarrow Happy(x)$

   Circle all the true choices.

   (a), (b), (c) are all true.

3. (8 pts) Given the following KB:
   $\forall x.P(x) \rightarrow P(f(x))$
   $\neg P(f(f(A)))$
   we can prove that:

   (a) $P(f(A))$

   (b) $P(A)$

   (c) $\neg P(f(A))$

   (d) $\neg P(A)$

   Circle all the true choices. Show the resolution proof for one of your true choices below (we have given you more than enough lines).

   (c) and (d) are true.

Step	Reason	Clause	Unifier
1	*KB*	$\neg P(x_1) \vee P(f(x_1))$	
2	*KB*	$\neg P(f(f(A)))$	
3	*negate conclusion*	$P(f(A))$	
4	*1,3*	$P(f(f(A)))$	$\{x_1/f(A)\}$
5	*2,4*	*false*	

4. (3 pts) What is the clause form of $\forall y.\,(\forall x.P(x,y)) \rightarrow Q(y)$?

   $\neg P(f(y),y) \vee Q(y)$

5. (3 pts) What is the result of applying resolution to $\neg Q(x,f(x)) \vee Q(A,x)$ and $Q(w,A)$, also give the unifier (note that A is a constant) .

   There is no unifier. We cannot unify $f(x)$ and A.

6. (5 pts) Show the unifier and the result of resolution for:
   $P(x,f(x)) \vee \neg P(g(y),y)$
   $P(g(f(A)),f(w)) \vee \neg P(f(w),w)$

   Using the unifier $\{w/A, y/f(w)\}$, the result is $P(x,f(x)) \vee \neg P(f(A),A)$.

7. (3 pts) **True** or **False**: If we can successfully apply resolution to a first-order **clause** C and a copy of itself, it must be logically equivalent to $True$.

   False. For example, the clause $P(x) \vee \neg P(y)$ can be resolved with a copy of itself using the unifier $\{y/x\}$, but it is not true under every interpretation.

8. (3 pts) **True** or **False**: Binary resolution is the only inference rule that you need to prove that a first-order sentence follows from a KB.

   False. Factoring is needed in some cases.

9. (3 pts) **True** or **False**: Proof by resolution refutation in first-order logic cannot, in general, decide whether a sentence is **not** entailed by a KB.

   True. Proof by resolution refutation in first-order logic is sound, complete, but only semi-decidable.

3 Planning in FOL (35 points)

Consider a delivery robot operating in a building with multiple floors and an elevator. We want to formulate a planning problem in FOL for the robot to deliver items in the building. Write the axioms that you will need to represent the problem. Here are some basic facts that describe the situation:

- Robbie, the robot, starts out on floor 1 and has Coffee.
- Fred is located on floor 2; he doesn't ever leave the floor.
- If Robbie is on the same floor as a person, Robbie can find that person.
- If Robbie has an item and Robbie can find a person, then Robbie can deliver the item to the person, who will then have the item.
- If Robbie is not on the same floor as the person, he can take the elevator to the floor where the person is.

Now, we need to write axioms.

- Use two actions $Deliver(item, person)$ (deliver item to person) and $Elevate(floor_1, floor_2)$ (take elevator from $floor_1$ to $floor_2$). You should make "reasonable" assumptions about these actions: don't take the elevator from one floor to itself, don't deliver something that the person already has, etc.
- Use the following predicates
  - $OnFloor(entity, floor, s)$, where an entity is a person or robot, indicating the entity is on that floor.
  - $CanFind(person, s)$, indicating that the robot can find the person, and
  - $Have(entity, item, s)$, indicating that a person or robot has an item.

1. (6 pts) Write a set of axioms to describe the initial situation.

   Call the initial situation $S0$ (a constant). Then:

   $$OnFloor(Robbie, Floor1, S0)$$

   $$OnFloor(Fred, Floor2, S0)$$

   $$Have(Robbie, Coffee, S0)$$

   There are other, situation independent facts, such as the fact that Fred is always on Floor2 and the definition of CanFind. However, these are not strictly-speaking part of describing the initial situation.

   $$\forall s.OnFloor(Fred, Floor2, s)$$

   $$\forall s.\forall f.\forall p.OnFloor(Robbie, f, s) \wedge OnFloor(p, f, s) \rightarrow CanFind(p, s)$$

2. (3 pts) Write the goal statement to construct a plan for Fred to have Coffee.

$$\exists s.Have(Fred, Coffee, s)$$

3. (8 points) Write the effect axiom for the *Deliver* action.

$$\forall s.\forall x.\forall p.Have(Robbie, x, s) \wedge CanFind(p, s) \rightarrow Have(p, x, Result(Deliver(x, p), s))$$

One can add a pre-condition that $\neg Have(p, x, s)$ *to avoid unnecessary deliveries.*

4. (9 points) Write any necessary frame axioms for the *Deliver* action. Explain why these are all the axioms you need.

We need to consider whether we need to specify an axiom for each of the predicates. $CanFind$ *is always defined by the definition above in terms of* $OnFloor$*, so we don't need to re-specify a frame axiom for it. But, we do need frame axioms for* $Have$ *and* $OnFloor$*.*

A delivery does not change anyone's floor.

$$\forall s, x, p, f.OnFloor(p, f, s) \rightarrow OnFloor(p, f, Result(Deliver(x, p), s)$$

Only the delivered object changes hands, all other $Have$ *assertions remain true.*

$$\forall p, p_1, x, x_1, s.Have(p_1, x_1, s) \wedge (\neg(p = p_1) \vee \neg(x = x_1)) \rightarrow Have(p_1, x_1, Result(Deliver(p, x), s))$$

5. (5 pts) Explain the role of Green's trick in getting the answer from the proof (if we were to do it).

It allows us to answer existential questions concretely, by keeping track of the substitutions for the variables in the goal that lead to a contradiction during resolution refutation.

6. (4 pts) Write the logical term that will be the result of the proof (for Fred to Have Coffee) and which shows us the plan. You don't have to do the proof to answer this question.

$$Result(Deliver(Fred, Coffee), Result(Elevate(Floor1, Floor2), S0))$$