York University
CSE 2001 – Unit 1 First Order Logic
Instructor: Jeff Edmonds

Read Jeff's notes. Read the book. Go to class. Ask lots of question. Study the slides. Work hard on solving these questions on your own. Talk to your friends about it. Talk to Jeff about it. Only after this should you read the posted solutions. Study the solutions. Understand the solutions. Memorize the solutions. The questions on the tests will be different. But the answers will be surprisingly close.

1. For each prove whether true or not when each variable is a real value. Be sure to play the correct game as to who is providing what value.

   1) $\forall x\ \exists y\ x + y = 5$ 2) $\exists y\ \forall x\ x + y = 5$
   3) $\forall x\ \exists y\ x \cdot y = 5$ 4) $\forall x\ \exists y\ x \cdot y = 0$
   5) $[\forall x\ \exists y\ P(x, y)] \Rightarrow [\exists y\ \forall x\ P(x, y)]$
   6) $[\forall x\ \exists y\ P(x, y)] \Leftarrow [\exists y\ \forall x\ P(x, y)]$
   7) $\forall a\ \exists y\ \forall x\ x \cdot (y + a) = 0$
   8) $\exists a\ \forall x\ \exists y\ [x = 0 \text{ or } x \cdot y = 5]$

2. The game *Ping* has two rounds. Player-A goes first. Let m_1^A denote his first move. Player-B goes next. Let m_1^B denote his move. Then player-A goes m_2^A and player-B goes m_2^B. The relation $AWins(m_1^A, m_1^B, m_2^A, m_2^B)$ is true iff player-A wins with these moves.

   (a) Use universal and existential quantifiers to express the fact that player-A has a strategy in which he wins no matter what player-B does. Use $m_1^A, m_1^B, m_2^A, m_2^B$ as variables.
   (b) What steps are required in the Prover/Adversary technique to prove this statement?
   (c) What is the negation of the above statement in standard form?
   (d) What steps are required in the Prover/Adversary technique to prove this negated statement?

3. Let $Works(P, A, I)$ to true if algorithm A halts and correctly solves problem P on input instance I. Let $P = Halting$ be the Halting problem which takes a Java program I as input and tells you whether or not it halts on the empty string. Let $P = Sorting$ be the sorting problem which takes a list of numbers I as input and sorts them. For each part, explain the meaning of what you are doing and why you don't do it another way.

   Extra:
   Let $A_{insertionsort}$ be the sorting algorithm which we learned in class.
   Let A_{yes} be the algorithm that on input I ignores the input and simply halts and says "yes".
   Let A_∞ be the algorithm that on input I ignores the input and simply runs for ever.

   (a) Recall that a problem is *computable* if and only if there is an algorithm that halts and returns the correct solution on every valid input. Express in first order logic that *Sorting* is computable.
   (b) Express in first order logic that *Halting* is not computable.
   (c) Express in first order logic that there are uncomputable problems.
   (d) What does the following mean and either prove or disprove it: $\forall I, \exists A, Works(Halting, A, I)$. (Not simply by saying the same in words "For all I, exists A, $Works(Halting, A, I)$")
   (e) What does the following mean and either prove or disprove it $\forall A, \exists P, \forall I, Works(P, A, I)$. Hint: An algorithm A on an input I can either halt and give the correct answer, halt and give the wrong answer, or run for ever.

4. First Order Logic:
   Let P be some computable problem,
   k an integer,

A an algorithm (Java Program),
and I an input string.
Let $Lines(A, I) = k$ to be the statement that algorithm A has k actual lines of code (in the print out of the program) when run on input I.
Let $A(I) = L(I)$ to be the statement that A gives the correct answer for P on input I.
For each of the following first order logic statements, is it true and what are the ramifications/consequences of this with respect to solving P? i.e. why is it true/false.
The types of things that the first order logic will say are "Computable means that a fixed algorithm can get the right answer on each and every input" and "The number of lines of code does not change with the input."

(a) $\exists k,\ \exists A,\ \forall I,\ Lines(A, I) = k$ and $A(I) = P(I)$
(b) $\forall k,\ \exists A,\ \forall I,\ Lines(A, I) = k$
(c) $\forall A,\ \exists k,\ \forall I,\ Lines(A, I) = k$
(d) $\forall k,\ \exists A,\ \forall I,\ Lines(A, I) = k$ and $A(I) = P(I)$
(e) $\forall I,\ \exists A,\ A(I) = P(I)$
(f) $\forall A,\ \exists I,\ A(I) \neq P(I)$