CS206 Practice Homework #2

- ***Do not turn in this assignment. This is only meant for your practice.***
- *Be brief, complete and stick to what has been asked.*

1. In this question, we will talk about models of predicate logic formulae with only one binary predicate symbol L other than $=$. Let $\mathcal{M}_1 = (A_1, L_1)$ and $\mathcal{M}_2 = (A_2, L_2)$ be two models, where A_i denotes a universe of elements and $L_i \subseteq A_i \times A_i$ is an interpretation of the predicate L. Models $\mathcal{M}_1$ and $\mathcal{M}_2$ are said to be *isomorphic* iff there exists a *bijection* $h : A_1 \to A_2$ such that $L_2(h(x_1), h(y_1)) = L_1(x_1, y_1)$ for all $x_1, y_1 \in A_1$.

   (a) Give a predicate logic sentence ϕ such that every model $\mathcal{M}$ of ϕ is isomorphic to $(\mathbf{N}, <)$, i.e. the set of natural numbers with the usual less-than binary predicate.

   (b) Give a predicate logic sentence ψ such that every model of ψ is isomorphic to a rooted infinite binary tree in which every path is of infinite length. Clearly, such a binary tree qualifies to be a model $\mathcal{M}_1 = (A_1, L_1)$, where A_1 is the set of nodes of the tree and $L_1(x, y)$ is true iff y is a child of x.

   (c) Is it possible to have a predicate logic sentence ϕ such that every model of ϕ is isomorphic to $(\Re, <)$, i.e. the set of real numbers with the usual less-than binary predicate? You must justify your answer.

   (d) Let ϕ be a sentence in predicate logic and let $\mathcal{M}_1 = (A_1, L_1)$ and $\mathcal{M}_2 = (A_2, L_2)$ be two isomorphic models of ϕ with $A_1 \cap A_2 = \emptyset$. Is it possible to have a model $\mathcal{M}_3 = (A_3, L_3)$ of ϕ such that $\mathcal{M}_1$ and $\mathcal{M}_2$ are submodels? In other words, can we have $\mathcal{M}_3 \models \phi$ with $A_3 = A_1 \cup A_2$ and $L_3 \cap A_1 \times A_1 = L_1$ and $L_3 \cap A_2 \times A_2 = L_2$? You must justify your answer.

2. Prove the following sequents using natural deduction for predicate logic. Remember that a proof is simply a transformation of sets of formulae according to given rules. You must not use any knowledge about the semantics of the formulae in going from one step of the proof to the next.

   (a) $\forall x\, f(f(f(x))) = f(f(x)),\ \forall x \forall y\, ((y = f(x)) \to (f(y) = x)) \vdash \forall x (x = f(x))$

   (b) $\forall x\, P(a, x, x),\quad \forall x \forall y \forall z\, (P(x, y, z) \to P(f(x), y, f(z))) \vdash P(f(a), a, f(a))$ (Problem 13(a) of textbook)

   (c) $\forall x\, (\neg(x = f(x)) \wedge (f(f(x)) = x)),\ \exists x \exists y \exists z\, (\neg(x = y) \wedge \neg(y = z) \wedge \neg(x = z)) \vdash \exists x \exists y\, (\neg(y = f(x)) \wedge \neg(x = f(y)))$

   (d) $\exists x \exists y\, (H(x, y) \vee H(y, x)),\ \neg \exists x\, H(x, x) \vdash \exists x \exists y\, \neg(x = y)$ (Problem 11(c) of textbook)

   (e) $\forall x \forall y\, (\neg(x = y) \to (P(x, y) \vee P(y, x))),\ \forall x \forall y \forall z\, (P(x, y) \wedge P(y, z) \to P(x, z)),\ \forall x\, \neg P(x, x) \vdash \forall x \forall y\, \neg(P(x, y) \wedge P(y, x))$

3. For each of the following predicate logic sentences, do the following: (i) convert the sentence to Skolem Normal Form while keeping the arities of Skolem functions to the minimum possible, (ii) construct the corresponding Herbrand universe, and (iii) prove the unsatisfiability of the sentence by showing the propositional unsatisfiability of a finite number of ground clauses (Herbrand's Theorem).

   (a) $(\forall x \exists y\, (P(x, y) \to \neg P(y, x))) \wedge (\exists x \forall y\, (P(x, y) \wedge P(y, x)))$

(b) $(\forall y \forall z\, (\exists x(R(y,x) \wedge R(z,x))) \rightarrow\ R(y,z)) \wedge (\exists y \forall x\, R(x,y)) \wedge (\exists y \exists z\, \neg R(y,z))$

(c) $(\forall x\, (\neg(x = f(x)) \wedge f(f(x) = x))) \wedge\ (\exists y \forall z \neg(y = f(z)))$

4. (a) Let P and Q be unary predicates. Is it possible to have a predicate logic sentence ϕ such that (i) for every model satisfying ϕ, the cardinality of the set of elements for which P evaluates to true is equal to the cardinality of the set for which Q evaluates to true, and (ii) every model in which the above cardinalities are equal satisfies ϕ?

   (b) Consider a unary predicate R.

      i. Is it possible to have a predicate logic sentence ϕ such that (i) for every model satisfying ϕ, there are only a finite number of elements for which P evaluates to true, and (ii) every model which has finitely many elements for which P evaluates to true, also satisfies ϕ?

      ii. Is it possible to have a predicate logic sentence ϕ such that (i) for every model satisfying ϕ, there are countably infinite elements in its universe and countably infinite of them satisfy P, and (ii) every model that has countably infinite elements and countably infinite of them satisfy P, also satisfies ϕ?

   In all questions above, you must either provide the predicate logic sentence along with justification for why the sentence satisfies the required conditions, or you must prove that such a sentence cannot exist.

5. Two processes P_1 and P_2, as described below, are run concurrently on a uniprocessor computer system.

Process P1:	Process P2:
repeat forever	repeat forever
i11: x := (1 + y) mod 2;	i21: y := (1 + x) mod 2;
i12: y := (x + 1) mod 2;	i22: x := (y + 1) mod 2;

   In the above description, variables `x` and `y` are of type `natural number`. The symbol `:=` refers to an assignment operation and `mod` refers to the usual remaindering operation when dividing by integers. Label i_{kl} refers to the l^{th} instruction of the k^{th} process. Each such instruction is assumed to be *atomic*, i.e., once it starts executing on the processor, it cannot be terminated until it has completed execution, and the variable to which an assignment is made in the statement has actually assumed its new value.

   Variables `x` and `y` are shared by both the processes, i.e., both processes read from and write to the same memory locations when referring to the same variable. However, since there is a single processor, only one of the instructions can be executing at any given time. The execution of the instructions can, however, be interleaved in all possible ways, as long as an i_{k1} instruction is executed between every two successive executions of i_{k2} and vice versa. Thus, the following is an allowed execution sequence of instructions: $i_{11}, i_{12}, i_{11}, i_{21}, i12, i_{11}, i_{22}, \ldots$.

   (a) We wish to describe the evolution of the above system using a Kripke structure, where each state of the Kripke structure is labeled by the values of the variables x and y. Note that the definition of a Kripke structure does not prohibit labeling multiple states with the same labels. Draw a Kripke structure describing the behaviour of the above system, assuming that both x and y are initialized (prior to the start of execution of P_1 and P_2) to 1. You must label each state of your Kripke structure with a tuple (x, y) giving the values of x and y in that state.

   (b) Using atomic propositions q_1 denoting $(x = 0)$ and q_2 denoting $(y = 0)$, express the following properties as CTL formulae:

i. Whenever x becomes 0, the system evolves to ensure that y increases to 1 at some time in the future (excluding the current time instant), and then x changes to a value other than 0 at some time instant after that.

ii. Whenever x becomes 0, there is at least one way for the system to evolve such that x stays at 0 while the value of y changes twice.

(c) Using the CTL model checking algorithm described in class, check whether the above properties hold at the initial state of your Kripke structure. You must clearly show the labeling of states with subformulae.

6. Assuming that p, q and r are atomic propositions, express the following properties as CTL formulae. In case you feel that a particular property cannot be expressed using CTL, give an informal justification for your conclusion.

   (a) Every occurrence of p is followed by at least one occurrence of q.

   (b) There is at least one path in which there are no two consecutive states in which p is true.

   (c) Whenever r becomes true, it stays true until a state is reached from which there exists a path on which p is always true.

   (d) There is no path on which p is true infinitely often and q is true finitely often.

   (e) On every path, q occurs infinitely many times only if p occurs finitely many times.