Logic in Computer Science Midterm

1. **Induction and Recursion**

   Consider augmenting the alphabet of propositional logic with two new symbols: $\top$ and $\bot$, where $\top$ is a symbol that can be used wherever a propositional symbol can be used but always evaluates to true and $\bot$ is a similar symbol except that it always evaluates to false.

   (a) Give an inductive definition of propositional formulas including $\top$ and $\bot$. You may restrict the usual set of five Boolean connectives to any complete set. The formulas should be freely generated (but you don't have to prove that they are).

   Answer: (10 points)

   Let A be the alphabet of propositional logic with the addition of $\top$ and $\bot$.

   - U = the set of all expressions over A.
   - B = the set of expressions consisting of a single propositional symbol, $\top$, or $\bot$.
   - F = the set of formula-building operations:
     - $\mathcal{E}_{\neg}(\alpha) = (\neg\alpha)$
     - $\mathcal{E}_{\wedge}(\alpha, \beta) = (\alpha \wedge \beta)$
     - $\mathcal{E}_{\vee}(\alpha, \beta) = (\alpha \vee \beta)$
     - $\mathcal{E}_{\rightarrow}(\alpha, \beta) = (\alpha \rightarrow \beta)$
     - $\mathcal{E}_{\leftrightarrow}(\alpha, \beta) = (\alpha \leftrightarrow \beta)$

   (b) Let W_{tf} be the set of propositional formulas defined inductively in part (a). Define a recursive function *simp* which maps formulas in W_{tf} to a logically equivalent formula in $W \cup \{\top, \bot\}$, where W is the subset of formulas in W_{tf} that do not contain $\top$ and $\bot$ (i.e. *simp* either returns a formula that does not contain $\top$ or $\bot$ or it returns $\top$ or $\bot$).

   Answer: (10 points)

   For $\alpha \in B$, $simp(\alpha) = \alpha$.

   Now, suppose that γ is an expression built using one of the formula-building operations from α and (possibly) β. We must define $simp(\gamma)$ in terms of $simp(\alpha)$ and $simp(\beta)$.

   - $simp(\neg\alpha) = \begin{cases} \bot & \text{if } simp(\alpha) = \top \\ \top & \text{if } simp(\alpha) = \bot \\ (\neg simp(\alpha)) & \text{otherwise} \end{cases}$

- $simp(\alpha \wedge \beta) = \begin{cases} \bot & \text{if } simp(\alpha) = \bot \text{ or } simp(\beta) = \bot \\ simp(\alpha) & \text{if } simp(\beta) = \top \\ simp(\beta) & \text{if } simp(\alpha) = \top \\ (simp(\alpha) \wedge simp(\beta)) & \text{otherwise} \end{cases}$
- $simp(\alpha \vee \beta) = \begin{cases} \top & \text{if } simp(\alpha) = \top \text{ or } simp(\beta) = \top \\ simp(\alpha) & \text{if } simp(\beta) = \bot \\ simp(\beta) & \text{if } simp(\alpha) = \bot \\ (simp(\alpha) \vee simp(\beta)) & \text{otherwise} \end{cases}$
- $simp(\alpha \rightarrow \beta) = \begin{cases} \top & \text{if } simp(\alpha) = \bot \text{ or } simp(\beta) = \top \\ simp(\neg\alpha) & \text{if } simp(\beta) = \bot \\ simp(\beta) & \text{if } simp(\alpha) = \top \\ (simp(\alpha) \rightarrow simp(\beta)) & \text{otherwise} \end{cases}$
- $simp(\alpha \leftrightarrow \beta) = \begin{cases} simp(\alpha) & \text{if } simp(\beta) = \top \\ simp(\beta) & \text{if } simp(\alpha) = \top \\ simp(\neg\alpha) & \text{if } simp(\beta) = \bot \\ simp(\neg\beta) & \text{if } simp(\alpha) = \bot \\ (simp(\alpha) \leftrightarrow simp(\beta)) & \text{otherwise} \end{cases}$

2. **Decidability and Semi-Decidability**

(a) Prove that the union of two effectively enumerable sets is again effectively enumerable.

Answer: (10 points)

Suppose A and B are two effectively enumerable sets. Then there exist effective procedures P_A and P_B which list all elements of A and B respectively. However, these may run forever.

To list all elements of $A \cup B$, we must run P_A and P_B in an interleaved fashion. In other words, run P_A until it lists one element of A, then switch to P_B and run it until it lists one element of B, then switch back to P_A for another element of A and back to P_B for another element of B and so on. In this way, if an element was listed as the n^{th} element of either A or B, it will appear as element $2n$ or $2n-1$ in the interleaved list. Since this is an effective procedure for listing exactly the elements of $A \cup B$, $A \cup B$ is effectively enumerable.

(b) Prove that the intersection of two effectively enumerable sets is again effectively enumerable.

Answer: (10 points)

Suppose A and B are two effectively enumerable sets. Then there exist semi-decision procedures P_A and P_B such that, given an element x of the universal set, they will output "yes" if x is in A or B respectively. They may run forever if x is not in A (respectively B).

A semi-decision procedure for $A \cap B$ is the following: given x, run P_A. If it succeeds, run P_B. If it also succeeds, output "yes". This clearly outputs "yes" if and only if $x \in A \cap B$. Thus $A \cap B$ is effectively enumerable.

3. **First-Order Logic: Proofs and Models**

Following are several instances of $\Gamma \vdash \phi$. For each instance, either prove that $\Gamma \vdash \phi$ (either give an actual formal deduction or prove that one exists) or give a model and variable assignment which satisfies Γ but not ϕ (showing that $\Gamma \not\models \phi$ and therefore, by soundness, $\Gamma \nvdash \phi$).

(a) $\emptyset \vdash \neg\forall y \exists x (Py \wedge \neg Px)$

Answer: (5 points)

By quantifier manipulation (see p. 160), $\neg\forall y \exists x (Py \wedge \neg Px)$ is logically equivalent to $\neg(\forall y\, Py \wedge \exists x \neg Px)$, so by completeness, it suffices to show:

$$\vdash \neg(\forall y\, Py \wedge \exists x \neg Px).$$

By Boolean reasoning, it is sufficient to show

$$\vdash \forall y\, Py \rightarrow \neg\exists x \neg Px,$$

which is equivalent (substituting quantifiers) to

$$\vdash \forall y\, Py \rightarrow \forall x\, Px.$$

By the deduction theorem, it is sufficient to show

$$\forall y\, Py \vdash \forall x\, Px,$$

but this follows easily by alphabetic variants.

(b) $\forall x (x = a) \vdash \forall y \forall z (y = z)$

Answer: (5 points)

1.	$y = a$	assumption, axiom 2, MP
2.	$z = a$	assumption, axiom 2, MP
3.	$z = a \rightarrow a = z$	Eq2 (p. 127), (axiom 2, MP) twice
4.	$a = z$	MP(2,3)
5.	$y = a \rightarrow a = z \rightarrow y = z$	Eq3 (p. 128), (axiom 2, MP) three times
6.	$y = z$	MP(1,4,5)
7.	$\forall y \forall z (y = z)$	generalization

(c) $\forall x\,(Px \to \neg Qx) \vdash (\exists y\,Qy \to \exists z\,\neg Pz)$

Answer: (5 points)

1.	$\forall x\,(Px \to \neg Qx)$	assumption
2.	$\forall x\,Px \to \forall x \neg Qx$	axiom 3, MP with 1
3.	$\neg\forall x\,\neg Qx \to \neg\forall x\,Px$	Boolean equivalent of 2 (contrapositive)
4.	$\exists x\,Qx \to \exists x\,\neg Px$	substituting $\exists$ for $\forall$
5.	$\exists y\,Qy \to \exists z\,\neg Pz$	alphabetical variant

(d) $\forall x\,\forall y\,\forall z\,(Pxy \to Pyz \to Pxz) \vdash \forall x\,\forall y\,(Pxy \to Pyx)$

Answer: (5 points)

Consider the model M with $dom(M) = \{a, b\}$ and $P^M = \{(a, b)\}$. P^M is (trivially) transitive, but is not symmetric, so $\forall x\forall y\forall z\,(Pxy \to Pyz \to Pxz) \not\vdash \forall x\,\forall y\,(Pxy \to Pyx)$.

4. **Homomorphisms and Definability**

(a) Show that if A is a substructure of B and $\models_A \phi[s]$ for some quantifier-free formula ϕ and variable assignment s, then $\models_B \phi[s]$.

Answer: (5 points)

If A is a substructure of B, then the identity function i is an embedding of A into B. By the homomorphism theorem, it follows that for any quantifier-free formula ϕ and variable assignment s, $\models_A \phi[s]$ iff $\models_B \phi[i \circ s]$. But i is the identity function, so we have $\models_A \phi[s]$ iff $\models_B \phi[s]$

(b) A formula is *existential* if it is of the form $\exists x_1 \cdots \exists x_n\,\theta$, where θ is quantifier-free. Show that if A is a substructure of B and $\models_A \phi[s]$ for some existential formula ϕ and variable assignment s, then $\models_B \phi[s]$.

Answer: (5 points)

	$\models_A \exists x_1 \cdots \exists x_n\,\theta[s]$		
$\Rightarrow$	$\models_A \theta[s(x_1\|d_1)\cdots(x_n\|d_n)]$ for some $d_1, \ldots, d_n \in dom(A)$	by definition of $\exists$
$\Rightarrow$	$\models_B \theta[s(x_1\|d_1)\cdots(x_n\|d_n)]$ for some $d_1, \ldots, d_n \in dom(A)$	by part (a)
$\Rightarrow$	$\models_B \theta[s(x_1\|d_1)\cdots(x_n\|d_n)]$ for some $d_1, \ldots, d_n \in dom(B)$	since $dom(A) \subseteq dom(B)$
$\Rightarrow$	$\models_B \exists x_1 \cdots \exists x_n\,\theta[s]$	by definition of $\exists$

(c) Let Σ be a signature with equality and a single binary predicate symbol: $<$. Let M be a Σ-model whose domain is the set of positive integers and which interprets $<$ as follows: $x <^M y$ iff x divides y. Show that the set of prime numbers is definable in M.

Answer: (10 points)

Assuming we have equality (an acceptable assumption), the set of prime numbers is the set of numbers other than 1 which are divisible only by 1 and themselves:

$$\forall y\,(\forall z\,(y < z) \rightarrow (x \neq y \wedge \forall z\,(z < x \rightarrow (z = y \vee z = x)))).$$

If we do not assume we have equality, it can be defined as $x < y \wedge y < x$, which yields the following definition of the primes:

$$\forall y(\forall z(y < z) \rightarrow (\neg(x < y \wedge y < x) \wedge \forall z(z < x \rightarrow ((z < y \wedge y < z) \vee (z < x \wedge x < z))))).$$

5. **First-Order Compactness**

(a) Suppose that Γ is a set of sentences and τ is a sentence. Show that if $Mod\Gamma = Mod\tau$, then there is a finite $\Gamma_0 \subseteq \Gamma$ such that $Mod\Gamma_0 = Mod\tau$.

Answer: (10 points)

If $\models_M \Gamma$, then since $Mod\Gamma = Mod\tau$, it follows that $\models_M \tau$. Thus, $\Gamma \models \tau$. By compactness, we know there exists a finite $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \models \tau$, from which it follows that $Mod\Gamma_0 \subseteq Mod\tau$. But since $\Gamma_0 \subseteq \Gamma$, we also have $Mod\Gamma \subseteq Mod\Gamma_0$. Thus, $Mod\Gamma = Mod\tau$ and $Mod\Gamma \subseteq Mod\Gamma_0 \subseteq Mod\tau$, from which it follows that $Mod\Gamma_0 = Mod\tau$.

(b) Recall that a class $\mathcal{K}$ of models is first-order definable iff $\mathcal{K} = Mod\tau$ for some sentence τ. Show that the class of all infinite groups is not first-order definable. You may use the fact that there are arbitrarily large finite groups.

Answer: (10 points)

Suppose the class of infinite groups is first-order definable. Then it is definable by a sentence τ. We also know that the class of infinite groups is defined by $\Gamma = G \cup \{\lambda_2, \lambda_3, \ldots\}$ where G contains the group axioms and λ_i is the sentence stating "there are at least i objects". Since both Γ and τ define the same class, we have $Mod\Gamma = Mod\tau$. By part (a), it follows that the class of infinite groups can also be defined by some finite subset Γ_0 of Γ. But any such Γ_0 contains a largest λ_k and thus $Mod\Gamma_0$ must include finite groups of size k or larger. Clearly $Mod\Gamma_0$ is not the class of infinite groups, a contradiction.

Alternative solution: Suppose the class of infinite groups is first-order definable. Then it is definable by a sentence τ. Let G be a sentence which contains the group axioms. Then $G \wedge \neg\tau$ defines the finite groups (all things which are groups and which are not infinite groups). But since there are arbitrarily large models of $G \wedge \neg\tau$ (there are arbitrarily large finite groups), there must be an infinite group $g \in Mod(G \wedge \neg\tau)$. But then we would have $\models_g \tau$ and $\models_g \neg\tau$, a contradiction.