Assignment 1 *(Propositional Logic)* (12 + 6 = 18 points)

Let Π be a set of propositional variables with $P \in \Pi$, $Q \in \Pi$. For every propositional formula F over Π, let $\phi(F)$ be the formula that one obtains from F by replacing every occurrence of P by $(P \vee Q)$. (For instance, if $F = ((R \vee \neg P) \wedge (Q \vee P))$, then $\phi(F) = ((R \vee \neg(P \vee Q)) \wedge (Q \vee (P \vee Q)))$, and if $F = R$, then $\phi(F) = R$.

Part (a) Prove: If $\phi(F)$ is satisfiable, then F is satisfiable.

(Note: Somewhere in the proof you need an induction over the structure of formulas. It is sufficient if you check the base cases and $\wedge$, $\vee$, and $\neg$. The other boolean connectives can be handled analogously; you may omit them.)

Part (b) Refute: If $\phi(F)$ is valid, then F is valid.

Assignment 2 *(First-Order Logic, Semantics)* (10 points)

Let $\Sigma = (\Omega, \emptyset)$ with $\Omega = \{f/1,\ c/0\}$. Give a Σ-model $\mathcal{A}$ of

$$\neg f(c) \approx c \quad \wedge \quad \forall x\,(f(f(x)) \approx x$$

with $U_{\mathcal{A}} = \{1, 2, 3\}$.

Assignment 3 *(Ordered Resolution)* (9 + 5 = 14 points)

Let $\Sigma = (\{f/1,\ g/1,\ h/1,\ b/0,\ c/0\},\ \{P/2,\ Q/1,\ R/2\})$; let N be the following set of clauses over Σ:

$$P(x, f(x)) \ \vee\ P(x, x) \qquad (1)$$

$$\neg P(h(z), x) \ \vee\ \neg P(y, f(f(x))) \ \vee\ \neg Q(x) \ \vee\ Q(f(x)) \qquad (2)$$

$$\neg Q(h(f(x))) \ \vee\ R(h(b), y) \qquad (3)$$

$$\neg R(y, g(c)) \ \vee\ Q(g(x)) \qquad (4)$$

$$\neg Q(h(y)) \qquad (5)$$

Part (a) Suppose that the atom ordering $\succ$ is a Knuth-Bendix ordering with weight 1 for all function and predicate symbols and variables and the precedence $P > Q > R > f > g > h > b > c$ and that the selection function *sel* selects no literals. Compute all $\mathrm{Res}^{\succ}_{sel}$ inferences between the clauses (1)–(5). (Do not compute inferences between derived clauses. Do not compute any inferences that violate the restrictions of the calculus.)

Part (b) One of the clauses (1)–(5) is redundant with respect to the others. Which one? Why? Give a brief explanation.

Assignment 4 *(Tableaux)* (7 + 7 = 14 points)

Check whether the following propositional formulas are valid or not using semantic tableaux. Give a brief explanation. (Use exactly the expansion rules given in the lecture. Do not use shortcuts; do not convert the formulas to NNF, CNF, or DNF.)

Part (a)

$$(P \rightarrow Q) \rightarrow ((Q \rightarrow R) \rightarrow (P \rightarrow R))$$

Part (b)

$$(R \wedge (R \rightarrow P)) \rightarrow (P \wedge \neg Q)$$

Assignment 5 *(Term Orderings)* (12 points)

Decide for each of the following statements whether they are true or false:

(1) $f(g(x)) \succ f(x)$ in every simplification ordering $\succ$.

(2) $f(f(x)) \succ f(y)$ in every simplification ordering $\succ$.

(3) If $\succ$ is an LPO, then $f(x) \succ g(x)$ implies $f(x) \succ g(g(x))$.

(4) If $\succ$ is a KBO, then $f(x) \succ g(x)$ implies $f(x) \succ g(g(x))$.

(5) If $\succ$ is an LPO, then $h(f(x), y, y) \succ h(x, z, z)$.

(6) If $\succ$ is a KBO, then $h(f(x), f(y), z) \succ h(x, f(z), y)$.

(7) There is a reduction ordering $\succ$ such that $f(x) \succ g(f(x))$.

(8) There is a reduction ordering $\succ$ such that $f(f(x)) \succ f(g(f(x)))$.

(Note on grading: A yes/no answer is sufficient; you do not have to give any explanations. However, you need at least five correct answers to get any points for assignment 5. Missing answers count like false answers.)

Assignment 6 *(Dependency Pairs)* (3 + 4 + 5 = 12 points)

Let $\Sigma = (\Omega, \emptyset)$ with $\Omega = \{f/1,\ b/0, c/0\}$.

Part (a) Find a terminating term rewrite system over Σ that consists of exactly one rewrite rule and has no dependency pairs.

Part (b) Find a terminating term rewrite system over Σ that consists of exactly one rewrite rule and has exactly one dependency pair. Give the rule and the dependency pair.

Part (c) Find a terminating term rewrite system over Σ that consists of exactly one rewrite rule and has exactly two dependency pairs. Give the rule and the dependency pairs.