Problem 1 *(Algebras)* (10 points)

Let N be a set of (universally quantified) clauses without equality. Let Π_1 be the set of all predicate symbols that occur in positive literals in N, let Π_2 be the set of all predicate symbols that occur in negative literals in N, let $\Pi_0 = (\Pi_1 \setminus \Pi_2) \cup (\Pi_2 \setminus \Pi_1)$, and let N_0 be the set of all clauses of N in which no predicate symbol from Π_0 occurs. Prove: N is satisfiable if and only if N_0 is satisfiable.

Problem 2 *(CNF transformation)* (10 points)

Transform the following formula into CNF using Miniscoping:

$$\forall x\, \neg\exists y\, \forall z\, (p(z) \wedge r(x,y)) \leftrightarrow \exists x\, r(x,x)$$

Problem 3 *(Resolution, candidate interpretations)* (10 points)

Let $\Sigma = (\Omega, \Pi)$ with $\Omega = \{a, b, c\}$ and $\Pi = \{p, q, r\}$ and let $\succ$ be a total and well-founded ordering on ground atoms such that $r(\dots) \succ q(\dots) \succ p(\dots)$. Let N be a set of (universally quantified) clauses without equality that is saturated up to redundancy w. r. t. the resolution calculus $Res_S^{\succ}$ and does not contain the empty clause. Suppose that $G_\Sigma(N)$ contains (among others) the following five clauses:

$$\begin{aligned}
C_1 &= \neg r(a) \vee q(c) \\
C_2 &= \neg q(b) \vee q(b) \vee r(c) \\
C_3 &= p(a) \vee q(a) \vee r(a) \\
C_4 &= \neg p(b) \vee q(b) \\
C_5 &= p(a) \vee q(a) \vee q(a)
\end{aligned}$$

If we construct a candidate interpretation as in Sect. 2.11 of the lecture, then at most one of the five clauses C_1, C_2, C_3, C_4, C_5 can be productive. Which one? Explain briefly for each of the other four clauses why it cannot be productive.

Problem 4 *(Tableaux)* (12 points)

Prove the formula below via a closed AMGU tableau:

$$[\forall x\, \forall y\, (p(x) \vee r(x,y)) \wedge \forall x\, \forall y\, \exists z\, (\neg p(z) \vee r(x,y))] \rightarrow \exists x\, \forall y\, r(x,y)$$

Problem 5 *(Reduction orderings)* (10 + 8 = 18 points)

Let $\succ$ be a strict partial ordering on a set A. An element $s \in A$ is called successor of $t \in A$ with respect to A and $\succ$, if for all $u \in A$,

$$u \succ t \quad \text{if and only if} \quad u = s \;\text{ or }\; u \succ s.$$

An element $s \in A$ is called predecessor of $t \in A$ with respect to A and $\succ$, if for all $u \in A$,

$$u \prec t \quad \text{if and only if} \quad u = s \;\text{ or }\; u \prec s.$$

Part (a)

Prove: If the signature Σ contains at least one non-constant function symbol and $\succ$ is a reduction ordering over $\mathrm{T}_\Sigma(X)$ that is total on ground terms, then every ground term $t \in \mathrm{T}_\Sigma(\emptyset)$ has a successor w. r. t. $\mathrm{T}_\Sigma(\emptyset)$ and $\succ$.

Part (b)

Give an example of a signature Σ, a reduction ordering $\succ$ over $\mathrm{T}_\Sigma(X)$, and two ground term $s, t \in \mathrm{T}_\Sigma(\emptyset)$ such that $t \succ s$ but t does not have a predecessor w. r. t. $\mathrm{T}_\Sigma(\emptyset)$ and $\succ$.

Problem 6 *(Knuth-Bendix completion)* (10 points)

Apply the Knuth-Bendix procedure to the set of equations

$$\{\, f(f(x)) \approx g(x),\; f(a) \approx b \,\}$$

and transform it into a finite convergent term rewrite system; use the Knuth-Bendix ordering with weight 1 for all function symbols and variables and the precedence $g > f > a > b$.

Problem 7 *(Superposition)* (10 points)

Find a reduction ordering such that the clause set below is saturated w. r. t. superposition up to redundancy. Show that all possible inferred clauses are redundant. As usual, x and y are the only variables.

$$\begin{array}{rcrl}
C_1 & = & & \mathit{List}(\mathit{nil}) \approx \mathit{tt} \\
C_2 & = & \mathit{List}(x) \not\approx \mathit{tt} \;\vee & \mathit{List}(\mathit{cons}(x,y)) \approx \mathit{tt} \\
C_3 & = & & \mathit{singleton}(x) \approx \mathit{cons}(x, \mathit{nil}) \\
C_4 & = & \mathit{List}(x) \not\approx \mathit{tt} \;\vee & x \approx \mathit{nil} \;\vee\; \mathit{List}(\mathit{tail}(x)) \approx \mathit{tt} \\
C_5 & = & \mathit{List}(x) \not\approx \mathit{tt} \;\vee & \mathit{tail}(\mathit{cons}(y,x)) \approx x
\end{array}$$

Aufgabe 1 *(Algebren)* (10 Punkte)

Sei N eine Menge von (allquantifizierten) Klauseln ohne Gleichheit. Sei Π_1 die Menge aller Prädikatsymbole, die in positiven Literalen in N auftreten, sei Π_2 die Menge aller Prädikatsymbole, die in negativen Literalen in N auftreten, sei $\Pi_0 = (\Pi_1 \setminus \Pi_2) \cup (\Pi_2 \setminus \Pi_1)$, und sei N_0 die Menge aller Klauseln aus N, in denen kein Prädikatsymbol aus Π_0 vorkommt. Zeigen Sie: N ist genau dann erfüllbar, wenn N_0 erfüllbar ist.

Aufgabe 2 *(CNF-Transformation)* (10 Punkte)

Transformieren Sie die folgende Formel unter Verwendung von Miniscoping in Klauselnormalform:

$$\forall x\, \neg \exists y\, \forall z\, (p(z) \land r(x, y)) \leftrightarrow \exists x\, r(x, x)$$

Aufgabe 3 *(Resolution, Kandidateninterpretation)* (10 Punkte)

Sei $\Sigma = (\Omega, \Pi)$ mit $\Omega = \{a, b, c\}$ und $\Pi = \{p, q, r\}$, und sei $\succ$ eine totale und wohlfundierte Ordnung auf Grundatomen, so daß $r(\dots) \succ q(\dots) \succ p(\dots)$. Sei N eine Menge von (allquantifizierten) Klauseln ohne Gleichheit, die bezüglich des Resolutionskalküls $\mathit{Res}_S^{\succ}$ bis aus Redundanz saturiert ist und nicht die leere Klausel enthält. Nehmen wir an, daß $G_\Sigma(N)$ (unter anderem) die folgenden fünf Klauseln enthält:

$$\begin{aligned}
C_1 &= \neg r(a) \lor q(c) \\
C_2 &= \neg q(b) \lor q(b) \lor r(c) \\
C_3 &= p(a) \lor q(a) \lor r(a) \\
C_4 &= \neg p(b) \lor q(b) \\
C_5 &= p(a) \lor q(a) \lor q(a)
\end{aligned}$$

Falls wir wie in Abschnitt 2.11 der Vorlesung eine Kandidateninterpretation konstruieren, dann kann höchstens eine der fünf Klauseln C_1, C_2, C_3, C_4, C_5 produktiv sein. Welche? Erläutern Sie für jede der anderen vier Klauseln kurz, warum sie nicht produktiv sein kann.

Aufgabe 4 *(Tableaus)* (12 Punkte)

Beweisen Sie die folgende Formel mittels eines geschlossenen AMGU-Tableaus:

$$[\forall x\, \forall y\, (p(x) \lor r(x, y)) \land \forall x\, \forall y\, \exists z\, (\neg p(z) \lor r(x, y))] \rightarrow \exists x\, \forall y\, r(x, y)$$

Aufgabe 5 *(Reduktionsordnungen)* (10 + 8 = 18 Punkte)

Sei $\succ$ eine strikte partielle Ordnung auf einer Menge A. Ein Element $s \in A$ heißt Nachfolger von $t \in A$ bezüglich A und $\succ$, falls für alle $u \in A$,

$$u \succ t \quad \text{genau dann, wenn} \quad u = s \ \text{ oder } \ u \succ s.$$

Ein Element $s \in A$ heißt Vorgänger von $t \in A$ bezüglich A und $\succ$, falls für alle $u \in A$,

$$u \prec t \quad \text{genau dann, wenn} \quad u = s \ \text{ oder } \ u \prec s.$$

Part (a)

Zeigen Sie: Falls die Signatur Σ mindestens ein nicht-konstantes Funktionssymbol enthält und $\succ$ eine Reduktionsordnung über $\mathrm{T}_\Sigma(X)$ ist, die total auf Grundtermen ist, dann hat jeder Grundterm $t \in \mathrm{T}_\Sigma(\emptyset)$ einen Nachfolger bezüglich $\mathrm{T}_\Sigma(\emptyset)$ und $\succ$.

Part (b)

Geben Sie ein Beispiel für eine Signatur Σ, eine Reduktionsordnung $\succ$ über $\mathrm{T}_\Sigma(X)$ und zwei Grundterme $s, t \in \mathrm{T}_\Sigma(\emptyset)$, so daß $t \succ s$ gilt, aber t keinen Vorgänger bezüglich $\mathrm{T}_\Sigma(\emptyset)$ und $\succ$ besitzt.

Aufgabe 6 *(Knuth-Bendix-Vervollständigung)* (10 Punkte)

Wenden Sie die Knuth-Bendix-Prozedur auf die Gleichungsmenge

$$\{\, f(f(x)) \approx g(x),\ f(a) \approx b \,\}$$

an und transformieren Sie sie in ein endliches und konvergentes Termersetzungssystem; benutzen Sie die Knuth-Bendix-Ordnung mit Gewicht 1 für alle Funktionssymbole und Variablen und mit der Präzedenz $g > f > a > b$.

Aufgabe 7 *(Superposition)* (10 Punkte)

Finden Sie eine Reduktionsordnung, so daß die nachstehende Klauselmenge bezüglich Superposition bis auf Redundanz saturiert ist. Zeigen Sie, daß alle möglichen inferierten Klauseln redundant sind. Wie üblich sind x und y die einzigen Variablen.

$$\begin{array}{rcl}
C_1 & = & List(nil) \approx tt \\
C_2 & = & List(x) \not\approx tt \ \vee\ List(cons(x,y)) \approx tt \\
C_3 & = & singleton(x) \approx cons(x, nil) \\
C_4 & = & List(x) \not\approx tt \ \vee\ x \approx nil \ \vee\ List(tail(x)) \approx tt \\
C_5 & = & List(x) \not\approx tt \ \vee\ tail(cons(y,x)) \approx x
\end{array}$$