Aufgabe 1 Herbrand-Modelle

a) Gib ein Herbrand-Modell für $R(x, f(x)) \wedge \neg P(x)$ an

b) Erfüllt jedes Herbrand-Modell für $P(a) \wedge P(b) \wedge P(c)$ auch $\forall x\, P(x)$? Wie ist es mit kanonischen Modellen? Wie ist es mit beliebigen Modellen?

c) Sind Herbrand-Modelle immer abzählbar?

Aufgabe 2 Aussagen- und Prädikatenlogik

a) Gib eine quantorfreie Formel mit nur einer Variable an, die AL-erfüllbar, aber nicht FO-erfüllbar ist

b) Gibt es auch eine entsprechende Formel ohne Variablen?

c) Wenn eine beliebige FO-Formel nur nullstellige Prädikate verwendet, welche Beziehung besteht dann zwischen AL- und FO-Erfüllbarkeit bzw. -Gültigkeit?

Aufgabe 3 Folgerungen aus dem Satz von Herbrand

a) Ist Unerfüllbarkeit einer beliebigen FO-Formel entscheidbar bzw. semi-entscheidbar?

b) Ist Gültigkeit einer beliebigen FO-Formel entscheidbar bzw. semi-entscheidbar?

c) Ist Erfüllbarkeit einer beliebigen FO-Formel entscheidbar bzw. semi-entscheidbar?

d) Gibt es eine (erfüllbare) quantorfreie FO-Formeln, deren Modelle alle überabzählbar groß sind?

Aufgabe 4 Der Satz von Herbrand

a) Zeige, dass es für jedes $n \in \mathbb{N}$ eine quantorfreie Formel p_n gibt, sodass man für den Beweis der Unerfüllbarkeit über den Satz von Herbrand mindestens n Grundinstanzen benötigt.

b) Zeige, dass $\exists x \forall y (P(x) \to P(y))$ gültig ist (Beispiel aus dem Vortrag). Dazu reicht es zu zeigen, dass die Negation unerfüllbar ist (Negation, Skolem, Satz von Herbrand).

Aufgabe 5 Unification Algorithmus

Bestimme für die folgenden Paare von Termen jeweils den MGU, falls er existiert:

a) f(a)
f(b)

b) f(x,y)
f(y,x)

c) f(x,g(y))
f(f(z),w)

d) f(x,g(y))
f(y,x)

e) f(x, f(g(z), z))
f(f(y, a), x)

f) g(g(x, g(f(z), z)), g(h(u), z))
g(g(y, y), g(z, h(x)))

Aufgabe 6 Spezialisierung von Unifiern

Sei σ ein Unifier für S. Zeige, dass für alle Instanziierungen τ die Komposition `tsubst` $\tau \circ$ `tsubst` σ ebenfalls einen Unifier für S darstellt.

Aufgabe 7 Mögliche Fälle in `unify`

Finde für jeden der 7 Fälle in `unify` ein Argument `eqs`, sodass bei der Berechnung von `unify undefined eqs` der entsprechende Fall ausgeführt wird.

Aufgabe 8 Anti-Unification

a) Zeige, dass die allgemeiner-Relation auf Instanziierungen verwendet werden kann, um eine Verband-Struktur (Lattice) zu definieren, wobei Unification verwendet werden kann, um kleinste obere Schranken zu bestimmen.

b) Implementiere einen Algorithmus für "Anti-Unification", der größte untere Schranken bestimmt.