Assignment 1

Foundations of Logic Programming

October 12, 2012

1. Given the definitions of σ (a function from variables to expressions) and $\widehat{\sigma}$ (extension of σ to the function from expressions to expressions) from the lecture, prove that for each expression t:

$$t\widehat{\gamma\tau} = (t\widehat{\gamma})\widehat{\tau}$$

2. Show that composition of subsitutions is associative:

   $(\sigma\gamma)\tau = \sigma(\gamma\tau)$.

3. We say that a substitution γ is a *renaming* if γ is $1-1$ and *onto* mapping from its domain to itself (i.e. a *permutation* of the domain).

   Prove that for every renaming θ there exists only one substitution θ^{-1} such that $\theta\theta^{-1} = \theta^{-1}\theta = \epsilon$. Prove that θ^{-1} is a renaming of θ.

4. Consider two substitutions, $\theta = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ and $\eta = \{y_1 \mapsto s_1, \ldots, y_m \mapsto s_m\}$. Prove that the composition $\theta\eta$ equals the result of the following procedure:

   - remove from the sequence:
     $x_1 \mapsto t_1\eta, \ldots, x_n \mapsto t_n\eta, y_1 \mapsto s_1, \ldots, y_m \mapsto s_m$,
     the bindings $x_i \mapsto t_i\eta$, for which $x_i = t_i\eta$ and the bindings $y_j \mapsto s_j$ for which $y_j \in \{x_1, \ldots, x_n\}$.
   - form a substitution from the resulting sequence of bindings.

5. s is called an *instance* of t if $s = t\sigma$ for a substitution σ. Prove that s is a variant of t iff s is an instance of t and t is an instance of s.

6. In the lecture we have seen that θ is not more general than τ in the following examples:

   (a) $\theta = \{x \mapsto y\}, \tau = \{x \mapsto a\}$

   (b) $\theta = \{x \mapsto f(y, z)\}, \tau = \{x \mapsto f(a, a)\}$

   Try to modify the definitions of θ and\or τ so that $\theta \leq \tau$.

7. Prove **Renaming Lemma**: $\theta \leq \eta$ and $\eta \leq \theta$ iff there is γ a renaming of θ such that, $\eta = \theta\gamma$.