Advanced Type Systems
Homework #1

Instructor: Derek Dreyer

Assigned: January 4, 2006
Due: January 18, 2006

1 Anti-Substitution

Recall the typing rules for the simply-typed λ-calculus:

$$\frac{x : \tau \in \Gamma}{\Gamma \vdash x : \tau} \qquad \frac{}{\Gamma \vdash c : \mathbf{T}} \qquad \frac{\Gamma, x : \tau \vdash e : \tau'}{\Gamma \vdash \lambda x : \tau.\, e : \tau \to \tau'} \qquad \frac{\Gamma \vdash e_1 : \tau_2 \to \tau \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash e_1(e_2) : \tau}$$

In class, we reviewed the proof of the *substitution lemma*, which is needed in order to prove type soundness. One way of stating this lemma is as follows:

Suppose $\Gamma, x : \tau \vdash e' : \tau'$. Then, for all e, $\Gamma \vdash e : \tau$ implies $\Gamma \vdash e'[e/x] : \tau'$.

Part (a): Prove or disprove the converse of substitution, which I call *anti-substitution*. That is:

Suppose that, for all e, $\Gamma \vdash e : \tau$ implies $\Gamma \vdash e'[e/x] : \tau'$. Then, $\Gamma, x : \tau \vdash e' : \tau'$.

Part (b): Let's say we remove constants c from the language, along with the second typing rule above, but we leave the base type $\mathbf{T}$ in the language. Again, prove or disprove anti-substitution.

Note/Hint/Clarification: There is an easy solution to one part of this assignment that relies on the fact that well-typed programs in the simply-typed λ-calculus terminate. You get extra credit for pointing out this solution. However, the intention of this assignment is that you should not need to make use of any logical relations argument in your solution. Relying on termination counts as an implicit use of logical relations, so your primary solution should not rely on it.