CS 440: Programming Languages

Assignment: Lambda Calculus

Logistics and Submission

Please submit your solutions as a PDF (typed or *neatly* handwritten!) on Blackboard by the due date.

Abstract syntax trees

For each of the following lambda calculus expressions, (a) draw the corresponding abstract syntax tree – following the conventions used in lecture, and (b) circle all the *free* variables in the AST. (5 points each)

1. $\lambda x.x\ \lambda z.z\ x$
2. $(\lambda x.y)\ (\lambda y.x)$
3. $(\lambda a.\lambda b.a\ b)\ (\lambda x.x\ x)\ (\lambda y.b\ a)$

Normal form

For each of the following lambda calculus expressions, (a) indicate whether applicative-order evaluation will eventually find normal form, and (b) if normal form can be reached via either applicative- or normal-order evaluation, show all the steps (β/η-reductions or α-conversions) that lead to it, and normal form itself. If (b) is not possible, show at least three reductions and explain why further normalization is not possible. (5 points each)

4. $(\lambda x.\lambda w.w\ x)\ w\ \lambda z.y\ z$
5. $(\lambda x.y)\ ((\lambda w.w\ w\ w)\ (\lambda y.y\ y\ y))$
6. $(\lambda x.x\ y)\ (\lambda w.w\ w)$
7. $(\lambda x.\lambda y.x\ y\ x)\ (\lambda w.w)\ (\lambda z.z\ z)$

Alternative numbers

As alternatives to the Church numerals discussed in class, consider the following definitions:

$$
\begin{aligned}
\text{TRUE} &\equiv \lambda x.\lambda y.x \\
\text{FALSE} &\equiv \lambda x.\lambda y.y \\
0 &\equiv \lambda x.x \\
\text{INC} &\equiv \lambda n.\lambda x.(x\ \text{FALSE})\ n \\
1 &\equiv \text{INC}\ 0 \\
2 &\equiv \text{INC}\ 1 \\
&\vdots
\end{aligned}
$$

8. Based on the definitions above, define the DEC (decrement) function, such that DEC (INC n) $\equiv$ n, and demonstrate — by showing the necessary reductions — that DEC (DEC 2) $\equiv$ 0. (5 points)

9. Define the IS_ZERO function, which should evaluate to TRUE when called on 0 and FALSE otherwise. (5 points)