HW 11 – Lambda Calculus

CS 421 – Fall 2015

Revision 1.0

Assigned Wednesday, November 18, 2015
Due Wednesday, December 2, 2015, 23:59pm
Extension 48 hours (20% penalty)

1 Change Log

1.0 Initial Release.

2 Turn-In Procedure

Answer the problems below, save your work as a PDF (either scanned if handwritten or converted from a program), and hand in the PDF. Your file should be named `hw11-submission.pdf`.

3 Objectives and Background

The purpose of this HW is to test your understanding of:

- Alpha and beta conversion in the lambda calculus
- The consequences of different evaluation schemes
- experience answering non-programming written questions similar to those on the final

4 Problems

1. (15 pts) Prove that $\lambda x.x(\lambda z.zxz)$ is α-equivalent $\lambda z.z(\lambda x.xzx)$. You should label every use of α-conversion and congruence.

   Solution:
   By α-conversion
   $$\lambda x.x(\lambda z.zxz)\xrightarrow{\alpha}\lambda w.w(\lambda z.zwz).$$
   Because α-conversion implies α-equivalence, we have
   $$\lambda x.x(\lambda z.zxz)\overset{\alpha}{\sim}\lambda w.w(\lambda z.zwz).$$
   By α-conversion
   $$\lambda z.zwz\xrightarrow{\alpha}\lambda x.xwx$$
   and thus
   $$\lambda z.zwz\overset{\alpha}{\sim}\lambda x.xwx$$
   By congruence for application, we have
   $$w(\lambda z.zwz)\overset{\alpha}{\sim}w(\lambda x.xwx),$$

and by congreunce for abstraction, we have

$$\lambda w.w(\lambda z.zwz)\overset{\alpha}{\sim}\lambda w.w(\lambda x.xwx).$$

By transitivity, we then have

$$\lambda x.x(\lambda z.zxz)\overset{\alpha}{\sim}\lambda w.w(\lambda x.xwx).$$

By α-conversion,

$$\lambda w.w(\lambda x.xwx)\xrightarrow{\alpha}\lambda z.z(\lambda x.xzx).$$

Again, because α-conversion implies α-equivalence, we have

$$\lambda w.w(\lambda x.xwx)\overset{\alpha}{\sim}\lambda z.z(\lambda x.xzx).$$

Once more by transitivity of α-equivalence, we have

$$\lambda x.x(\lambda z.zxz)\overset{\alpha}{\sim}\lambda z.z(\lambda x.xzx)$$

as was to be shown.

2. (15 pts) Given the following term:

$$(\lambda x.x(\lambda y.xy))((\lambda u.u)(\lambda w.w))$$

reduce this term as much as possible using each of

a. eager evaluation

b. lazy evaluation

c. unrestricted $\alpha\beta$-reduction (*i.e.* by $\alpha\beta$ conversion that can be applied anywhere)

Label each step of reduction with the rule justifying it. You do not need to label uses of congruence, or break them out as separate steps, in this problem.

Solution:

Eager Evaluation

$$\begin{array}{l}(\lambda x.x(\lambda y.xy))((\lambda u.u)(\lambda w.w))\\ \xrightarrow{\beta}(\lambda x.x(\lambda y.xy))(\lambda w.w)\\ \xrightarrow{\beta}(\lambda w.w)(\lambda y.(\lambda w.w)y)\\ \xrightarrow{\beta}\lambda y.(\lambda w.w)y\end{array}$$

Lazy Evaluation

$$\begin{array}{l}(\lambda x.x(\lambda y.xy))((\lambda u.u)(\lambda w.w))\\ \xrightarrow{\beta}((\lambda u.u)(\lambda w.w))(\lambda y.((\lambda u.u)(\lambda w.w))y)\\ \xrightarrow{\beta}(\lambda w.w)(\lambda y.((\lambda u.u)(\lambda w.w))y)\\ \xrightarrow{\beta}\lambda y.((\lambda u.u)(\lambda w.w))y\end{array}$$

Full $\alpha\beta$ Reduction: Continuing on from eager evaluation:

$$\begin{array}{l}\lambda y.(\lambda w.w)y\\ \xrightarrow{\beta}\lambda y.y\end{array}$$

Full $\alpha\beta$ Reduction: Continuing on from lazy evaluation:

$$\begin{array}{l}\lambda y.((\lambda u.u)(\lambda w.w))y\\ \xrightarrow{\beta}\lambda y.(\lambda w.w)y\\ \xrightarrow{\beta}\lambda y.y\end{array}$$