CSE 3302/5307 Programming Language Concepts

Homework3 - Fall 2023

Due Date: Sep.16, 2023, 8:00p.m. Central Time

Problem1 - 40%

Evaluate the following λ expressions using call-by-value and call-by-name. Show the complete steps of evaluation.

(a) $(\lambda x.\ ((\lambda y.\ x + z + 3)\ 3)5)$

(b) $((\lambda v.(\lambda w.w))\ ((\lambda x.x)\ (y\ (\lambda z.z))))$

(c) $((\lambda x.\ x\ x)\ (\lambda y.\ y\ y))$

(d) $((\lambda x.\lambda y.x)\ (\lambda z.z\ \lambda u.u))$

Problem2 - 30%

Prove by induction: If e_1 is closed and $e_1 \to e_2$, then e_2 is closed. Suppose using call by value evaluation.

- Given the following definitions:
  1. Rules of free variables (You can use these rules directly without writing ”By ...”)

$$\frac{}{FV(x) = \{x\}} \qquad \frac{FV(e_1) = S_1 \quad FV(e_2) = S_2}{FV(e_1\ e_2) = S_1 \cup S_2} \qquad \frac{FV(e) = S}{FV(x.e) = S - \{x\}}$$

  2. Judgment form: **define** $e_1 \to e_2$

$$\frac{}{(\lambda x.e)\ v \to e[v/x]} \qquad \frac{e_1 \to e_1^{'}}{e_1\ e_2 \to e_1^{'}\ e_2} \qquad \frac{e_2 \to e_2^{'}}{v\ e_2 \to v\ e_2^{'}}$$

  3. Judgment form: **define** $e_1 {\to}^* e_2$

$$\frac{}{e_1 \to^* e_1} \qquad \frac{e_1 \to e_2 \qquad e_2 \to^* e_3}{e_1 \to^* e_3}$$

- And given this lemma:

  Lemma 1. $FV(e_1[e_2/x]) \subseteq (FV(e_1) - \{x\}) \cup FV(e_2)$

  You can use this lemma directly. (Proof is in the appendix)

Problem3 - 30%

Church encoding is a means of embedding data and operators into the λ calculus, the most familiar form being the Church numerals, a representation of the natural numbers using λ notation. Church numerals $\mathbf{0}, \mathbf{1}, \mathbf{2}, ...$, are defined as follows:

$$
\begin{aligned}
\mathbf{0} &= \lambda f.\lambda x.\ x \\
\mathbf{1} &= \lambda f.\lambda x.\ f\ x \\
\mathbf{2} &= \lambda f.\lambda x.\ f\ (f\ x) \\
\mathbf{3} &= \lambda f.\lambda x.\ f\ (f\ (f\ x)) \\
&\ \ \dots \\
\mathbf{n} &= \lambda f.\lambda x.\ f^n\ x \\
&\ \ \dots
\end{aligned}
$$

Church numerals takes two parameters f and x. Church numerals n means apply f to x n times. (You can refer to Wikipedia or other references about Church encoding to know more about the idea of church encoding)

(a) Define addition in λ calculus, and then show the evaluation of $3 + 2$.

(b) Define multiplication in λ calculus (Hint: use definition of addition), and then show the evaluation of 3×2.

(c) Give a definition of multiplication on Church numerals without using addition.

Remark:
Please email **.pdf** files to TA.
File name format: HW_X_FirstName_10digitID.pdf
Example: HW_3_Sinong_1001001000.pdf

A Proof of Lemma 1

Lemma $FV(e_1[e/x]) \subseteq (FV(e_1) - \{x\}) \cup FV(e)$

Proof. By induction on the derivation of substitution. (Actually you can treat substitution as a special judgement form and the definitions as rules)

1. case $x[e/x] = e$

   Need to prove: $FV(x[e/x]) \subseteq (FV(x) - \{x\}) \cup FV(e)$

   (1) $FV(x[e/x]) = FV(e)$ (by assumption)

   (2) $FV(e) \subseteq (FV(x) - \{x\}) \cup FV(e)$

   (3) $FV(x[e/x]) \subseteq (FV(x) - \{x\}) \cup FV(e)$ (by (1) and (2))

2. case $y[e/x] = y$

   Need to prove: $FV(y[e/x]) \subseteq (FV(y) - \{x\}) \cup FV(e)$

   (1) $FV(y[e/x]) = FV(y)$ (by assumption)

   (2) $FV(y) \subseteq FV(y) - \{x\} \subseteq (FV(y) - \{x\}) \cup FV(e)$

   (3) $FV(y[e/x]) \subseteq (FV(y) - \{x\}) \cup FV(e)$ (by (1) and (2))

3. case $(e_1\ e_2)[e/x] = e_1[e/x]\ e_2[e/x]$ (rule format: $\frac{e_1[e/x]=a \quad e_2[e/x]=b}{(e_1\ e_2)[e/x]=a\ b}$)

   Need to prove: $FV((e_1\ e_2)[e/x]) \subseteq (FV(e_1\ e_2) - \{x\}) \cup FV(e)$

   (1) $FV((e_1\ e_2)[e/x]) = FV(e_1[e/x]) \cup FV(e_2[e/x])$ (by assumption)

   (2) $FV(e_1[e/x]) \subseteq (FV(e_1) - \{x\}) \cup FV(e)$ (by I.H.)

   (3) $FV(e_2[e/x]) \subseteq (FV(e_2) - \{x\}) \cup FV(e)$ (by I.H.)

   (4) $FV((e_1\ e_2)[e/x]) \subseteq (FV(e_1) - \{x\}) \cup (FV(e_2) - \{x\}) \cup FV(e) \subseteq (FV(e_1\ e_2) - \{x\}) \cup FV(e)$ (by (1), (2) and (3))

4. case $(\lambda x.e_1)[e/x] = \lambda x.e_1$

   Need to prove $FV((\lambda x.e_1)[e/x]) \subseteq (FV(\lambda x.e_1) - \{x\}) \cup FV(e)$

   (1) $FV((\lambda x.e_1)[e/x]) = FV(\lambda x.e_1)$ (by assumption)

   (2) $FV((\lambda x.e_1)[e/x]) \subseteq (FV(\lambda x.e_1) - \{x\}) \cup FV(e)$ (by (1))

5. case $(\lambda y.e_1)[e/x] = \lambda y.(e_1[e/x])$ ($y \neq x$ and $y \notin FV(e)$)

   Need to prove $FV((\lambda y.e_1)[e/x]) \subseteq (FV(\lambda y.e_1) - \{x\}) \cup FV(e)$

   (1) $FV((\lambda y.e_1)[e/x]) = FV(\lambda y.(e_1[e/x])) = FV(e_1[e/x]) - \{y\}$ (by assumption)

   (2) $FV(e_1[e/x]) \subseteq (FV(e_1) - \{x\}) \cup FV(e)$ (by I.H.)

   (3) $FV(\lambda y.e_1) = FV(e_1) - \{y\}$

   (4) $FV((\lambda y.e_1)[e/x]) \subseteq (FV(\lambda y.e_1) - \{x\}) \cup FV(e)$ (by (1), (2) and (3))

6. case $(\lambda y.e_1)[e/x] = \lambda z.(e_1[[z/y]][e/x])$ ($y \neq x$, $y \in FV(e)$ and $z \notin FV(e) \cup Vars(e_1)$)
   Need to prove $FV((\lambda y.e_1)[e/x]) \subseteq (FV(\lambda y.e_1) - \{x\}) \cup FV(e)$
   (1) $FV((\lambda y.e_1)[e/x]) = FV(\lambda z.(e_1[[z/y]][e/x])) = FV(e_1[[z/y]][e/x]) - \{z\}$ (by assumption)
   (2) $FV(e_1[[z/y]][e/x]) \subseteq (FV(e_1[[z/y]]) - \{x\}) \cup FV(e)$ (by I.H.)
   (3) $FV(e_1[[z/y]]) \subseteq (FV(e_1) - \{y\}) \cup FV(z)$ (by I.H.)
   (4) $FV(\lambda y.e_1) = FV(e_1) - \{y\}$
   (5) $FV((\lambda y.e_1)[e/x]) \subseteq (FV(\lambda y.e_1) - \{x\}) \cup FV(e)$ (by (1), (2), (3) and (4))

□

About the proof of this Lemma, you just need to know the idea. I found another proof (This is another proof) from the internet. You will find the idea is the same. Both proofs use induction to prove this property.