Midterm 2 (22C:123), Spring 2000

Open Books and Notes

1. (50) **(Fixed Point Semantics)** The function F is defined as follows:

$$F = \lambda g.\lambda n.(\textbf{if } (n = 1) \textbf{ then } 2 \textbf{ else if } even(n) \textbf{ then } n + g(n/2) \textbf{ else } g(2n - 3))$$

   (a) Please list the functions $F\ \bot$, $F^2\ \bot$, $F^3\ \bot$, in their simplest form.
   Answer:

   - $F\ \bot\ n = \textbf{if } n = 1 \textbf{ then } 2 \textbf{ else } \bot$;
   - $F^2\ \bot\ n = \textbf{if } n = 1 \vee n = 2 \textbf{ then } 2n \textbf{ else } \bot$;
   - $F^3\ \bot\ n = \textbf{if } n = 1 \vee n = 2 \vee n = 4 \textbf{ then } 2n \textbf{ else } \bot$.

   (b) Please provide the function $F^i\ \bot$ and prove by induction on i that your $F^i\ \bot$ is correct.
   Answer:
   $F^i\ \bot\ n = \textbf{if } n = 2^j\ (0 \leq j < i) \textbf{ then } 2n \textbf{ else } \bot$.
   $F^0\ \bot\ = \bot$ is the base case when $n = 0$.
   For the inductive case,

$$\begin{aligned} F^{i+1}\ \bot\ n &= \textbf{if } (n = 1) \textbf{ then } 2 \textbf{ else} \\ &\qquad \textbf{if } even(n) \textbf{ then } n + F^i\ \bot\ (n/2) \textbf{ else } F^i\ \bot\ (2n - 3) \\ &= \textbf{if } n = 2^j\ (0 \leq j \leq i) \textbf{ then } 2n \textbf{ else } \bot\ . \end{aligned}$$

   (c) Please find the least fixed point f_1 of F.
   Answer: $f_1\ n = \textbf{if } n = 2^j\ (0 \leq j) \textbf{ then } 2n \textbf{ else } \bot$.

   (d) Please find two other different fixed points f_2 and f_3 of F such that $f_1 \subseteq f_2 \subseteq f_3$, where $\subseteq$ is the partial order on functions as given in the textbook.
   Answer:

   - $f_2\ n = \textbf{if } n = 0 \vee n = 2^j\ (0 \leq j) \textbf{ then } 2n \textbf{ else } \bot$.
   - $f_3\ n = \textbf{if } n = 0 \vee n = 2^j \vee n = 3 * 2^j (0 \leq j) \textbf{ then } 2n \textbf{ else } \bot$.

   We can verify that $F\ f_i = f_i$ for $i = 1, 2, 3$ and $f_1 \subseteq f_2 \subseteq f_3$.

2. (50) **(Denotational Semantics)** We like to add into Wren two features of C language.

   (a) One is the C-style conditional expression:

   Expression ::= ... | Expression? Expression : Expression

   Please provide denotational semantics for the conditional expression in Wren and use your definition to prove the semantic equivalence of the following two commands:

   $m := e_1?\ e_2 : e_3;$

   and

   if e_1 **then** $m := e_2$ **else** $m := e_3;$

   Answer: Please see the answer to Problem 9.3.1 b).

   (b) The other is the operators ++ and --: $++i$ in an expression will return one plus the value of i; $i++$ will return the value of i. In both cases, the value of i will be increased by one. The meaning for -- is similar.

   Please modify Wren's denotational sementics to handle ++ and -- (both before and after an integer identifier).

   Answer: The store needs to be updated and the updated store should be returned by `evaluate`. Please see the answer to Problem 9.3.1 c).