This lack of λ confirms that the semantics is first-order. Second, the predomain equation for V can be solved in the category of sets, i.e., without using domain theory. That is, we can define a set V s.t.

$$V = \mathbb{Z} + \widehat{V}_{fun} + \widehat{V}_{cont}$$

equality.

where $\widehat{V}_{fun}$ and $\widehat{V}_{cont}$ are defined as before.

(5) I tried to derive the program transformation from this first-order semantics. But I couldn't find a simple way to do so. Sorry guys.

Let me instead show you how one can derive a small-step evaluation relation (or more commonly called small-step operational semantics) from the first-order denotational semantics. The idea is to replace $=$ by a single evaluation step $\longrightarrow$.

$$\langle n, \eta, \kappa \rangle \longrightarrow \langle \text{cont}, \kappa, n \rangle$$

$$\langle -e, \eta, \kappa \rangle \longrightarrow \langle e, \eta, \langle \text{negate}, \kappa \rangle \rangle$$

$$\langle e_0 \overset{\div}{\underset{*}{+}} e_1, \eta, \kappa \rangle \longrightarrow \langle e_0, \eta, \langle \text{add}_1 / \text{div}_1 / \text{mul}_1, e_1, \eta, \kappa \rangle \rangle$$

$$\langle v, \eta, \kappa \rangle \longrightarrow \langle \text{cont}, \kappa, (\text{get } \eta\ v) \rangle$$

$$\langle e_0\, e_1, \eta, \kappa \rangle \longrightarrow \langle e_0, \eta, \langle \text{app}_1, e_1, \eta, \kappa \rangle \rangle$$

$$\langle \lambda v. e, \eta, \kappa \rangle \longrightarrow \langle \text{cont}, \kappa, \langle \text{abstract}, v, e, \eta \rangle \rangle$$

$$\langle \text{callcc } e, \eta, \kappa \rangle \longrightarrow \langle e, \eta, \langle \text{ccc}, \kappa \rangle \rangle$$

$$\langle \text{throw } e\ e', \eta, \kappa \rangle \longrightarrow \langle e, \eta, \langle \text{thw}, e', \eta \rangle \rangle$$

$$\langle \text{letrec } v_0 \equiv \lambda u_0. e_0 \text{ in } e, \eta, \kappa \rangle \longrightarrow \langle e, \langle \text{recenv}, \eta, v_0, u_0, e_0 \rangle, \kappa \rangle$$

$$\langle \text{cont}, \langle \text{negate}, \kappa \rangle, a \rangle \longrightarrow \langle \text{cont}, \kappa, -a \rangle \quad (\text{if } a \in \mathbb{Z})$$

$$\langle \text{cont}, \langle \text{add}_1, e, \eta, \kappa \rangle, a \rangle \longrightarrow \langle e, \eta, \langle \text{add}_2, a, \kappa \rangle \rangle$$

$$\langle \text{cont}, \langle \text{add}_2, a, \kappa \rangle, b \rangle \longrightarrow \langle \text{cont}, \kappa, a+b \rangle \quad (\text{if } a, b \in \mathbb{Z})$$

$\text{mul}_1, \text{mul}_2, \text{div}_1$ are similar

$$\langle \text{cont}, \langle \text{div}_2, a, \kappa \rangle, b \rangle \longrightarrow \langle \text{cont}, \kappa, a \div b \rangle \quad (\text{if } a, b \in \mathbb{Z} \text{ and } b \neq 0)$$

$$\langle \text{cont}, \langle \text{app}_1, e, \eta, \kappa \rangle, a \rangle \longrightarrow \langle e, \eta, \langle \text{app}_2, a, \kappa \rangle \rangle$$

$$\langle \text{cont}, \langle \text{app}_2, a, \kappa \rangle, b \rangle \longrightarrow \langle \text{apply}, a, b, \kappa \rangle \quad (\text{if } a \in \widehat{V}_{fun})$$

$$\langle \text{cont}, \langle \text{ccc}, \kappa \rangle, a \rangle \longrightarrow \langle \text{apply}, a, \kappa, \kappa \rangle \quad (\text{if } a \in \widehat{V}_{fun})$$