Harvard School of Engineering and Applied Sciences — CS 152: Programming Languages

Abstract register machines, delimited continuations Section and Practice Problems

Monday April 20, 2015

---

1 Abstract register machines

(a) Consider the following program in the language from the lecture notes (Lecture 20):

$$((((\lambda f.\, \lambda g.\, \lambda x.\, \lambda y.\, f\ (x) + g\ (y))\ \lambda x.\, 0)\ \lambda x.\, 42)\ 42)\ 42$$

Show its evaluation trace under CC, SCC, CK and CEK semantics.

(b) Design a SCC machine for the language from the notes of lecture 19 with errors, conditionals and primitives to check the type of values. This language consists of the first language of the lecture notes and its first extension — ignore error handling for this question.

(c) Show that the CK machine from the lecture notes is equivalent to the CK machine from the lecture notes. Put differently, show that for any e in our language, $\langle e, [\cdot] \rangle \xrightarrow{*}_{SCC} \langle e, [\cdot] \rangle$ iff $\langle e, \mathbf{mt} \rangle \xrightarrow{*}_{CK} \langle e, \mathbf{mt} \rangle$. To do that, define a bijection that maps evaluation contexts to continuations. Here is one case of the bijection's definition: $\mathcal{T}[\![E[[\cdot]\ e]]\!] = \langle \mathbf{arg}, e, \mathcal{T}[\![E]\!] \rangle$. Use the bijection to show that $\langle e, E \rangle \longrightarrow_{SCC} \langle e', E' \rangle$ iff $\langle e, \mathcal{T}[\![E]\!] \rangle \longrightarrow_{CK} \langle e', \mathcal{T}[\![E']\!] \rangle$ and derive the desired conclusion with an induction on the length of the traces (they have the same length).

2 Delimited continuations

(a) Show the evaluation trace of the following program under the CC machine semantics of the first section of the lecture notes (Lecture 21):

$$\mathsf{try}\ ((\lambda f.\, \lambda x.\, f\ x\ )\ \lambda x.\, x + x)\ \lambda x.\, 42\ \mathsf{resume}\ f,x.(\mathsf{try}\ f\ \lambda x.\, x + x\ \mathsf{resume}\ f,x.f\ x)$$

(b) Show the evaluation trace of the following program under the CC machine semantics of the second section of the lecture notes:

$$\%\ (\%\ (((\lambda f.\, \lambda x.\, f\ x\ )\ (\lambda x.\, x + \mathsf{abort}\ \lambda x.\, x))\ 42)\ \mathsf{call/c}\ \lambda f.\, f\ 42)$$

(c) Translate the following program using the second version of the translation of the last section of the lecture notes (ignore the final translation for this question):

$$\mathsf{try}\ ((\lambda f.\, \lambda x.\, f\ x\ )\ (\lambda x.\, x + \mathsf{raise}\ 42))\ 42\ \mathsf{catch}\ x.x$$

Then show the evaluation trace of the translated program under the semantics of the second section of the lecture notes. Compare the result with the result of evaluating the same program under the CC machine semantics of the first section of the lecture notes.