Models of Computation

Written Exam on September 2, 2011

(MOD students: Exercises 1-5, 180 minutes Previous TSD students: Exercises 1-3, 120 minutes)

Exercise 1 (7)

Assume that $<\textbf{while } b \textbf{ do } c, \sigma> \to \sigma' \Rightarrow <b, \sigma'> \to \mathit{false}$ and the corresponding denotational property, namely $\mathcal{C}[\![\textbf{while } b \textbf{ do } c]\!]\sigma = \sigma' \Rightarrow \mathcal{B}[\![b]\!]\sigma' = \mathit{false}$.

Prove, employing both the operational and the denotational semantics (but not their equivalence!) that the commands

$$c_1 = (\textbf{while } b \textbf{ do } c); \textbf{while } b \textbf{ do } c \qquad c_2 = \textbf{while } b \textbf{ do } (c; \textbf{while } b \textbf{ do } c)$$

are equivalent.

Exercise 2 (6)

Let $(\mathcal{P}(\omega), \subseteq)$ be the cpo with bottom consisting of the subsets of natural numbers ordered by inclusion. Every term F of the abstract syntax

$$F \ ::= \ X \mid F \cup F \mid F \cap F \mid C \quad \text{con} \quad C \in \mathcal{P}(\omega)$$

represents, according to the following definition by structural induction,

$$[\![X]\!](S) = S \quad [\![F_1 \cup F_2]\!](S) = [\![F_1]\!](S) \cup [\![F_2]\!](S) \quad [\![F_1 \cap F_2]\!](S) = [\![F_1]\!](S) \cap [\![F_2]\!](S) \quad [\![C]\!](S) = C$$

a function $[\![F]\!] : \mathcal{P}(\omega) \to \mathcal{P}(\omega)$.

Prove by structural induction that all the functions $[\![F]\!]$ are (i) monotone (ii) continuous.

Exercise 3 (7)

Modify the semantics of HOFL assuming the following operational semantics for the conditional command:

$$\frac{t_0 \to 0 \ \ t_1 \to c_1 \ \ t_2 \to c_2}{\textbf{if } t_0 \textbf{ then } t_1 \textbf{ else } t_2 \to \ c_1} \qquad \frac{t_0 \to n \ \ n \neq 0 \ \ t_1 \to c_1 \ \ t_2 \to c_2}{\textbf{if } t_0 \textbf{ then } t_1 \textbf{ else } t_2 \to \ c_2}.$$

Exibit the corresponding denotational semantics and prove that also for the modified semantics it holds that $t \to c$ implies $[\![t]\!] = [\![c]\!]$. Finally, compute the operational and the denotational semantics of $(\mathit{fact}\ 0)$, with $\mathit{fact} = \mathit{rec}\ f.\lambda x.\ \textbf{if } x \textbf{ then } 1 \textbf{ else } x \times (f\ (x-1))$, and check if they coincide..

Exercise 4 (5)

Let us consider the π-calculus with the additional operations $P ::= (\nu_y)P$, $P ::= (o_y)P$ and $P ::= \overline{x}(y).P$. The new operations are characterized by the following axioms:
(i) $(y)P = (\nu_y)P + (o_y)P$; (ii) $(\nu_y)\alpha.P = \textbf{if } y \in n(\alpha) \textbf{ then } \mathit{nil} \textbf{ else } \alpha.(y)P$; (iii) $(\nu_y)(P+Q) = (\nu_y)P + (\nu_y)Q$; (iv) $(o_y)\overline{x}y.P = \textbf{if } x = y \textbf{ then } \mathit{nil} \textbf{ else } \overline{x}(y).P$; (v) $(o_y)(P+Q) = (o_y)P + (o_y)Q$; and (vi)$(\nu_y)\mathit{nil} = (o_y)\mathit{nil} = \mathit{nil}$.
Define inference rules also for the new operators $P ::= (\nu_y)P$, $P ::= (o_y)P$ and $P ::= \overline{x}(y).P$. Furthermore, prove using the rules just defined that the axioms (i), (ii) and (iv) do *bisimulate*. Namely, given a generic transition of P, for (i); the transition $\alpha.P \xrightarrow{\alpha} P$, for (ii); and $\overline{x}y.P \xrightarrow{\overline{x}y} P$ for (iv), every transition of the left hand side of the axiom should be matched by a transition of the right hand side (and viceversa) with the same action and with continuations which can be proved equivalent using the axioms.

Exercise 5 (5)

In a multiprocessor system with shared memory, processes must compete to use the memory bus. Consider the case of two identical processes. Each process has cyclic behaviour: it performs some local activity (local action *think*), accesses the bus (synchronization action *get*), operates on the memory (local action *use*) and then releases the bus (synchronization action *rel*). The bus has cyclic behavior with actions *get* and *rel*. Define a PEPA program representing the system and derive the corresponding CTMC (with actions). Find the bisimilar states and draw the minimal CTMC.