Principles for software composition 2019/20

Self assessment - April 2020

[Ex. 1] Suppose we extend IMP with the arithmetic expression for integer division a_0/a_1, whose operational semantics is defined by the inference rule

$$\frac{\langle a_0, \sigma\rangle \to n_0 \quad \langle a_1, \sigma\rangle \to n_1 \quad n_1 \neq 0}{\langle a_0/a_1, \sigma\rangle \to n_0 \ \underline{\mathbf{div}}\ n_1}$$

1. Prove that termination does not hold anymore for arithmetic and boolean expressions.
2. Prove that determinacy continues to hold for arithmetic and boolean expressions.
3. Redefine the denotational semantics of *Aexp*, *Bexp* and *Com* to take into account non termination of arithmetic expressions.

[Ex. 2] Let $c \stackrel{\text{def}}{=} \mathbf{while}\ x \neq y\ \mathbf{do}\ (x := x+1; y := y-1)$. Find the largest set of memories S such that $\forall \sigma \in S.\ \langle c, \sigma\rangle \not\to$.

[Ex. 3] Let $(D, \sqsubseteq_D)$ be a CPO. Given any chain $\{d_n\}_{n\in\mathbb{N}}$ in D prove that

$$\bigsqcup_{n\in\mathbb{N}} d_n = \bigsqcup_{n\in\mathbb{N}} d_{2n}$$

Hint: Prove that the chains $\{d_n\}_{n\in\mathbb{N}}$ and $\{d_{2n}\}_{n\in\mathbb{N}}$ have the same set of upper bounds and therefore the same lub.

[Ex. 4] Write a Haskell function `nodup` that takes a list and checks that no element occurs more than once in the list. For example, `nodup [1,5,3]` must return to `true`, while `nodup [1,5,3,5]` must return to `false`. Write down also the type of `nodup`.

[Ex. 5] Which of the following HOFL pre-terms are well-formed?
If they are, write their principal type. If they are not, explain why.

1. $\lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ \mathbf{fst}(x)\ \mathbf{else}\ \mathbf{snd}(x)$
2. $\mathbf{rec}\ f.\ \lambda x.\ f(\mathbf{snd}(x), \mathbf{fst}(x))$
3. $\mathbf{rec}\ f.\ \lambda x.\ \mathbf{if}\ \mathbf{fst}(x)\ \mathbf{then}\ \mathbf{fst}(x)\ \mathbf{else}\ f\ (\mathbf{snd}(x), 0)$
4. $\mathbf{rec}\ f.\ \lambda g.\ \lambda x.\ g\ (f\ x))$
5. $\lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x))$
6. $\lambda n.\ \lambda f.\ \lambda x.\ f\ ((n\ f)\ x)$

[Ex. 6] (substitution preserves types) Let $\widetilde{e} : \widetilde{\tau}$ stand for $e_1 : \tau_1, ..., e_n : \tau_n$, and $t[^{\widetilde{e}}/_{\widetilde{x}}]$ for $t[^{e_1}/_{x_1}, ..., ^{e_n}/_{x_n}]$. Prove that $\forall t.\ P(t)$ where

$$P(t) \stackrel{\text{def}}{=} \forall \tau, \widetilde{x}, \widetilde{e}, \widetilde{\tau}.\ (t : \tau \wedge \widetilde{x} : \widetilde{\tau} \wedge \widetilde{e} : \widetilde{\tau}) \ \Rightarrow\ t[^{\widetilde{e}}/_{\widetilde{x}}] : \tau$$