Home Assignment 2
for Programming Theory (TDDA 43)

Deadline: Wednesday week 20 (13 May 2009) at noon
Homework reporting session: 14 May 8.15

Provide explanations to justify your solutions. Put your solutions, addressed to Jonas Wallgren, in the "Post till IDA" slot at Café Java. Keep a copy of your solutions for the homework reporting session.

The numbers in parentheses refer to the old version of our textbook.

A. Exercise 5.7 (4.7) from the book. (Comparing functions from **State** $\hookrightarrow$ **State**).

B. 1. Complete (in two ways) the following definition of functions g_n: **State** $\hookrightarrow$ **State**

$$g_n s = \begin{cases} s[\mathtt{x} \mapsto \ldots] & \text{if } s\,\mathtt{y} < n \\ \underline{\text{undef}} & \text{if } s\,\mathtt{y} \geq n \end{cases}$$

(where $n \in \mathbb{Z}$, i.e. n may be any integer) so that $Y = \{\, g_n \mid n \in \mathbb{Z} \,\}$ is:

(a) a chain,

(b) not a chain

in the ccpo (**State** $\hookrightarrow$ **State**, $\sqsubseteq$). Explain that the set is indeed a chain, respectively not a chain.

2. Consider the set Y above that is a chain. What is the least upper bound $\bigsqcup Y$ of Y? Add one element g to Y so that $Y \cup \{g\}$ does not have an upper bound. Add one element h to Y so that $Y \cup \{h\}$ is not a chain, but has a least upper bound. Can you find an h such that the least upper bounds of Y and of $Y \cup \{h\}$ are (i) the same, (ii) not the same?

C. Let $(D, \sqsubseteq)$, $(D', \sqsubseteq)$ be partially ordered sets and $f\colon D \to D'$ be a function such that for each non-empty chain $Y \subseteq D$ the least upper bound $\bigsqcup'\{\, f d \mid d \in Y \}$ exists and

$$\bigsqcup{}'\{\, f d \mid d \in Y \} = f\left(\bigsqcup Y\right).$$

Show that f is monotone.

Hint. You have to show that $f d_1 \sqsubseteq f d_2$ for any $d_1 \sqsubseteq d_2$. Consider $Y = \{d_1, d_2\}$.

D. Which of the functions given in Exercise 5.28 (4.28) are monotone?

E. Exercise 6.9 (4.70) from the textbook. (Add call by value arguments to the language **Proc**; modify its denotational semantics to describe this extension.) Part of the task is already done, by defining the semantics of the procedure call by:

$$\mathcal{S}_{\mathrm{ds}}[\![\texttt{call}\ p(a_1, a_2)]\!]\ env_V\ env_P\ sto \;=\; env_P\ p\ (\mathcal{A}[\![a_1]\!]s,\ \mathcal{A}[\![a_2]\!]s)\ sto$$
$$\text{where}\quad s = \text{lookup}\ env_V\ sto$$

It remains to describe the semantics of procedure declarations.

F. 1. Check whether each of the following lambda terms has a normal form or not. If the normal form exists show a reduction deriving it, otherwise explain why the normal form does not exist.

(a) $(\lambda x.xy)((\lambda y.yz)(\lambda z.zu))$

(b) $(\lambda x.xxx)(\lambda x.xxx)$

(c) $(\lambda y.x)(\lambda x.xxx)(\lambda x.xxx)$

Explain in your own words the Church-Rosser theorem and illustrate it on one of the terms above.

2. Find a lambda term T such that for any lambda terms X, Y the term TXY reduces in lambda calculus to YX.

Repeat the above for combinatory terms, i.e. find a combinatory term T, using only combinators S and K, such that for any combinatory terms X, Y the term TXY reduces in combinatory logic to YX.

Check the second construction on an example.

Could the combinatory term be simplified if the I combinator were allowed?

The maximal marks for problems A, B, C, D, E and F are, respectively, 2, 4+4, 3, 2, 7, 2+2. To pass one needs 13 points including at least 3 for problem E, and 2 for problem F. Your answers may be in English or Swedish.

It is allowed to discuss the exercises with others, but you are supposed to solve each exercise individually. It is absolutely not allowed to copy solutions from others.