Semantik och Principer för
Programmeringsspråk

Examination
22 August 2002, 3pm–8pm

Faron Moller
UU/CSD

- Please give solutions in English if possible. If you are particularly uncomfortable with English, you may of course use Swedish.

- Write your name on every page!

- The maximum number of points is given with each problem. There are 60 points possible in total; these are added to the points earned on the exercises, to give a total score out of 80. To pass, you must score 36 out of 80. To get VG, you must score 56 out of 80. (Unless you have prior credit for the exercises, in which case you must score 27 out of 60 on the exam to pass, and 42 out of 60 to get a VG.)

- Good Luck!

1. (a) Build derivation trees for the two inferences

\[ \langle x := y, \sigma \rangle \rightarrow \sigma_1 \quad \text{and} \quad \langle x := y; y := x, \sigma \rangle \rightarrow \sigma_2. \]

(b) Use these derivation trees to demonstrate that \( \sigma_1 = \sigma_2 \), that is, that the two commands are operationally equivalent. [3 marks]

2. (a) What does it mean for a semantic definition to be \textit{compositional}, and what bearing does this have on the types of induction proofs which can be used to prove theorems about the semantic definitions? [5 marks]

(b) Prove the following:

\textbf{Theorem}: For all commands \( c \in \text{Com} \) which do not contain any assignment statements, if \( \langle c, \sigma \rangle \rightarrow \sigma' \) then \( \sigma' = \sigma \). [7 marks]
3. Consider the imperative language \textbf{Imp2} which extends \textbf{Imp} by allowing both integers and booleans to be stored by variables. Specifically, the domain of storable values is to be $\mathbb{V} = \mathbb{Z} + \mathbb{T}$, where $\mathbb{Z}$ is the set of integers and $\mathbb{T} = \{\text{true}, \text{false}\}$. A state $\sigma \in \Sigma$ is now a mapping from $\textbf{Var}$ to $\mathbb{V}$.

We also introduce the element $\text{err}$ to denote the outcome of mismatched types, and define $\mathbb{V}_{\text{err}} = \mathbb{V} \cup \{\text{err}\}$. For example, the result of “3 or true” is $\text{err}$.

Finally, we also have $\Sigma_{\text{err}} = \Sigma \cup \{\text{err}\}$.

\textbf{Imp2} does not distinguish between $\textbf{Aexp}$ and $\textbf{Bexp}$, but rather it uses a single set $\textbf{Exp}$ of expressions defined by the following BNF expression:

$$e ::= n \mid t \mid x \mid e_0 + e_1 \mid e_0 = e_1 \mid \text{not } e \mid e_0 \text{ or } e_1$$

where $n$ ranges over $\mathbb{N}$, $t$ over $\mathbb{B}$ and $x$ over $\textbf{Var}$. (For simplicity we omit the other expressions from \textbf{Imp} such as $e_0 - e_1$ and $e_0 \times e_1$.) The syntax for commands $\textbf{Com}$ is given as

$$c ::= \text{skip} \mid x := e \mid e_0 ; e_1 \mid \text{if } e \text{ then } e_0 \text{ else } e_1 \mid \text{while } e \text{ do } c$$

where $e$ ranges over $\textbf{Exp}$.

(a) Give operational semantic rules for evaluating expressions. They should derive expressions of the form $(e, \sigma) \rightarrow p$ where $e \in \textbf{Exp}$, $\sigma \in \Sigma$ and $p \in \mathbb{V}_{\text{err}}$. [3 marks]

(b) Give operational semantic rules for executing commands. They should derive expressions of the form $(c, \sigma) \rightarrow \rho$ where $c \in \textbf{Com}$, $\sigma \in \Sigma$ and $\rho \in \Sigma_{\text{err}}$. [3 marks]

(c) Define the denotational semantic function for expressions

$$\mathcal{E} : \textbf{Exp} \rightarrow (\Sigma \rightarrow \mathbb{V}_{\text{err}}).$$

(d) Define the denotational semantic function for commands

$$\mathcal{C} : \textbf{Com} \rightarrow (\Sigma \rightarrow \Sigma_{\text{err}}).$$

(e) Consider the domain $(\Sigma_{\text{err}})_{\bot} = \Sigma \cup \{\text{err}, \bot\}$. What relationship (in terms of $\sqsubseteq$) do you propose between $\text{err}$ and $\bot$ in $(\Sigma_{\text{err}})_{\bot}$? Justify your answer. [3 marks]
4. (a) Define the following terms: [4 marks]
(ii) least upper bound.
(iii) complete partial order (domain).
(iv) monotonic.
(v) continuous.

(b) Let $T_\bot = \{\bot_T, \text{false, true}\}$ and $O = \{\bot, \top\}$, with the ordering $\sqsubseteq$ defined by $\bot_T \sqsubseteq \text{false}, \bot_T \sqsubseteq \text{true}, \bot \sqsubseteq \top$, and $x \sqsubseteq x$ for $x \in T_\bot \cup O$.

(i) How many functions $f : T_\bot \to O$ are there? [2 marks]
How many functions $f : O \to T_\bot$ are there?

(ii) How many continuous functions $f : T_\bot \to O$ are there? [2 marks]
How many continuous functions $f : O \to T_\bot$ are there?

(iii) Which functions $f : T_\bot \to O$ are not continuous? [2 marks]
Which functions $f : O \to T_\bot$ are not continuous?

5. Give appropriate semantic domains for the following SML data types representing a student’s name, score, and passing status: [4 marks]

\[
\begin{align*}
\text{datatype NAME} & = \text{Name of string;} \\
\text{datatype SCORE} & = \text{Score of int;} \\
\text{datatype GRADE} & = \text{U | G | VG;} \\
\text{datatype RESULT} & = \text{Results of NAME \* SCORE \* GRADE;}
\end{align*}
\]
6. Consider the following process definition.

\[
\begin{align*}
X_1 & \overset{\text{def}}{=} a.X_1 + b.X_3 \\
X_2 & \overset{\text{def}}{=} a.X_3 + a.X_6 + b.X_1 \\
X_3 & \overset{\text{def}}{=} a.X_5 \\
X_4 & \overset{\text{def}}{=} a.X_1 + b.X_3 \\
X_5 & \overset{\text{def}}{=} a.X_3 + a.X_6 + b.X_1 \\
X_6 & \overset{\text{def}}{=} a.X_3 + a.X_5 + b.X_4
\end{align*}
\]

(a) Draw the labelled transition system for the above process.
How many states, actions, and transitions does it have? \[4 \text{ marks}\]

(b) The following labelled transition system represents the minimal process equivalent to the above process.

![Diagram]

Explain precisely in what sense this is true. \[3 \text{ marks}\]

(c) Give a process definition for this minimized transition system. \[3 \text{ marks}\]

(d) For each state of the minimized transition system, give a modal formula which is true of it but of no other state in the automaton. \[4 \text{ marks}\]