Semantik och Principer för
Programmeringsspråk

Examination
5 June 2002, 3pm–8pm

Faron Møller
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. Prove the following:

   If $\langle \text{while } b \text{ do } c, \sigma \rangle \rightarrow \sigma'$ then $\langle b, \sigma' \rangle \rightarrow \text{false.}$

   [5 marks]

---

2. Consider adding **double assignment** as a command in Imp:

   **Syntax:** $(x, y) := (a_0, a_1)$

   **Informal Semantics:** evaluate both expressions $a_0$ and $a_1$ **before** assigning their values to $x$ and $y$, respectively.

   (a) Give operational and denotational definitions for this new command. [5 marks]

   (b) Give an example demonstrating that, in general, $(x, y) := (a_0, a_1)$ is different from $x := a_0 ; y := a_1$. [3 marks]
3. Imagine introducing the \texttt{break} command into \texttt{Imp}, whose intended behaviour is to immediately abort execution of the smallest enclosing \texttt{while}-loop (if any) and to return control to the following commands (if any).

Thus for example, using this new command, the command \texttt{while }\texttt{b }\texttt{do }\texttt{c} is equivalent to

\texttt{while true do if b then c else break}

The syntax of the extended language is

\[ c ::= \text{skip} \mid \text{break} \mid x := a \mid c_0; c_1 \mid \text{if } b \text{ then } c_0 \text{ else } c_1 \mid \text{while } b \text{ do } c. \]

In order to give operational semantic rules, it no longer suffices to have a transition relation of the form \( \langle c, \sigma \rangle \rightarrow \sigma' \). It is necessary to record whether or not a \texttt{break} has been encountered. To do this we use a transition relation of the form \( \langle c, \sigma \rangle \rightarrow \langle \sigma', t \rangle \), with the following interpretation:

- \( \langle c, \sigma \rangle \rightarrow \langle \sigma', \text{true} \rangle \) means that the execution of \( c \) in state \( \sigma \) terminates in state \( \sigma' \) by encountering a \texttt{break} command at the topmost level (not inside the body of a while-loop).

- \( \langle c, \sigma \rangle \rightarrow \langle \sigma', \text{false} \rangle \) means that the execution of \( c \) in state \( \sigma \) terminates in state \( \sigma' \) without encountering a \texttt{break} command at the topmost level.

Thus for example, the rules for \texttt{skip} and \texttt{break} would be as follows.

\[ \langle \text{skip}, \sigma \rangle \rightarrow \langle \sigma, \text{false} \rangle \quad \text{and} \quad \langle \text{break}, \sigma \rangle \rightarrow \langle \sigma, \text{true} \rangle. \]

(a) Give an operational semantic definition for commands in this language. \hspace{1cm} [8 marks]

(b) Give a denotational semantic definition for commands in this language. \hspace{1cm} [8 marks]
4. A function $f : D \to E$ between $cpos (D, \sqsubseteq_D)$ and $(E, \sqsubseteq_E)$ is continuous iff
   
   (i) $f$ is monotonic; \textit{and}
   
   (ii) $f$ “preserves lubs of chains”: for all chains $d_0 \sqsubseteq_D d_1 \sqsubseteq_D d_2 \sqsubseteq_D \cdots$ in $D$,

   $$f\left( \bigcup_{n \in \omega} d_n \right) = \bigcup_{n \in \omega} f(d_n).$$

   Prove that the first clause is redundant, that is, prove the following: \hspace{1cm} [5 marks]

   If $f$ preserves lubs of chains then $f$ is monotonic.

5. The following is a Pascal data type which represents a person’s name (as a string of $k$ characters), as well as \textit{either} a salary if she/he is employed \textit{or} an indication as to whether that person has a disability:

   ```pascal
   record
       name : array [1..k] of char;
     case employed : boolean of
         true : ( salary : integer );
         false : ( disabled : boolean )
   end
   ```

   Give an appropriate semantic domain for representing objects of this type.
6. Consider the following function definition contained in a Rec declaration $d$:

$$\text{test}(x,y,z) = \text{if } x \text{ then } y \text{ else } z$$

Using the Call-by-Value Semantics, the Rec term $\text{test}(t_0, t_1, t_2)$ behaves differently in general from the term $\text{if } t_0 \text{ then } t_1 \text{ else } t_2$, but these two terms are indistinguishable using the Call-by-Name Semantics. Prove these facts as follows.

(a) Prove that $\text{if } \left[ \text{test}(t_0, t_1, t_2) \rightarrow^{d_{\text{va}}} n \right] \text{ then } \left[ \text{if } t_0 \text{ then } t_1 \text{ else } t_2 \rightarrow^{d_{\text{na}}} n \right]$. [4 marks]

(b) Give an example showing that we can have $\left[ \text{if } t_0 \text{ then } t_1 \text{ else } t_2 \rightarrow^{d_{\text{na}}} n \right]$ without having $\left[ \text{test}(t_0, t_1, t_2) \rightarrow^{d_{\text{va}}} n \right]$. [4 marks]

(c) Prove that $\text{test}(t_0, t_1, t_2) \rightarrow^{d_{\text{na}}} n \iff \left[ \text{if } t_0 \text{ then } t_1 \text{ else } t_2 \rightarrow^{d_{\text{na}}} n \right]$. [4 marks]

7. Consider the following two statements about a computer: [7 marks]

(a) “The computer consists of three parts: a CPU, a memory unit, and a bus for communication with the environment.”

(b) “The emergency button can be pushed; this will halt the computer, which will then not do anything further.”

One of these statements can be expressed in the modal logic $M$; express it.

The other statement cannot be formalised in $M$; explain why not.

8. Will Sweden make it past the first round of the World Cup? [1 mark]