Semantics

The language IMP’ comprises integer and boolean expressions and commands, defined by

\[
\begin{align*}
\text{ie} \in Iexp &::= n \mid x \mid \text{ie}_1 \text{iop} \text{ie}_2 \\
\text{be} \in Bexp &::= b \mid \text{ie}_1 \text{bop} \text{ie}_2 \\
C \in \text{Com} &::= \text{skip} \mid x := \text{ie} \mid (C_1; C_2) \mid \\
& \quad \text{if } \text{be} \text{ then } C_1 \text{ else } C_2 \mid \text{repeat } C \text{ until } \text{be}
\end{align*}
\]

where \(n \in \mathbb{Z}, b \in \{\text{true, false}\}, \text{iop} \in \{+, \times, -\}, \text{bop} \in \{<, =\} \) and \(x \in P\text{var}\), a set of program variables.

(a) Give an annotated evaluation semantics for IMP’, expressing the usual behaviour of each command and expression form, which derives statements of the forms

\[
\text{ie}, S \Rightarrow n; R \quad \text{be}, S \Rightarrow b; R \quad C, S \Rightarrow C' \quad R, W
\]

for \(S, S' \in \text{States} = (P\text{var} \rightarrow \mathbb{Z}) \) and \(R, W \subseteq P\text{var}\). \(C, S \Rightarrow C' \quad R, W\) means ‘in state \(S\), command \(C\) executes to state \(S'\) whilst reading the set of variables \(R\) and writing the set of variables \(W\)’. Similarly, if \(e \in Iexp \cup Bexp\) then \(e, S \Rightarrow v; R\) means ‘in state \(S\), \(e\) reads variables \(R\) in evaluating to \(v\)’.

[5 marks]

(b) For \(be \in Bexp, ie \in Iexp, C \in \text{Com} \) use induction on the structure of phrases to give simple definitions of sets

\[
\mathcal{R}(ie), \mathcal{R}(be), \mathcal{PR}(C), \mathcal{PW}(C) \subseteq P\text{var}
\]

where \(\mathcal{R}(e)\) is the set of variables accessed by \(e\), \(\mathcal{PR}(C)\) is a set of variables possibly read during the execution of \(C\) and \(\mathcal{PW}(C)\) is a set of variables possibly written to during the execution of \(C\). Give an example to show that it is not in general true that

\[
C, S \Rightarrow C \quad S'; R, W \quad \text{implies} \quad W = \mathcal{PW}(C).
\]

[5 marks]

(c) Prove that for any \(C, S, S', R, W\)

\[
C, S \Rightarrow C \quad S'; R, W \quad \text{implies} \quad (\forall x \in P\text{var}. x \notin W \Rightarrow S(x) = S'(x))
\]

and that for any \(be, S, S', b, R\)

\[
(\forall x \in \mathcal{R}(be).S(x) = S'(x)) \quad \text{implies} \quad (be, S \Rightarrow b; R \iff be, S' \Rightarrow b; R)
\]

[5 marks]

(d) Prove that for any \(C_1, C_2, C_3\) and \(be\) that if \(\mathcal{R}(be) \cap \mathcal{PW}(C_1) = \emptyset\) then

\[
(C_1; \text{if } be \text{ then } C_2 \text{ else } C_3) \approx \text{if } be \text{ then } (C_1; C_2) \text{ else } (C_1; C_3)
\]

You may assume without proof that if \(C, S \Rightarrow C \quad S'; R, W\) then \(W \subseteq \mathcal{PW}(C)\).

[5 marks]