8 Semantics of Programming Languages (pes20)

Consider the following C-like language, tinyC. It has locally-scoped mutable variables, and functions that take a single argument. Its operational semantics is defined as a transition system over configurations $\langle e, E, s\rangle$ where E is an environment $\{x_1 \mapsto n_1, .., x_j \mapsto n_j\}$, mapping the variable names currently in scope to their addresses, and s is a store $\{n_1 \mapsto v_1, .., n_k \mapsto v_k\}$, mapping each currently allocated address to either an integer n or **undef**. In this question n ranges over $0 \ldots 2^{63}-1$. Programs p consist of finite sets of definitions with distinct names.

$$\begin{array}{lll} \textit{expression}, & e ::= & n \mid x \mid x{=}e' \mid \{\,\mathbf{int}\,x;\, e\} \mid e_1; e_2 \mid f(e) \mid \mathbf{undef} \mid \mathbf{kill}\,x \\ \textit{definition}, & d ::= & \mathbf{int}\,f(\,\mathbf{int}\,x)\{e\} \end{array}$$

$$\frac{E(x){=}n \quad s(n){=}n'}{\langle x, E, s\rangle \longrightarrow \langle n', E, s\rangle}\ \textsc{deref} \qquad \frac{E(x){=}n \quad n \in \mathbf{dom}\,(s)}{\langle \mathbf{kill}\,x, E, s\rangle \longrightarrow \langle 0, E\backslash x, s\backslash n\rangle}\ \textsc{kill}$$

$$\frac{\langle e, E, s\rangle \longrightarrow \langle e', E', s'\rangle}{\langle x{=}e, E, s\rangle \longrightarrow \langle x{=}e', E', s'\rangle}\ \textsc{as1} \qquad \frac{E(x){=}n \quad s(n){=}v}{\langle x{=}n', E, s\rangle \longrightarrow \langle n', E, s + [n \mapsto n']\rangle}\ \textsc{as2}$$

$$\frac{x \notin \mathbf{dom}\,(E) \quad n \notin \mathbf{dom}\,(s) \quad \neg\exists n' < n.\ n' \notin \mathbf{dom}(s)}{\langle\{\,\mathbf{int}\,x;\, e\}, E, s\rangle \longrightarrow \langle e; \mathbf{kill}\,x, E + [x \mapsto n], s + [n \mapsto \mathbf{undef}]\rangle}\ \textsc{local}$$

$$\frac{\langle e_1, E, s\rangle \longrightarrow \langle e_1', E', s'\rangle}{\langle e_1; e_2, E, s\rangle \longrightarrow \langle e_1'; e_2, E', s'\rangle}\ \textsc{seq1} \qquad \frac{}{\langle n; e, E, s\rangle \longrightarrow \langle e, E, s\rangle}\ \textsc{seq2}$$

$$\frac{\langle e, E, s\rangle \longrightarrow \langle e', E', s'\rangle}{\langle f(e), E, s\rangle \longrightarrow \langle f(e'), E', s'\rangle}\ \textsc{cl1} \qquad \frac{\mathbf{int}\,f(\,\mathbf{int}\,x)\{e\} \in p}{\langle f(n), E, s\rangle \longrightarrow \langle\{\,\mathbf{int}\,x; (x{=}n;\, e)\}, E, s\rangle}\ \textsc{cl2}$$

(*a*) For the configuration $\langle \mathrm{g}(3), \{\,\}, \{\,\}\rangle$ and program $\mathbf{int}\,\mathrm{g}(\,\mathbf{int}\,\mathrm{y})\{\{\,\mathbf{int}\,\mathrm{z}; \mathrm{z}{=}\mathrm{y}\}\}$, give the sequence of 11 configurations it transitions to. For each transition, include the list of rule names involved in its derivation, but not the derivation itself. [9 marks]

(*b*) For each of the following, briefly explain the key points of its tinyC semantics and what it illustrates, referring to the transitions and rules, and to the relationship between tinyC and the full C language, as appropriate.

(*i*) $\langle\{\,\mathbf{int}\,\mathrm{y}; \mathrm{g}(\mathrm{y})\}, \{\,\}, \{\,\}\rangle$. [3 marks]

(*ii*) $\langle\{\,\mathbf{int}\,\mathrm{y}; 4\}; \mathrm{y}, \{\,\}, \{\,\}\rangle$ [3 marks]

(*iii*) $\langle \mathrm{h}(5), \{\,\}, \{\,\}\rangle$, with the program $\mathbf{int}\,\mathrm{h}(\,\mathbf{int}\,\mathrm{y})\{\mathrm{y}{=}6; \mathrm{y}\}$, [3 marks]

(*iv*) $\langle\{\,\mathbf{int}\,\mathrm{y}; (\mathrm{y}{=}3; \{\,\mathbf{int}\,\mathrm{y}; \mathrm{y}{=}4\}); \mathrm{y}\}, \{\,\}, \{\,\}\rangle$. [2 marks]