GrumpyIR Dynamic Semantics

April 5, 2021

The GrumpyIR dynamic (big-step operational) semantics is given as a six-place relation between a function environment δ, a variable environment ρ, heap stores μ and μ', an expression e, and value v, written $\delta \mid \rho \vdash e \mid \mu \Downarrow v \mid \mu'$, pronounced "under δ and ρ, in memory state μ, e evaluates to v resulting in new state μ'". Formally, the dynamic semantics is taken to be the smallest relation satisfying the following transition rules:

Values and variables

$$\textsc{E-val}\quad \frac{}{\delta \mid \rho \vdash v \mid \mu \Downarrow v \mid \mu} \qquad\qquad \textsc{E-var}\quad \frac{\rho(x) = v}{\delta \mid \rho \vdash x \mid \mu \Downarrow v \mid \mu}$$

A value evaluates to itself and a variable x steps to a value v whenever ρ maps x to v.

Unary and binary operators

$$\textsc{E-neg}\quad \frac{\delta \mid \rho \vdash e \mid \mu \Downarrow b \mid \mu' \qquad b \in \{\text{true}, \text{false}\}}{\delta \mid \rho \vdash (\text{neg } e) \mid \mu \Downarrow \neg b \mid \mu'}$$

$$\textsc{E-binop}\quad \frac{\begin{array}{c}\delta \mid \rho \vdash e_1 \mid \mu \Downarrow n_1 \mid \mu' \\ \delta \mid \rho \vdash e_2 \mid \mu' \Downarrow n_2 \mid \mu'' \qquad n_1 \, b \, n_2 = n \qquad b \in \{+, *, -, /\}\end{array}}{\delta \mid \rho \vdash (b \; e_1 \; e_2) \mid \mu \Downarrow n \mid \mu''}$$

Let-expressions and sequencing

$$\textsc{E-let}\quad \frac{\delta \mid \rho \vdash e_1 \mid \mu \Downarrow v_1 \mid \mu' \qquad \delta \mid [x \mapsto v_1]\rho \vdash e_2 \mid \mu' \Downarrow v_2 \mid \mu''}{\delta \mid \rho \vdash (\text{let } x \; e_1 \; e_2) \mid \mu \Downarrow v_2 \mid \mu''}$$

$$\textsc{E-seq}\quad \frac{\delta \mid \rho \vdash e_1 \mid \mu \Downarrow v_1 \mid \mu' \qquad \delta \mid \rho \vdash e_2 \mid \mu' \Downarrow v_2 \mid \mu''}{\delta \mid \rho \vdash (\text{seq } e_1 \; e_2) \mid \mu \Downarrow v_2 \mid \mu''}$$

Arrays

E-ALLOC

$$\frac{\delta \mid \rho \vdash e_{size} \mid \mu \Downarrow n \mid \mu' \qquad \delta \mid \rho \vdash e_{init} \mid \mu' \Downarrow v_{init} \mid \mu'' \qquad l \text{ is a freshly allocated store location} \qquad 0 \leq n \qquad \forall i, v_i = v_{init}}{\delta \mid \rho \vdash (\text{alloc } e_{size}\ e_{init}) \mid \mu \Downarrow l \mid [l \mapsto (v_1, v_2, ..., v_n)]\mu''}$$

E-SET

$$\frac{\delta \mid \rho \vdash e_{arr} \mid \mu \Downarrow l \mid \mu' \qquad \delta \mid \rho \vdash e_{ix} \mid \mu' \Downarrow i \mid \mu'' \qquad \delta \mid \rho \vdash e \mid \mu'' \Downarrow v \mid \mu''' \qquad 0 \leq i \qquad \forall j \neq i, v_j = \mu'''(l)_j}{\delta \mid \rho \vdash (\text{set } e_{arr}\ e_{ix}\ e) \mid \mu \Downarrow \text{tt} \mid [l \mapsto (v_1, ..., v_{i-1}, v, v_{i+1}, ..., v_n)]\mu'''}$$

E-GET

$$\frac{\delta \mid \rho \vdash e_{arr} \mid \mu \Downarrow l \mid \mu' \qquad \delta \mid \rho \vdash e_{ix} \mid \mu' \Downarrow i \mid \mu'' \qquad 0 \leq i \qquad \mu''(l)_i = v}{\delta \mid \rho \vdash (\text{get } e_{arr}\ e_{ix}) \mid \mu \Downarrow v \mid \mu''}$$

Conditionals and function calls

E-COND-TRUE

$$\frac{\delta \mid \rho \vdash e_{cond} \mid \mu \Downarrow \text{true} \mid \mu' \qquad \delta \mid \rho \vdash e_1 \mid \mu' \Downarrow v_1 \mid \mu''}{\delta \mid \rho \vdash (\text{cond } e_{cond}\ e_1\ e_2) \mid \mu \Downarrow v_1 \mid \mu}$$

E-COND-FALSE

$$\frac{\delta \mid \rho \vdash e_{cond} \mid \mu \Downarrow \text{false} \mid \mu' \qquad \delta \mid \rho \vdash e_2 \mid \mu' \Downarrow v_2 \mid \mu''}{\delta \mid \rho \vdash (\text{cond } e_{cond}\ e_1\ e_2) \mid \mu \Downarrow v_2 \mid \mu}$$

E-CALL

$$\frac{\delta \mid \rho \vdash e_{fun} \mid \mu \Downarrow p \mid \mu_0 \qquad \delta \mid \rho \vdash e_i \mid \mu_{i-1} \Downarrow v_i \mid \mu_i \qquad \delta(p) = ((x_1, x_2, ..., x_n), e_{body}) \qquad \delta \mid [x_i \mapsto v_i]\rho_0 \vdash e_{body} \mid \mu_n \Downarrow v \mid \mu'}{\delta \mid \rho \vdash (\text{call } e_{fun}\ e_1\ e_2\ ...\ e_n) \mid \mu \Downarrow v \mid \mu'}$$

where ρ_0 is the empty variable environment.