Identifiers

$$
\begin{array}{lcll}
x & \in & \mathbf{TmVar} & \text{(term variables)} \\
\alpha & \in & \mathbf{TyVar} & \text{(type variables)} \\
\beta & \in & \mathbf{SchemeVar} & \text{(scheme variables)} \\
\gamma & \in & \mathbf{AnnVar} & \text{(annotation variables)} \\
\eta & \in & \mathcal{L} & \text{(annotation value)}
\end{array}
$$

Terms

$$
\begin{array}{rcl}
\mathbf{k} & ::= & K^{\nu}\ \overline{x_i} \quad |\ x@(K^{\nu}\ \overline{x_i}) \\
\mathbf{c} & ::= & \mathbf{k} \rightarrow \mathbf{e} \\
\mathbf{e} & ::= & x \ \ |\ \lambda^{\nu}\ x \rightarrow \mathbf{e} \\
 & | & \mathbf{e}\ x \ \ |\ \mathit{seq}\ \mathbf{e_1}\ \mathbf{e_2} \\
 & | & \mathbf{let}\ \overline{x_i =^{\nu,\delta} \mathbf{e_i}}\ \mathbf{in}\ \mathbf{e} \\
 & | & \mathbf{k} \ \ |\ \mathbf{case}\ x\ \mathbf{of}\ \overline{\mathbf{c_i}} \\
\mathbf{ec} & ::= & \mathbf{e} \ \ |\ \mathbf{c}
\end{array}
$$

Types

$$
\begin{array}{rcl}
\tau & ::= & \alpha \ \ |\ @K \ \ |\ T\ \overline{\nu_l}\ \overline{\tau_k} \ \ |\ \rho_{\tau} \rightarrow \phi_{\tau} \\
\sigma & ::= & \beta \ \ |\ \forall \{\overline{\alpha}, \overline{\gamma}\}\ .\ C \Rightarrow \tau \\
\phi_{\mu} & ::= & \mu^{\nu} \\
\rho_{\mu} & ::= & {\phi_{\mu}}^{\delta} \text{(also written as: } \mu^{\nu,\delta}) \\
\nu, \delta & ::= & \gamma \ \ |\ \eta \\
\Gamma & ::= & \emptyset \ \ |\ \Gamma, x : \rho_{\sigma}
\end{array}
$$

References

[Hage and Holdermans(2008)] J. Hage and S. Holdermans. Heap recycling for lazy languages. In *Proceedings of the 2008 ACM SIGPLAN symposium on Partial evaluation and semantics-based program manipulation*, pages 189–197. ACM, 2008.

Generic analysis:

symbol	definition
$\mathcal{L}$	$\mathcal{P}(\mathbb{N})$
$\mathbb{0}$	$\{0\}$
$\mathbb{1}$	$\{1\}$
$\top$	$\mathbb{N}$
$\bot$	$\emptyset$
$x \oplus y$	$\{m+n \mid m \in x, n \in y\}$
$x \ominus y$	$\bigcap_{n \in y}\{m-n \mid m \in x, m \geq n\}$
$x \sqsubseteq y$	$x \subseteq y$
$x \sqcup y$	$x \cup y$
$x \cdot y$	$\{\sum_{i=1}^{m} n_i \mid m \in x, \forall i. n_i \in y\}$
$x \triangleright y$	$\bigcup_{m \in x}(m \equiv 0 \mathbin{?} \mathbb{0} : y)$

Using it as an sharing or strictness analysis:

$$x \diamond y = x \sqsubseteq y$$

Using it as an uniqueness analysis:

$$x \diamond y = x \equiv y$$

Specializing an analysis:

Symbols with subscript g denote the generic version of the symbol. Requires a specialized lattice $\mathcal{L}$ and the functions $p : \mathcal{L}_g \to \mathcal{L}$ and $q : \mathcal{L} \to \mathcal{L}_g$.

symbol	definition
$\mathbb{0}$	$p\ (\mathbb{0}_g)$
$\mathbb{1}$	$p\ (\mathbb{1}_g)$
$\top$	$p\ (\top_g)$
$\bot$	$p\ (\bot_g)$
$x \oplus y$	$p\ (q\ (x) \oplus q\ (y))$
$x \ominus y$	$p\ (q\ (x) \ominus q\ (y))$
$x \sqsubseteq y$	$q\ (x) \sqsubseteq q\ (y)$
$x \sqcup y$	$p\ (q\ (x) \sqcup q\ (y))$
$x \cdot y$	$p\ (q\ (x) \cdot q\ (y))$
$x \triangleright y$	$p\ (q\ (x) \triangleright q\ (y))$

Example:

Uniqueness analysis with $\mathcal{L} = \{\bot, 0, 1, \omega\}$.

$$x \diamond y \qquad x \equiv y$$

$$p(x_g) \quad \begin{cases} \omega & \exists m \in x_g. m \geq 2 \\ 1 & 1 \in x_g \\ 0 & 0 \in x_g \\ \bot & \text{otherwise} \end{cases}$$

$$q(x) \quad \begin{cases} \mathbb{N} & x \equiv \omega \\ \{0,1\} & x \equiv 1 \\ \{0\} & x \equiv 0 \\ \emptyset & \text{otherwise} \end{cases}$$

Static semantics $\boxed{\Gamma \vdash \mathbf{e} : \phi_\tau}$

$$\frac{\tau_1 \preccurlyeq \tau_2}{x : {\tau_2}^{\nu, \mathbb{1}} \vdash x : {\tau_1}^{\nu}} \text{ VAR}$$

$$\frac{\Gamma, x : (\forall \emptyset \,.\, \emptyset \Rightarrow \tau)^{\nu_1, \delta_1} \vdash \mathbf{e} : \phi}{\nu_2 \cdot \Gamma \vdash \lambda^{\nu_2}\ x \to \mathbf{e} : (\tau^{\nu_1, \delta_1} \to \phi)^{\nu_2}} \text{ ABS}$$

$$\frac{\Gamma \vdash_{\sqsubseteq} \mathbf{e} : ({\phi_2}^{\delta_2} \to \phi_3)^{\mathbb{1}} \qquad x : {\phi_4}^{\mathbb{1}} \vdash_{\diamond} x : \phi_2}{\Gamma \oplus x : {\phi_4}^{\delta_2} \vdash \mathbf{e}\ x : \phi_3} \text{ APP}$$

$$\frac{\begin{array}{c} \Gamma_i, \overline{x_j : \rho_{ij}} \vdash \mathbf{e_i} : {\tau_i}^{\nu_i} \qquad \Gamma_0, \overline{x_j : {\sigma_{0j}}^{\nu_{0j}, \delta_{0j}}} \vdash \mathbf{e} : \phi \\ (\forall \emptyset \,.\, \emptyset \Rightarrow \tau_i)^{\nu_i, \delta_i} = ??^{\nu_{0i}, \delta_{0i}} \oplus (\oplus_j\ (\delta_j \rhd \rho_{ji})) \qquad \Gamma = \Gamma_0 \oplus (\oplus_j\ (\delta_j \rhd \Gamma_j)) \\ C = \text{implicit left-over constraints} \qquad \sigma_{0i} = \mathit{gen}\ ({\tau_i}^{\nu_i, \delta_i}, C, \Gamma) \end{array}}{\Gamma \vdash \mathbf{let}\ \overline{x_i =^{\nu_i, \delta_i} \mathbf{e_i}}\ \mathbf{in}\ \mathbf{e} : \phi} \text{ LET}$$

Subeffecting $\boxed{\Gamma \vdash_{\backsim} \mathbf{e} : \phi_\tau}$

$$\frac{\Gamma \vdash \mathbf{e} : \tau^{\nu_2} \qquad \nu_1 \backsim \nu_2}{\Gamma \vdash_{\backsim} \mathbf{e} : \tau^{\nu_1}} \text{ SUB}(\backsim)$$

Extension: seq $\boxed{\Gamma \vdash \mathbf{e} : \phi_\tau}$

$$\frac{\Gamma_1 \vdash_{\sqsubseteq} \mathbf{e_1} : {\tau_1}^{\mathbb{0}} \qquad \Gamma_2 \vdash \mathbf{e_2} : \phi_2}{\Gamma_1 \oplus \Gamma_2 \vdash \mathit{seq}\ \mathbf{e_1}\ \mathbf{e_2} : \phi_2} \text{ SEQ}$$

Extension: datatypes $\boxed{\Gamma \vdash \mathbf{ec} : \phi_\tau}$

$$\frac{\Gamma_0 \vdash x : (T\ \overline{\nu_l}\ \overline{\tau_k})^{\nu} \qquad \Gamma_i \vdash \mathbf{c_i} : ((T\ \overline{\nu_l}\ \overline{\tau_k})^{\nu, \mathbb{1}} \to \phi)^{\mathbb{1}}}{\Gamma_0 \oplus (\sqcup_i\ \Gamma_i) \vdash \mathbf{case}\ x\ \mathbf{of}\ \overline{\mathbf{c_i}} : \phi} \text{ CASE}$$

$$\frac{\begin{array}{c} \mathbf{data}\ T\ \overline{u_l}\ \overline{\alpha_k} = \overline{K_i\ \overline{\rho_{ij}}} \\ \dfrac{{\tau_j}^{\nu_j, \delta_j} = \rho_{ij}\ [\overline{\nu_l}/\overline{u_l}, \overline{\tau_k}/\overline{\alpha_k}]}{\Gamma, x_j : (\forall \emptyset \,.\, \emptyset \Rightarrow \tau_j)^{\nu_j, \delta_j} \vdash \mathbf{e} : \phi} \quad x_j : {\tau'_j}^{\nu_j, \mathbb{1}} \vdash x_j : {\tau_j}^{\nu_j} \quad \mathbb{1} \sqsubseteq \nu \end{array}}{\Gamma \vdash {K_i}^{\nu}\ \overline{x_j} \to \mathbf{e} : ((T\ \overline{\nu_l}\ \overline{\tau_k})^{\nu, \mathbb{1}} \to \phi)^{\mathbb{1}}} \text{ CASE-ARM}$$

$$\frac{\begin{array}{c} \mathbf{data}\ T\ \overline{u_l}\ \overline{\alpha_k} = \overline{K_i\ \overline{\rho_{ij}}} \\ {\phi_j}^{\delta_j} = \rho_{ij}\ [\overline{\nu_l}/\overline{u_l}, \overline{\tau_k}/\overline{\alpha_k}] \qquad x_j : {\phi'_j}^{\mathbb{1}} \vdash x_j : \phi_j \end{array}}{\oplus_j\ (x_j : {\phi'_j}^{\delta_j}) \vdash {K_i}^{\nu}\ \overline{x_j} : (T\ \overline{\nu_l}\ \overline{\tau_k})^{\nu}} \text{ CON}$$

Extension: heap labels $\boxed{\Gamma \vdash \mathbf{ec} : \phi_\tau}$

$$\frac{\Gamma, x : (\forall \emptyset \,.\, \emptyset \Rightarrow @K)^{\nu_1, \delta_1} \vdash (K^{\nu}\ \overline{x_i}) \to \mathbf{e} : \phi \qquad \nu_1 \sqsubseteq \mathbb{1} \qquad \nu \sqsubseteq \mathbb{1}}{\Gamma \vdash x@(K^{\nu}\ \overline{x_i}) \to \mathbf{e} : \phi} \text{ CASE-ARM-LBL}$$

$$\frac{\Gamma_1 \vdash K^{\nu}\ \overline{x_i} : \phi \qquad \Gamma_2 \vdash_{\sqsubseteq} x : (@K)^{\mathbb{1}}}{\Gamma_1 \oplus \Gamma_2 \vdash x@(K^{\nu}\ \overline{x_i}) : \phi} \text{ CON-LBL}$$

Operational semantics must verify the annotations: during execution none of the thunk annotations may go below zero and after execution all thunk annotation must permit zero use. Operational semantics must also use heap recycling and update avoidance.

Operational semantics $\boxed{\langle H; \mathbf{e}; S\rangle \rightarrowtail \langle H; \mathbf{e}; S\rangle}$

$$\frac{\nu \ominus \mathit{use}\ (S) \not\equiv \bot \qquad \delta \ominus \mathbb{1} \not\equiv \bot}{\langle H, x =^{\nu,\delta} \mathbf{e}; x; S\rangle \rightarrowtail \langle H; \mathbf{e}; \#^{\nu \ominus \mathit{use}\ (S), \delta \ominus \mathbb{1}}\ x, S\rangle}\ \text{VAR}$$

$$\frac{\mathbb{1} \sqsubseteq \nu}{\langle H; \lambda^{\nu}\ x_1 \rightarrow \mathbf{e}; [\cdot]\ x_2, S\rangle \rightarrowtail \langle H; \mathbf{e}\ [{}^{x_2}/{}_{x_1}]; S\rangle}\ \text{ABS}$$

$$\frac{}{\langle H; \mathbf{e}\ x; S\rangle \rightarrowtail \langle H; \mathbf{e}; [\cdot]\ x, S\rangle}\ \text{APP}$$

$$\frac{\mathit{fresh}\ \overline{y_i} \qquad \phi = [{}^{\overline{y_i}}/{}_{\overline{x_i}}]}{\langle H; \mathbf{let}\ \overline{x_i =^{\nu_i,\delta_i} \mathbf{e_i}}\ \mathbf{in}\ \mathbf{e}; S\rangle \rightarrowtail \langle H, \overline{y_i =^{\nu_i,\delta_i} \mathbf{e_i}\ [\phi]}; \mathbf{e}\ [\phi]; S\rangle}\ \text{LET}$$

$$\frac{\nu_1 \ominus \nu_2 \not\equiv \bot}{\langle H;\ V^{\nu_1}; \#^{\nu_2,\delta_2}\ x, S\rangle \rightarrowtail \langle H, x =^{\nu_2,\delta_2}\ V^{\nu_2};\ V^{\nu_1 \ominus \nu_2}; S\rangle}\ \text{UPDATE}$$

$$\frac{}{\langle H; \mathit{seq}\ \mathbf{e_1}\ \mathbf{e_2}; S\rangle \rightarrowtail \langle H; \mathbf{e_1}; \mathit{seq}\ [\cdot]\ \mathbf{e_2}, S\rangle}\ \text{SEQ1}$$

$$\frac{\mathbb{0} \sqsubseteq \nu}{\langle H;\ V^{\nu}; \mathit{seq}\ [\cdot]\ \mathbf{e}, S\rangle \rightarrowtail \langle H; \mathbf{e}; S\rangle}\ \text{SEQ2}$$

$$\frac{}{\langle H; \mathbf{case}\ x\ \mathbf{of}\ \overline{\mathbf{c_i}}; S\rangle \rightarrowtail \langle H; x; \mathbf{case}\ x@[\cdot]\ \mathbf{of}\ \overline{\mathbf{c_i}}, S\rangle}\ \text{CASE}$$

$$\frac{\mathbb{1} \sqsubseteq \nu \qquad \mathbf{k_i} = \overline{K_i}\ \nu\ \overline{y_j}}{\langle H; \overline{K_i}\ \nu\ \overline{x_j}; \mathbf{case}\ x@[\cdot]\ \mathbf{of}\ \overline{\mathbf{k_i} \rightarrow \mathbf{e_i}}, S\rangle \rightarrowtail \langle H; \mathbf{e_i}\ [{}^{\overline{x_j}}/{}_{\overline{y_j}}]; S\rangle}\ \text{CON1}$$

$$\frac{\mathbb{1} \equiv \nu \qquad \mathbf{k_i} = y@(\overline{K_i}\ \nu\ \overline{y_j})}{\langle H, x =^{\mathbb{0},\mathbb{0}}\ V; \overline{K_i}\ \nu\ \overline{x_j}; \mathbf{case}\ x@[\cdot]\ \mathbf{of}\ \overline{\mathbf{k_i} \rightarrow \mathbf{e_i}}, S\rangle \rightarrowtail \langle H, x =^{??}\ V; \mathbf{e_i}\ [{}^{\overline{x_j}}/{}_{\overline{y_j}}, {}^{x}/{}_{y}]; S\rangle}\ \text{CON2}$$

$$\frac{}{\langle H, x =^{\nu_1,\delta_1}\ \mathbf{e}; x@(K^{\nu_2}\ \overline{x_j}); S\rangle \rightarrowtail \langle H, x =^{\nu_2,\delta_1}\ K^{\nu_2}\ \overline{x_j}; K^{\nu_2}\ \overline{x_j}; S\rangle}\ \text{LBL}$$

Evaluating to values $\boxed{H_1; \mathbf{e} \Downarrow_n H_2; v}$

$$\frac{H\ (x)\ \in\ v}{H; x \Downarrow_0 H; H\ (x)}\ \textsc{VarValueValue}$$

$$\frac{H_1\ (x)\ \in\ \mathbf{e} - v \qquad H_1; H_1\ (x) \Downarrow_n H_2; v}{H_1; x \Downarrow_n H_2, x = v; v}\ \textsc{VarTermValue}$$

$$\frac{}{H; \lambda x \to \mathbf{e} \Downarrow_0 H; \lambda x \to \mathbf{e}}\ \textsc{AbsValue}$$

$$\frac{\begin{array}{c} H_1; \mathbf{e_1} \Downarrow_{n_1} H_2; \lambda x_1 \to \mathbf{e_3} \\ H_2; \mathbf{e_2} \Uparrow_{n_2} H_3; x_2 \qquad H_3; \mathbf{e_3}\ [^{x_2}/_{x_1}] \Downarrow_{n_3} H_4; x_3 \end{array}}{H_1; \mathbf{e_1}\ \mathbf{e_2} \Downarrow_{n_1+n_2+n_3} H_4; x_3}\ \textsc{AppValue}$$

$$\frac{\textit{fresh}\ \overline{y_i} \qquad \phi = [^{\overline{y_i}}/_{\overline{x_i}}] \qquad H_1, \overline{y_i = \mathbf{e_i}\ [\phi]}; \mathbf{e}\ [\phi] \Downarrow_n H_2; x}{H_1; \mathbf{let}\ \overline{x_i = \mathbf{e_i}}\ \mathbf{in}\ \mathbf{e} \Downarrow_{n+\#\overline{x_i}} H_2; x}\ \textsc{LetValue}$$

$$\frac{i \in \{1 \ldots n\} \qquad H_i; \mathbf{e_i} \Uparrow_{n_i} H_{i+1}; x_i}{H_1; K\ \overline{\mathbf{e_i}} \Downarrow_{\sum\ \overline{n_i}} H_{n+1}; K\ \overline{x_i}}\ \textsc{ConValue}$$

$$\frac{H_1; \mathbf{e} \Downarrow_n H_2; v}{H_1; x@\mathbf{e} \Downarrow_n H_2; v}\ \textsc{LblValue}$$

$$\frac{H_1; \mathbf{e} \Downarrow_{n_1} H_2; K_i\ \overline{x_{ij}} \qquad H_2; \mathbf{e_i}\ [^{\overline{x_{ij}}}/_{\overline{y_{ij}}}] \Downarrow_{n_2} H_3; v}{H_1; \mathbf{case}\ \mathbf{e}\ \mathbf{of}\ \overline{K_i\ \overline{y_{ij}} \to \mathbf{e_i}} \Downarrow_{n_1+n_2} H_3; v}\ \textsc{NormalCaseValue}$$

$$\frac{\begin{array}{c} H_1; \mathbf{e} \Uparrow_{n_1} H_2; x \\ H_2\ (x) = K_i\ \overline{x_{ij}} \qquad H_2; \mathbf{e_i}\ [^{\overline{x_{ij}}}/_{\overline{y_{ij}}}, {}^{x}/_{y_i}] \Downarrow_{n_2} H_3; v \end{array}}{H_1; \mathbf{case}\ \mathbf{e}\ \mathbf{of}\ \overline{y_i@(K_i\ \overline{y_{ij}}) \to \mathbf{e_i}} \Downarrow_{n_1+n_2} H_3; v}\ \textsc{UpdateCaseValue}$$

Figure 1: Heap recycling - operational semantics part 1

Evaluating to heap locations $\boxed{H_1; \mathbf{e} \Uparrow_n H_2; x}$

$$\frac{H_1; x \Downarrow_n H_2; v}{H_1; x \Uparrow_n H_2; x} \textsc{VarHeap}$$

$$\frac{\textit{fresh } x_2}{H; \lambda x_1 \to \mathbf{e} \Uparrow_1 H, x_2 = \lambda x_1 \to \mathbf{e}; x_2} \textsc{AbsHeap}$$

$$\frac{\begin{array}{c} H_1; \mathbf{e_1} \Downarrow_{n_1} H_2; \lambda x_1 \to \mathbf{e_3} \\ H_2; \mathbf{e_2} \Uparrow_{n_2} H_3; x_2 \qquad H_3; \mathbf{e_3} \left[{}^{x_2}/_{x_1}\right] \Uparrow_{n_3} H_4; x_3 \end{array}}{H_1; \mathbf{e_1}\ \mathbf{e_2} \Uparrow_{n_1+n_2+n_3} H_4; x_3} \textsc{AppHeap}$$

$$\frac{\textit{fresh } \overline{y_i} \qquad \phi = \left[{}^{\overline{y_i}}/_{\overline{x_i}}\right] \qquad H_1, \overline{y_i = \mathbf{e_i}\ [\phi]}; \mathbf{e}\ [\phi] \Uparrow_n H_2; x}{H_1; \mathbf{let}\ \overline{x_i = \mathbf{e_i}}\ \mathbf{in}\ \mathbf{e} \Uparrow_{n+\#\overline{x_i}} H_2; x} \textsc{LetHeap}$$

$$\frac{\textit{fresh } x \qquad H_1; K\ \overline{\mathbf{e_i}} \Downarrow_n H_2; K\ \overline{x_i}}{H_1; K\ \overline{\mathbf{e_i}} \Uparrow_{n+1} H_2, x = K\ \overline{x_i}; x} \textsc{ConHeap}$$

$$\frac{H_1; \mathbf{e} \Downarrow_n H_2; v}{H_1; x@\mathbf{e} \Uparrow_n H_2, x = v; x} \textsc{LblHeap}$$

$$\frac{H_1; \mathbf{e} \Downarrow_{n_1} H_2; K_i\ \overline{x_{ij}} \qquad H_2; \mathbf{e_i} \left[{}^{\overline{x_{ij}}}/_{\overline{y_{ij}}}\right] \Uparrow_{n_2} H_3; x}{H_1; \mathbf{case}\ \mathbf{e}\ \mathbf{of}\ \overline{K_i\ \overline{y_{ij}} \to \mathbf{e_i}} \Uparrow_{n_1+n_2} H_3; x} \textsc{NormalCaseHeap}$$

$$\frac{\begin{array}{c} H_1; \mathbf{e} \Uparrow_{n_1} H_2; x \\ H_2\ (x) = K_i\ \overline{x_{ij}} \qquad H_2; \mathbf{e_i} \left[{}^{\overline{x_{ij}}}/_{\overline{y_{ij}}}, {}^{x}/_{y_i}\right] \Uparrow_{n_2} H_3; z \end{array}}{H_1; \mathbf{case}\ \mathbf{e}\ \mathbf{of}\ \overline{y_i@(K_i\ \overline{y_{ij}}) \to \mathbf{e_i}} \Uparrow_{n_1+n_2} H_3; z} \textsc{UpdateCaseHeap}$$

Figure 2: Heap recycling - operational semantics part 2

The following code recycles all the heap for any given list using the operational semantics from figures 1 and 2. A proper static semantics is still required.

$$\begin{array}{l} id = \lambda xs \to \mathbf{case}\ xs\ \mathbf{of} \\ \quad h@Nil \to h@Nil \\ \quad h@(Cons\ y\ ys) \to h@(Cons\ y\ (id\ ys)) \end{array}$$

However, this code is not allowed in [Hage and Holdermans(2008)]. The closest in can come to a similar function is the following code:

$$\begin{array}{l} id = \lambda xs \to \mathbf{case}\ xs\ \mathbf{of} \\ \quad Nil \to Nil \\ \quad Cons\ y\ ys \to \mathbf{let}\ r = id\ ys\ \mathbf{in}\ (\lambda x \to x)\ (xs@(Cons\ y\ r)) \end{array}$$

However, this function does one allocation for every Cons cell (the **let**). Also, the constructs needed to achieve this seem to be a little verbose.