Nob output

Nob (Nicholas Cameron)

March 11, 2011

Abstract

Generated by Nob v0.1, written by Nicholas Cameron

P	::=	$\overline{Q}\ e$	*programs*
Q	::=	$\texttt{class C} \lhd \texttt{D}\ \{\ \overline{T\ \texttt{f;}}\ \overline{M}\ \}$	*class definitions*
M	::=	$T\ \texttt{m(}\overline{T\ \texttt{x}}\texttt{)\{ return}\ e\texttt{;\}}$	*method definitions*
T, U	::=	$\texttt{C}$	*types*
e	::=	$\texttt{x} \mid \texttt{new C(}\overline{e}\texttt{)} \mid e\texttt{.f} \mid e\texttt{.m(}\overline{e}\texttt{)}$	*expressions*
v	::=	$\texttt{new C(}\overline{v}\texttt{)}$	*values*
$\texttt{x}, \texttt{y}, \texttt{z}, \texttt{this}$			*variables*
$\texttt{C}, \texttt{D}, \texttt{Object}$			*class names*
$\texttt{f}$			*field names*
$\texttt{m}$			*method names*
Γ	::=	$\overline{\texttt{x:}T}$	*environments*
$\mathcal{P}$	::=	$\overline{\texttt{C:}Q}$	*programs*

Figure 1: Syntax of Calculus.

$\boxed{fields(\texttt{C}) = \overline{\texttt{f}}; \overline{T}}$

$$\frac{}{fields(\texttt{Object}) = \emptyset; \emptyset}$$

$$\frac{\mathcal{P}(\texttt{C}) = \texttt{class C} \lhd \texttt{D}\ \{\ \overline{T\ \texttt{f;}}\ \overline{M}\ \} \qquad fields(\texttt{D}) = \overline{\texttt{f}'}; \overline{T'}}{fields(\texttt{C}) = \overline{\texttt{f}}, \overline{\texttt{f}'}; \overline{T}, \overline{T'}}$$

Figure 2:

$\boxed{fType(\mathtt{f},\mathtt{C}) = T}$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ \overline{T\ \mathtt{f;}}\ ...\ \}}{fType((\mathtt{f_i}),\mathtt{C}) = (T_\mathtt{i})}$$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ \overline{T\ \mathtt{f;}}\ ...\ \} \qquad \mathtt{f} \notin \overline{\mathtt{f}} \qquad fType(\mathtt{f},\mathtt{D}) = T}{fType(\mathtt{f},\mathtt{C}) = T}$$

Figure 3:

$\boxed{mType(\mathtt{m},\mathtt{C}) = \overline{T}; T}$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ ...\ \overline{M}\ \} \qquad T\ \mathtt{m}(\overline{T\ \mathtt{x}})... \in \overline{M}}{mType(\mathtt{m},\mathtt{C}) = \overline{T}; T}$$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ ...\ \overline{M}\ \} \qquad U\ \mathtt{m}\ ... \notin \overline{M} \qquad mType(\mathtt{m},\mathtt{D}) = \overline{T}; T}{mType(\mathtt{m},\mathtt{C}) = \overline{T}; T}$$

Figure 4:

$\boxed{mBody(\mathtt{m},\mathtt{C}) = \overline{\mathtt{x}}; e}$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ ...\ \overline{M}\ \} \qquad T\ \mathtt{m}(\overline{T\ \mathtt{x}})\{\ \mathtt{return}\ e\mathtt{;}\} \in \overline{M}}{mBody(\mathtt{m},\mathtt{C}) = \overline{\mathtt{x}}; e}$$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ \{\ ...\ \overline{M}\ \} \qquad T\ \mathtt{m}\ ... \notin \overline{M} \qquad mBody(\mathtt{m},\mathtt{D}) = \overline{\mathtt{x}}; e}{mBody(\mathtt{m},\mathtt{C}) = \overline{\mathtt{x}}; e}$$

Figure 5:

$\boxed{override(\mathtt{m},\mathtt{C},\overline{T},T)}$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ ... \qquad mType(\mathtt{m},\mathtt{D}) = \overline{U}; U\ \textit{undefined}}{override(\mathtt{m},\mathtt{C},\overline{T},T)}$$

$$\frac{\mathcal{P}(\mathtt{C}) = \mathtt{class\ C} \lhd \mathtt{D}\ ... \qquad mType(\mathtt{m},\mathtt{D}) = \overline{T}; T}{override(\mathtt{m},\mathtt{C},\overline{T},T)}$$

Figure 6:

Subtyping $\boxed{\vdash T <: T}$

$$\frac{}{\vdash T <: T}$$

(S-Reflex)

$$\frac{\vdash T_1 <: T_3 \qquad \vdash T_2 <: T_3}{\vdash T_1 <: T_2}$$

(S-Trans)

$$\frac{\mathcal{P}(\texttt{C}) = \texttt{class C} \lhd \texttt{D} \ldots}{\vdash \texttt{C} <: \texttt{D}}$$

(S-Sub-Class)

Figure 7: Calculus subtyping.

Well-formedness $\boxed{\vdash T \text{ OK}}$

$$\frac{\mathcal{P}(\texttt{C}) = \ldots}{\vdash \texttt{C OK}}$$

(F-Class)

Figure 8: Calculus well-formedness.

Type checking $\boxed{\Gamma \vdash e : T}$

$$\frac{\Gamma \vdash e : U \qquad \vdash U <: T}{\Gamma \vdash e : T}$$

(T-Subs)

$$\frac{T = \Gamma(\mathtt{x})}{\Gamma \vdash \mathtt{x} : T}$$

(T-Var)

$$\frac{\mathit{fields}(\mathtt{C}) = \overline{\mathtt{f}}; \overline{T} \qquad \Gamma \vdash \overline{e} : \overline{T}}{\Gamma \vdash \mathtt{new\ C}(\overline{e}) : \mathtt{C}}$$

(T-New)

$$\frac{\Gamma \vdash e : \mathtt{C} \qquad \mathit{fType}(\mathtt{f}, \mathtt{C}) = T}{\Gamma \vdash e.\mathtt{f} : T}$$

(T-Field)

$$\frac{\Gamma \vdash e : \mathtt{C} \qquad \mathit{mType}(\mathtt{m}, \mathtt{C}) = \overline{T}; T \qquad \Gamma \vdash \overline{e} : \overline{T}}{\Gamma \vdash e.\mathtt{m}(\overline{e}) : T}$$

(T-Invk)

Figure 9: Calculus type checking.

Well-typed methods $\boxed{\Gamma \vdash M \ \text{IN}\mathtt{C}}$

$$\frac{\vdash T \ \text{OK} \qquad \vdash \overline{T} \ \text{OK} \qquad \Gamma, \overline{\mathtt{x}{:}T} \vdash e : T \qquad \mathit{override}(\mathtt{m}, \mathtt{C}, \overline{T}, T)}{\Gamma \vdash T\ \mathtt{m}(\overline{T\ \mathtt{x}})\{\ \mathtt{return}\ e;\}\ \text{IN}\mathtt{C}}$$

(T-Method)

Figure 10: Calculus well-typed methods.

Well-typed classes $\boxed{\vdash Q}$

$$\frac{\vdash \mathtt{D} \ \text{OK} \qquad \vdash \overline{T} \ \text{OK} \qquad \mathtt{this{:}C} \vdash \overline{M} \ \text{IN}\mathtt{C}}{\vdash \mathtt{class\ C} \lhd \mathtt{D}\ \{\ \overline{T\ \mathtt{f};}\ \overline{M}\ \}}$$

(T-Class)

Figure 11: Calculus well-typed classes.

Well-typed programs $\boxed{\vdash P}$

$$\frac{\vdash \overline{Q} \qquad \emptyset \vdash e : T \qquad \overline{Q} = \overline{\texttt{class C} \ldots} \qquad \mathcal{P} = \texttt{Object:(class Object} \lhd \texttt{Object} \{\quad\}), \overline{\texttt{C:}Q}}{\vdash \overline{Q}\ e}$$

Figure 12: Calculus well-typed programs.