Lecture 14 — Automatic Type Inference (2)

CSE307: Programming Languages

Minseok Jeon

2026 Spring

Goal

- So far we have informally discussed how to derive type equations.
- In this lecture, we define the procedure precisely.

Language

$$
\begin{array}{rcl}
E & \to & n \\
& | & x \\
& | & E + E \\
& | & E - E \\
& | & \texttt{iszero}\ E \\
& | & \texttt{if}\ E\ \texttt{then}\ E\ \texttt{else}\ E \\
& | & \texttt{let}\ x = E\ \texttt{in}\ E \\
& | & \texttt{proc}\ x\ E \\
& | & E\ E \\
\\
T & \to & \mathsf{int} \\
& | & \mathsf{bool} \\
& | & T \to T \\
& | & \alpha\ (\in \mathit{TyVar})
\end{array}
$$

Type Equations

- Type equations are conjunctions of "type equalities": e.g.,

$$\begin{array}{rcl} \boldsymbol{t}_0 & = & \boldsymbol{t}_{\boldsymbol{f}} \to \boldsymbol{t}_1 \\ \boldsymbol{t}_1 & = & \boldsymbol{t}_{\boldsymbol{x}} \to \boldsymbol{t}_4 \\ \boldsymbol{t}_3 & = & \text{int} \\ \boldsymbol{t}_4 & = & \text{int} \\ \boldsymbol{t}_2 & = & \text{int} \\ \boldsymbol{t}_{\boldsymbol{f}} & = & \text{int} \to \boldsymbol{t}_3 \\ \boldsymbol{t}_{\boldsymbol{f}} & = & \boldsymbol{t}_{\boldsymbol{x}} \to \boldsymbol{t}_4 \end{array}$$

- Type equations (*TyEqn*) are defined inductively:

$$\begin{array}{rcl} \mathit{TyEqn} & \to & \emptyset \\ & \mid & \boldsymbol{T} \doteq \boldsymbol{T} \ \wedge \ \mathit{TyEqn} \end{array}$$

Deriving Type Equations

- Algorithm for generating equations:

$$\mathcal{V} : (Var \to \boldsymbol{T}) \times \boldsymbol{E} \times \boldsymbol{T} \to TyEqn$$

- $\mathcal{V}(\Gamma, e, t)$ generates the condition for e to have type t in Γ:

$$\Gamma \vdash e : t \text{ iff } \mathcal{V}(\Gamma, e, t) \text{ is satisfied.}$$

- Examples:
  - $\mathcal{V}([x \mapsto \text{int}], \texttt{x+1}, \alpha) = \alpha \doteq \text{int}$
  - $\mathcal{V}(\emptyset, \texttt{proc}\ (x)\ (\texttt{if}\ x\ \texttt{then}\ 1\ \texttt{else}\ 2), \alpha \to \beta) = \alpha \doteq \text{bool} \wedge \beta \doteq \text{int}$
- To derive type equations for closed expression E, we call $\mathcal{V}(\emptyset, E, \alpha)$, where α is a fresh type variable.

Deriving Type Equations

$$
\begin{aligned}
\mathcal{V}(\Gamma, n, t) &= t \doteq \mathsf{int} \\
\mathcal{V}(\Gamma, x, t) &= t \doteq \Gamma(x) \\
\mathcal{V}(\Gamma, e_1 + e_2, t) &= t \doteq \mathsf{int} \ \land\ \mathcal{V}(\Gamma, e_1, \mathsf{int}) \ \land\ \mathcal{V}(\Gamma, e_2, \mathsf{int}) \\
\mathcal{V}(\Gamma, \texttt{iszero}\ e, t) &= t \doteq \mathsf{bool} \ \land\ \mathcal{V}(\Gamma, e, \mathsf{int}) \\
\mathcal{V}(\Gamma, \texttt{if}\ e_1\ e_2\ e_3, t) &= \mathcal{V}(\Gamma, e_1, \mathsf{bool}) \ \land\ \mathcal{V}(\Gamma, e_2, t) \ \land\ \mathcal{V}(\Gamma, e_3, t) \\
\mathcal{V}(\Gamma, \texttt{let}\ x = e_1\ \texttt{in}\ e_2, t) &= \mathcal{V}(\Gamma, e_1, \alpha) \ \land\ \mathcal{V}([x \mapsto \alpha]\Gamma, e_2, t)\ (\mathsf{new}\ \alpha) \\
\mathcal{V}(\Gamma, \texttt{proc}\ (x)\ e, t) &= t \doteq \alpha_1 \to \alpha_2 \ \land\ \mathcal{V}([x \mapsto \alpha_1]\Gamma, e, \alpha_2) \\
&\quad (\mathsf{new}\ \alpha_1, \alpha_2) \\
\mathcal{V}(\Gamma, e_1\ e_2, t) &= \mathcal{V}(\Gamma, e_1, \alpha \to t) \ \land\ \mathcal{V}(\Gamma, e_2, \alpha)\ (\mathsf{new}\ \alpha)
\end{aligned}
$$

Example

$$
\begin{aligned}
&\mathcal{V}(\emptyset, (\texttt{proc}\ (x)\ (x))\ 1, \boldsymbol{\alpha}) \\
&= \mathcal{V}(\emptyset, \texttt{proc}\ (x)\ (x), \boldsymbol{\alpha}_1 \to \boldsymbol{\alpha}) \wedge \mathcal{V}(\emptyset, 1, \boldsymbol{\alpha}_1) && \text{new } \boldsymbol{\alpha}_1 \\
&= \boldsymbol{\alpha}_1 \to \boldsymbol{\alpha} \doteq \boldsymbol{\alpha}_2 \to \boldsymbol{\alpha}_3 \wedge \mathcal{V}([x \mapsto \boldsymbol{\alpha}_2], x, \boldsymbol{\alpha}_3) \wedge \boldsymbol{\alpha}_1 \doteq \mathsf{int} && \text{new } \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3 \\
&= \boldsymbol{\alpha}_1 \to \boldsymbol{\alpha} \doteq \boldsymbol{\alpha}_2 \to \boldsymbol{\alpha}_3 \wedge \boldsymbol{\alpha}_2 \doteq \boldsymbol{\alpha}_3 \wedge \boldsymbol{\alpha}_1 \doteq \mathsf{int}
\end{aligned}
$$

Exercise 1

$$\mathcal{V}(\emptyset, \texttt{proc}\ (\boldsymbol{f})\ (\boldsymbol{f}\ \texttt{11}), \boldsymbol{\alpha})$$

Exercise 2

$$\mathcal{V}([x \mapsto \mathsf{bool}], \texttt{if}\ x\ \texttt{then}\ (x - 1)\ \texttt{else}\ 0, \boldsymbol{\alpha})$$

Exercise 3

$$\mathcal{V}(\emptyset, \texttt{proc}\ (f)\ (\texttt{iszero}\ (f\ f)), \alpha)$$

Summary

We have defined the algorithm for deriving type equations from program text:

- Given a program $\boldsymbol{E}$, call $\mathcal{V}(\emptyset, \boldsymbol{E}, \boldsymbol{\alpha})$ to derive type equations.
- Solve the equations and find the type assigned to $\boldsymbol{\alpha}$.