G54FOP: Handout, Lecture 13

Types and Type Systems II

Henrik Nilsson

University of Nottingham, UK

Extension: Let Bindings (1)

Syntactic extension:

$$\begin{array}{rcll} t & \to & \ldots & \textit{terms:} \\ & | & x & \textit{variable} \\ & | & \texttt{let}\ x\ =\ t\ \texttt{in}\ t & \textit{let binding} \end{array}$$

New evaluation rules:

$$\texttt{let}\ x\ =\ v_1\ \texttt{in}\ t_2 \longrightarrow [x \mapsto v_1]t_2 \qquad \text{(E-LETV)}$$

$$\frac{t_1 \longrightarrow t_1'}{\texttt{let}\ x\ =\ t_1\ \texttt{in}\ t_2 \longrightarrow \texttt{let}\ x\ =\ t_1'\ \texttt{in}\ t_2} \qquad \text{(E-LET)}$$

Extension: Let Bindings (2)

Updated typing rules:

$$\Gamma \vdash \texttt{true} : \texttt{Bool} \qquad \text{(T-TRUE)}$$

$$\Gamma \vdash \texttt{false} : \texttt{Bool} \qquad \text{(T-FALSE)}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Bool} \quad \Gamma \vdash t_2 : T \quad \Gamma \vdash t_3 : T}{\Gamma \vdash \texttt{if}\ t_1\ \texttt{then}\ t_2\ \texttt{else}\ t_3 : T} \qquad \text{(T-IF)}$$

Extension: Let Bindings (3)

Updated typing rules:

$$\Gamma \vdash \texttt{0} : \texttt{Nat} \qquad \text{(T-ZERO)}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Nat}}{\Gamma \vdash \texttt{succ}\ t_1 : \texttt{Nat}} \qquad \text{(T-SUCC)}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Nat}}{\Gamma \vdash \texttt{pred}\ t_1 : \texttt{Nat}} \qquad \text{(T-PRED)}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Nat}}{\Gamma \vdash \texttt{iszero}\ t_1 : \texttt{Bool}} \qquad \text{(T-ISZERO)}$$

Extension: Let Bindings (4)

New typing rules:

$$\frac{x : T \in \Gamma}{\Gamma \vdash x : T} \quad \text{(T-VAR)}$$

$$\frac{\Gamma \vdash t_1 : T_1 \quad \Gamma, x : T_1 \vdash t_2 : T_2}{\Gamma \vdash \texttt{let}\ x\ =\ t_1\ \texttt{in}\ t_2 : T_2} \quad \text{(T-LET)}$$

Extension: Functions (1)

Syntactic extension:

t	$\rightarrow$	$\ldots$	*terms:*
	$\mid$	$\lambda x{:}T\,.\,t$	*abstraction*
	$\mid$	$t\ t$	*application*
v	$\rightarrow$	$\ldots$	*values:*
	$\mid$	$\lambda x{:}T\,.\,t$	*abstraction value*
T	$\rightarrow$	$\ldots$	*types:*
	$\mid$	$T \rightarrow T$	*type of functions*

Extension: Functions (2)

New evaluation rules:

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2} \quad \text{(E-APP1)}$$

$$\frac{t_2 \longrightarrow t_2'}{v_1\ t_2 \longrightarrow v_1\ t_2'} \quad \text{(E-APP2)}$$

$$(\lambda x{:}T_{11}\,.\,t_{12})v_2 \longrightarrow [x \mapsto v_2]t_{12} \quad \text{(E-APPABS)}$$

Note:

- left to right evaluation order: first the function (E-APP1), then the argument (E-APP2)
- ***call-by-value***: the argument fully evaluated before function “invoked” (E-APPABS).

Extension: Functions (3)

New typing rules:

$$\frac{\Gamma, x : T_1 \vdash t_2 : T_2}{\Gamma \vdash \lambda x{:}T_1\,.\,t_2 : T_1 \rightarrow T_2} \quad \text{(T-ABS)}$$

$$\frac{\Gamma \vdash t_1 : T_{11} \rightarrow T_{12} \quad \Gamma \vdash t_2 : T_{11}}{\Gamma \vdash t_1\ t_2 : T_{12}} \quad \text{(T-APP)}$$