

λ -calcolo tipato semplice

Nel λ -calcolo tutti i dati sono rappresentati tramite funzioni. Per evitare di svolgere operazioni tra valori che concettualmente rappresentano tipi di dato differenti, introduciamo un sistema dei tipi (in particolare una variante con primitive booleane).

Sintassi

$$\begin{array}{ll} e & ::= \text{fun } x : \tau = e \\ & \quad | \text{apply}(e, e) \\ & \quad | \text{true} \\ & \quad | \text{false} \\ & \quad | \text{if } e \text{ then } e \text{ else } e \\ \tau & ::= \text{Bool} \\ & \quad | \tau \rightarrow \tau \end{array}$$

Ambiente

Funzione parziale che associa tipi a nomi.

$$\Gamma, x : \tau(y) = \begin{cases} \tau & x = y \\ \Gamma(y) & \text{altrimenti} \end{cases}$$

Equivalente alla symbol table dei compilatori.

Controllo dei tipi

$$\begin{array}{c} \Gamma \vdash \text{true} : \text{Bool} \quad \Gamma \vdash \text{false} : \text{Bool} \quad \frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \\ \frac{\Gamma \vdash c : \text{Bool}, \Gamma \vdash t : \tau, \Gamma \vdash e : \tau}{\Gamma \vdash \text{if } c \text{ then } t \text{ else } e : \tau} \quad \frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \text{fun } x : \tau_1 = e : \tau_1 \rightarrow \tau_2} \end{array}$$

Type safety

Si dimostrano i lemmi di inversione e di forme canoniche e il teorema del progresso. Per dimostrare la conservazione, è necessario mostrare che il tipo di $\text{apply}(\text{fun } x : \tau = e_1, e_2)$ è lo stesso di $e_1[x := e_2]$. La sostituzione nel caso generale non produce un'espressione più “piccola” (albero sintattico di profondità minore), quindi non possiamo utilizzare l'induzione strutturale. Abbiamo bisogno del seguente lemma:

Substitution lemma

$$\Gamma, x : \tau \vdash e_1 : \tau' \wedge \Gamma \vdash e_2 : \tau \implies \Gamma \vdash e_1[x := e_2] : \tau'$$