Semânticas de Linguagens de Programação - CC3033

Exercícios 7

Linguagem funcional SFUN

1. Escreve um programa em **SFUN** que define funções para calcular o mínimo e o máximo valor entre um par de inteiros.

2. Escreve um programa em **SFUN** que define uma função para calcular n-ésimo número de Fibonacci.

3. Mostra que o programa **SFUN**

$$\begin{aligned} square(x) &= x \times x \\ double(x) &= 2 \times x \end{aligned}$$

   e o termo fechado $square(double(3))$ admitem tipo no ambiente

$$\begin{aligned} \varepsilon(square) &= \mathsf{int} \to \mathsf{int} \\ \varepsilon(double) &= \mathsf{int} \to \mathsf{int} \end{aligned}$$

   Indica qual o contexto (ambiente de variáveis) que é necessário para o programa.

4. Mostra que o programa **SFUN**

$$g(x) = \texttt{if}\ x = 0\ \texttt{then}\ 0\ \texttt{else}\ \texttt{if}\ x = 1\ \texttt{then}\ 1\ \texttt{else}\ g(x-1) + g(x-2)$$

   e o termo fechado $g(3)$ admitem tipo no ambiente $\varepsilon(g) = \mathsf{int} \to \mathsf{int}$. Indica qual o contexto (ambiente de variáveis) que é necessário para o programa.

5. Prova a unicidade de tipo, i.e. Para qualquer termo t de **SFUN**, contexto Γ e ambiente de símbolos funcionais ε, se $\Gamma \vdash_\varepsilon t : \tau$ e $\Gamma \vdash_\varepsilon t : \sigma$ então $\tau = \sigma$.

6. Para o factorial definido como

$$fact(x) = \texttt{if}\ x = 0\ \texttt{then}\ 1\ \texttt{else}\ x \times fact(x-1)$$

   avalia $fact(0)$ para a semântica $\Downarrow^v_P$.

7. Determina a avaliação do termo $square(double(3))$ com respeito ao programa

$$\begin{aligned} square(x) &= x \times x \\ double(x) &= 2 \times x \end{aligned}$$

   (a) para a semântica operacional com passagem de argumentos por valor
   (b) para a semântica operacional com passagem de argumentos por nome

8. Dá exemplo de um programa em **SFUN** e de um termo fechado cuja avaliação por nome e por valor seja diferente.

9. (Lema da Substituição) Mostra que se $x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash_\varepsilon t : \tau$ e $t_1, \ldots, t_n$ são termos fechados tal que $\vdash_\varepsilon t_i : \sigma_i$ para $1 \leq i \leq n$, então $\vdash_\varepsilon t[x_i \mapsto t_i] : \tau$. Nota: começa por definir formalmente a substituição $t[x \mapsto t']$ por indução no termo t.

10. (Preservação do Tipo para $\Downarrow^v_P$) Mostra que se t é um termo fechado com tipo σ e $t \Downarrow^v_P v$ então v tem tipo σ.

11. (Determinismo e Tipificação forte para $\Downarrow_P^n$) Mostrar que a avaliação de um termo t fechado e que admite tipo para um programa P não pode produzir erros. E se $t \Downarrow_P^n v_1$ e $t \Downarrow_P^n v_2$ então $v_1 = v_2$.

12. (Preservação do Tipo para $\Downarrow_P^n$) Mostra que se t é um termo fechado com tipo σ e $t \Downarrow_P^n v$ então v tem tipo σ.

13. Determina a avaliação do termo $g(3)$ com respeito ao programa

$$g(x) = \texttt{if } x = 0 \texttt{ then } 0 \texttt{ else if } x = 1 \texttt{ then } 1 \texttt{ else } g(x-1) + g(x-2)$$

    (a) para a semântica operacional com passagem de argumentos por valor
    (b) para a semântica operacional com passagem de argumentos por nome

Regras do sistema de tipos para SFUN

$$\frac{}{\Gamma \vdash_\varepsilon b : \mathsf{bool}}\ (\mathrm{b}) \qquad \frac{\Gamma \vdash_\varepsilon t_1 : \mathsf{bool} \qquad \Gamma \vdash_\varepsilon t_2 : \mathsf{bool}}{\Gamma \vdash_\varepsilon t_1 \wedge t_2 : \mathsf{bool}}\ (\mathrm{and})$$

$$\frac{}{\Gamma \vdash_\varepsilon n : \mathsf{int}}\ (\mathrm{n}) \qquad \frac{\Gamma \vdash_\varepsilon t : \mathsf{bool}}{\Gamma \vdash_\varepsilon \neg\, t : \mathsf{bool}}\ (\mathrm{not})$$

$$\frac{\Gamma(x) = \sigma}{\Gamma \vdash_\varepsilon x : \sigma}\ (\mathrm{var})$$

$$\frac{\Gamma \vdash_\varepsilon t_0 : \mathsf{bool} \qquad \Gamma \vdash_\varepsilon t_1 : \sigma \qquad \Gamma \vdash_\varepsilon t_2 : \sigma}{\Gamma \vdash_\varepsilon \texttt{if}\ t_0\ \texttt{then}\ t_1\ \texttt{else}\ t_2 : \sigma}\ (\mathrm{if})$$

$$\frac{\Gamma \vdash_\varepsilon t_1 : \mathsf{int} \qquad \Gamma \vdash_\varepsilon t_2 : \mathsf{int}}{\Gamma \vdash_\varepsilon t_1\ op\ t_2 : \mathsf{int}}\ (\mathrm{op})$$

$$\frac{\Gamma \vdash_\varepsilon t_1 : \sigma_1 \ \cdots \qquad \Gamma \vdash_\varepsilon t_{a_i} : \sigma_{a_i} \qquad \varepsilon(f_i) = (\sigma_1, \ldots, \sigma_{a_i}) \to \sigma}{\Gamma \vdash_\varepsilon f_i(t_1, \ldots, t_{a_i}) : \sigma}\ (\mathrm{fn})$$

$$\frac{\Gamma \vdash_\varepsilon t_1 : \mathsf{int} \qquad \Gamma \vdash_\varepsilon t_2 : \mathsf{int}}{\Gamma \vdash_\varepsilon t_1\ bop\ t_2 : \mathsf{bool}}\ (\mathrm{bop})$$

Semânticas com passagem de argumentos por valor e por nome para SFUN dado um programa P, $f_i(x_1, \ldots, x_{a_i}) = d_i$, $1 \leq i \leq k$.

$$\frac{}{n \Downarrow_P^v n}\ (\mathrm{n}) \qquad \frac{t \Downarrow_P^v b}{\neg\, t \Downarrow_P^v \neg\, b}\ (\mathrm{not})$$

$$\frac{}{b \Downarrow_P^v b}\ (\mathrm{b}) \qquad \frac{t_0 \Downarrow_P^v \mathsf{true} \qquad t_1 \Downarrow_P^v v_1}{\texttt{if}\ t_0\ \texttt{then}\ t_1\ \texttt{else}\ t_2 \Downarrow_P^v v_1}\ (\mathrm{ifT})$$

$$\frac{t_1 \Downarrow_P^v n_1 \qquad t_2 \Downarrow_P^v n_2}{t_1\ op\ t_2 \Downarrow_P^v n_1\ op\ n_2}\ (\mathrm{op}) \qquad \frac{t_0 \Downarrow_P^v \mathsf{false} \qquad t_2 \Downarrow_P^v v_2}{\texttt{if}\ t_0\ \texttt{then}\ t_1\ \texttt{else}\ t_2 \Downarrow_P^v v_2}\ (\mathrm{ifF})$$

$$\frac{t_1 \Downarrow_P^v n_1 \qquad t_2 \Downarrow_P^v n_2}{t_1\ bop\ t_2 \Downarrow_P^v n_1\ bop\ n_2}\ (\mathrm{bop}) \qquad \frac{t_1 \Downarrow_P^v v_1 \quad \ldots t_{a_i} \Downarrow_P^v v_{a_i} \qquad d_i[x_1 \mapsto v_1, \ldots x_{a_i} \mapsto v_{a_i}] \Downarrow_P^v v}{f_i(t_1, \ldots, t_{a_i}) \Downarrow_P^v v}\ (\mathrm{fn})$$

$$\frac{t_1 \Downarrow_P^v b_1 \qquad t_2 \Downarrow_P^v b_2}{t_1 \wedge t_2 \Downarrow_P^v b_1 \wedge b_2}\ (\mathrm{and}) \qquad \frac{d_i[x_1 \mapsto t_1, \ldots x_{a_i} \mapsto t_{a_i}] \Downarrow_P^n v}{f_i(t_1, \ldots, t_{a_i}) \Downarrow_P^n v}\ (fn_N)$$