Lista de soluções 4

Exercício 1 (Extensões)

1. Uma abordagem é de definir a semântica operacional natural do laço `repeat...until` usando o laço `while...do`

$$\frac{\mathbf{c}, \sigma \Downarrow \sigma' \quad \texttt{while}\ \neg\mathbf{b}\ \texttt{do}\ \mathbf{c}, \sigma' \Downarrow \sigma''}{\texttt{repeat}\ \mathbf{c}\ \texttt{until}\ \mathbf{b}, \sigma \Downarrow \sigma''}\ \text{repeat}$$

   Uma outra solução é de definir a semântica diretamente

$$\frac{\mathbf{c}, \sigma \Downarrow \sigma' \quad \mathbf{b}, \sigma' \Downarrow \text{true}}{\texttt{repeat}\ \mathbf{c}\ \texttt{until}\ \mathbf{b}, \sigma \Downarrow \sigma'}\ \text{repeat}$$

$$\frac{\mathbf{c}, \sigma \Downarrow \sigma' \quad \mathbf{b}, \sigma' \Downarrow \text{false} \quad \texttt{repeat}\ \mathbf{c}\ \texttt{until}\ \mathbf{b}, \sigma' \Downarrow \sigma''}{\texttt{repeat}\ \mathbf{c}\ \texttt{until}\ \mathbf{b}, \sigma \Downarrow \sigma''}\ \text{repeat}$$

2. Uma solução (neste caso usando a semântica operacional estrutural) é

$$\frac{\mathbf{c_1}, \sigma \to \mathbf{c_1'}, \sigma'}{\texttt{try}\ \mathbf{c_1}\ \texttt{catch}\ \mathbf{c_2}, \sigma \to \texttt{try}\ \mathbf{c_1}\ \texttt{catch}\ \mathbf{c_2}, \sigma'}\ \text{tct}$$

$$\frac{}{\texttt{try skip catch}\ \mathbf{c_2}, \sigma \to \texttt{skip}, \sigma}\ \text{tct}$$

$$\frac{}{\texttt{throw;}\mathbf{c}, \sigma \to \texttt{throw}, \sigma}\ \text{tct}$$

$$\frac{}{\texttt{try throw catch}\ \mathbf{c_2}, \sigma \to \mathbf{c_2}, \sigma}\ \text{tct}$$

   A idéa básica é que uma execução normal resulta em `skip` que a segunda regra usa para "eliminar" `try-catch`. No caso de uma exeção, `throw` "elimina" todas comandos que seguem e depois, chegando em `try-catch`, a parte `catch` é executada.

Exercício 2 (Interpredador de IMP: SOE)
Veja na página.

Exercício 3 (Substituições)
(a) $(f\,(\lambda x.x\,y)\,(\lambda z.x\,y\,z))[g/x] = f\,(\lambda x.x\,y)\,(\lambda z.g\,y\,z)$

(b) $(\lambda x.\lambda y.f\,x\,y)[x/y] = \lambda x.\lambda y.f\,x\,y$

(c) $((\lambda x.f\,x)(\lambda f.f\,x))[g\,x/f] = (\lambda x.(g\,x)\,x)(\lambda f.f\,x)$

(d) $(\lambda f.\lambda y.f\,x\,y)[f\,y/x] = (\lambda g.\lambda z.g\,(f\,y)\,z)$

Exercício 4 (Avaliação de expressões do cálculo lambda)
1. Avaliação de expressões

   (a)

$$\underline{\text{and true}}\,\text{true} \to_{app_1,\beta} \underline{(\lambda v'.\text{true}\,v'\,\text{false})\text{true}} \to_\beta \underline{\text{true true}}\,\text{false} \to_{app_1,\beta} \underline{(\lambda f.\text{true})\,\text{false}} \to_\beta \text{true}$$

(b)

$$\underline{\text{and false}}\,\text{true} \to_{app_1,\beta} \underline{(\lambda v'.\text{false}\,v'\,\text{false})\,\text{true}} \to_\beta \underline{\text{false true}}\,\text{false} \to_{app_1,\beta} \underline{(\lambda f.f)\,\text{false}} \to_\beta \text{false}$$

(c)

$$\underline{\text{or true}}\,\text{true} \to_{app_1,\beta} \underline{(\lambda v'.\text{true true}\,v')\,\text{true}} \to_\beta \underline{\text{true true}}\,\text{true} \to_{app_1,\beta} \underline{(\lambda f.\text{true})\,\text{true}} \to_\beta \text{true}$$

(d)

$$\underline{\text{or false}}\,\text{false} \to_{app_1,\beta} \underline{(\lambda v'.\text{false true}\,v')\,\text{false}} \to_\beta \underline{\text{false true}}\,\text{false} \to_{app_1,\beta} \underline{(\lambda f.f)\,\text{false}} \to_\beta \text{false}$$

(e)

$$\begin{aligned}\underline{\text{plus}\,2}\,2 &\to_{app_1,\beta} \underline{\lambda n.\lambda s.\lambda z.2\,s(n\,s\,z)} \to_\beta \lambda s.\lambda z.\underline{2\,s}(2\,s\,z) \to_{abs,abs,\beta} \lambda s.\lambda z.\underline{(\lambda z'.s\,(s\,z'))\,(\underline{2\,s}\,z)}\\ &\to_{abs,abs,\beta} \lambda s.\lambda z.s\,(s\,(2\,s\,z)) \to_{abs,abs,app_2,app_2,\beta} \lambda s.\lambda z.s\,(s\,\underline{(\lambda z'.s\,(s\,z'))\,z})\\ &\to_{abs,abs,app_2,app_2,\beta} \lambda s.\lambda z.s\,(s\,(s\,(s\,z))) \equiv 4\end{aligned}$$

(f)

$$\begin{aligned}\underline{\text{times}\,2}\,2 &\to_{app_1,\beta} \underline{(\lambda n.2\,(\text{plus}\,n)\,0)\,2}\\ &\to_\beta \underline{2\,(\text{plus}\,2)}\,0 \to_{app_1,\beta} \underline{(\lambda z.(\text{plus}\,2)((\text{plus}\,2)z))\,0} \to_\beta (\text{plus}\,2)(\underline{(\text{plus}\,2)\,0})\\ &\to^* \underline{(\text{plus}\,2)\,2} \to^* 4\end{aligned}$$

2. Seja $\text{not} \equiv \lambda b.\lambda t.\lambda f.b\,f\,t$. Logo

$$\underline{\text{not true}} \to_\beta \lambda t.\lambda f.\underline{\text{true}\,f}\,t \to_{abs,abs,app1,\beta} \lambda t.\lambda f.\underline{(\lambda f'.f)\,t} \to_{abs,abs,\beta} \lambda t.\lambda f.f \equiv \text{false}$$

e

$$\underline{\text{not false}} \to_b eta\lambda t.\lambda f.\underline{\text{false}\,f}\,t \to_{abs,abs,app1,\beta} \lambda t.\lambda f.\underline{(\lambda f'.f')\,t} \to_{abs,abs,\beta} \lambda t.\lambda f.t \equiv \text{true}$$

3. Seja $\text{exp} \equiv \lambda b.\lambda e.e\,(\text{times}\,b)\,1$. Logo

$$\begin{aligned}\underline{\text{exp}\,2}\,3 &\to_{app1,\beta} \underline{(\lambda e.e\,(\text{times}\,2)\,1)\,3} \to_\beta \underline{3\,(\text{times}\,2)}\,1\\ &\to_{app1,\beta} \underline{\lambda z.((\text{times}\,2)((\text{times}\,2)(\text{times}\,2\,z')))\,1}\\ &\to_\beta (\text{times}\,2)((\text{times}\,2)(\underline{\text{times}\,2\,1})) \to^* (\text{times}\,2)\underline{((\text{times}\,2)\,2)}\\ &\to^* (\underline{\text{times}\,2}\,4) \to^* 8\end{aligned}$$