

Semânticas de Linguagens de Programação - Exercícios 2
Semântica das expressões e Semântica operacional *big-step* para a linguagem
While

Semântica das expressões

1. Sendo $s(x) = 2$ calcula
 - (a) $\mathcal{A}[\![x + 2]\!] s$
 - (b) $\mathcal{A}[\![x * y]\!] s$
 - (c) $\mathcal{B}[\![\neg(x < 1)]\!] s$
2. Sejam s e s' tal que $s(x) = s'(x)$ para todo $x \in FV(a)$. Mostra que $\mathcal{A}[\![a]\!] s = \mathcal{A}[\![a]\!] s'$. Análogo para $b \in \mathbf{Bexp}$.
3. Mostra que $\mathcal{A}[\![a[y \mapsto a_0]]\!] s = \mathcal{A}[\![a]\!] s[y \mapsto \mathcal{A}[\![a_0]\!] s]$.
4. Define a substituição para expressões booleanas $b[x \mapsto a_0]$ e mostra a proposição equivalente ao do exercício anterior.

Semântica operacional para Expressões (aritméticas e booleanas)

Seja $\mathbf{Mem} = [\mathbf{Var} \rightarrow \mathbb{Z}]$ e $s \in \mathbf{Mem}$. Para cada $a \in \mathbf{Aexp}$, $\mathcal{A}[[a]] : \mathbf{Mem} \hookrightarrow \mathbb{Z}$:

$$\begin{aligned}
 \mathcal{A}[[n]] s &= \mathcal{N}[[n]] \\
 \mathcal{A}[[x]] s &= \begin{cases} s(x) & \text{se } x \in \text{dom}(s) \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{A}[[a_1 + a_2]] s &= \begin{cases} z_1 + z_2 & \text{se } z_1 = \mathcal{A}[[a_1]] s \text{ e } z_2 = \mathcal{A}[[a_2]] s \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{A}[[a_1 - a_2]] s &= \begin{cases} z_1 - z_2 & \text{se } z_1 = \mathcal{A}[[a_1]] s \text{ e } z_2 = \mathcal{A}[[a_2]] s \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{A}[[a_1 * a_2]] s &= \begin{cases} z_1 \cdot z_2 & \text{se } z_1 = \mathcal{A}[[a_1]] s \text{ e } z_2 = \mathcal{A}[[a_2]] s \\ \text{indefinido} & \text{caso contrário} \end{cases}
 \end{aligned}$$

$\mathbb{T} = \{\text{true}, \text{false}\}$ e $\mathcal{B} : \mathbf{Bexp} \rightarrow (\mathbf{Mem} \hookrightarrow \mathbb{T})$

$$\begin{aligned}
 \mathcal{B}[[\text{true}]] s &= \text{true} \\
 \mathcal{B}[[a_1 = a_2]] s &= \begin{cases} \text{true} & \text{se } z_1 = \mathcal{A}[[a_1]] s, z_2 = \mathcal{A}[[a_2]] s \text{ e } z_1 = z_2 \\ \text{false} & \text{se } z_1 = \mathcal{A}[[a_1]] s, z_2 = \mathcal{A}[[a_2]] s \text{ e } z_1 \neq z_2 \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{B}[[a_1 \leq a_2]] s &= \begin{cases} \text{true} & \text{se } z_1 = \mathcal{A}[[a_1]] s, z_2 = \mathcal{A}[[a_2]] s \text{ e } z_1 \leq z_2 \\ \text{false} & \text{se } z_1 = \mathcal{A}[[a_1]] s, z_2 = \mathcal{A}[[a_2]] s \text{ e } z_1 > z_2 \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{B}[[\neg b]] s &= \begin{cases} \text{true} & \text{se } \mathcal{B}[[b]] s = \text{false} \\ \text{false} & \text{se } \mathcal{B}[[b]] s = \text{true} \\ \text{indefinido} & \text{caso contrário} \end{cases} \\
 \mathcal{B}[[b_1 \wedge b_2]] s &= \begin{cases} \text{true} & \text{se } \mathcal{B}[[b_1]] s = \text{true} \text{ e } \mathcal{B}[[b_2]] s = \text{true} \\ \text{indefinido} & \text{se } \mathcal{B}[[b_1]] s \text{ ou } \mathcal{B}[[b_2]] s \text{ são indefinidos} \\ \text{false} & \text{caso contrário} . \end{cases}
 \end{aligned}$$