Trabajo Práctico 1

Análisis de lenguajes de programación

LCC

Erik Gimenez
Manuel Spreutels
Augusto Rabbia

Septiembre, 2023

1 Soluciones

1.1 Ejercicio 1

Se agrega a la categoría sintáctica *intexp* las siguiente reglas:

$$\underbrace{\begin{array}{rl} intexp \ ::= & var = intexp \\ & \mid intexp \ , \ intexp \end{array}}_{\text{Sintaxis abstracta}} \qquad \underbrace{\begin{array}{rl} intexp \ ::= & var \text{ '=' } intexp \\ & \mid intexp \text{ ',' } intexp \end{array}}_{\text{Sintaxis concreta}}$$

1.2 Ejercicio 2

Se agregaron en el archivo AST.hs los siguientes constructores:

- EAssgn :: Variable $\rightarrow$ Exp Int $\rightarrow$ Exp Int
- ESeq :: Exp Int $\rightarrow$ Exp Int $\rightarrow$ Exp Int

1.3 Ejercicio 3

En el archivo Parser.hs

1.4 Ejercicio 4

Se agregan a la semántica operacional Big-Step para expresiones las siguientes reglas:

$$\frac{\langle e, \sigma \rangle \Downarrow_{exp} \langle n, \sigma' \rangle}{\langle x = e, \sigma \rangle \Downarrow_{exp} \langle n, [\sigma' | x : n] \rangle} \text{ EASSIGN}$$

$$\frac{\langle e_1, \sigma \rangle \Downarrow_{exp} \langle n_1, \sigma' \rangle \qquad \langle e_2, \sigma' \rangle \Downarrow_{exp} \langle n_2, \sigma'' \rangle}{\langle e_1; e_2, \sigma \rangle \Downarrow_{exp} \langle n_2, \sigma'' \rangle} \text{ ESEQ}$$

1.5 Ejercicio 5

Suponemos $\alpha \rightsquigarrow \beta$ y $\alpha \rightsquigarrow \gamma$. Hacemos inducción sobre $\alpha \rightsquigarrow \beta$. Supongamos que la última regla utilizada fue:

- **ASS**.
  Luego, sabemos:

  a) $\langle e, \sigma \rangle \Downarrow \langle n, \sigma' \rangle$

b) $\alpha = \langle v = e, \sigma\rangle$

c) $\beta = \langle \textbf{skip}, [\sigma'|v : n]\rangle$

Analizamos $\alpha \rightsquigarrow \gamma$:

La regla aplicada debe ser **ASS** por la forma de α. Es decir, $\alpha \rightsquigarrow \gamma$ tiene la forma

$$\frac{\langle e, \sigma\rangle \Downarrow_{exp} \langle n', \sigma''\rangle}{\langle v = e, \sigma\rangle \rightsquigarrow \langle \textbf{skip}, [\sigma''|v : n]\rangle}\ \text{ASS}$$

Luego, como $\Downarrow$ es determinista, se tiene $\langle n, \sigma'\rangle = \langle n', \sigma''\rangle$ y por lo tanto,
$\gamma = \langle \textbf{skip}, [\sigma''|v : n']\rangle = \langle \textbf{skip}, [\sigma'|v : n]\rangle = \beta$

- **SEQ1**.
  Sabemos:

  a) $\alpha = \langle \textbf{skip}; c_1, \sigma\rangle$

  b) $\beta = \langle c_1, \sigma\rangle$

  Analizamos $\alpha \rightsquigarrow \gamma$: la regla aplicada debe ser **SEQ1** puesto que **skip** es el primer comando de la secuencia en α. Es decir, $\alpha \rightsquigarrow \gamma$ tiene la forma

$$\frac{}{\langle \textbf{skip}; c_1, \sigma\rangle \rightsquigarrow \langle c_1, \sigma\rangle}\ \text{SEQ1}$$

  Luego, $\gamma = \langle c_1, \sigma\rangle = \beta$.

- **IF1**.
  Se tiene:

  a) $\langle b, \sigma\rangle \Downarrow \langle \textbf{true}, \sigma'\rangle$

  b) $\alpha = \langle \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1, \sigma\rangle$

  c) $\beta = \langle c_0, \sigma'\rangle$

  Analizamos $\alpha \rightsquigarrow \gamma$: observemos que por la forma de α, solo pueden aplicarse las reglas **IF1** e **IF2** para derivar $\alpha \rightsquigarrow \gamma$. Supongamos que se aplicó **IF2**:

$$\frac{\langle b, \sigma\rangle \Downarrow \langle \textbf{false}, \sigma''\rangle}{\langle \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1, \sigma\rangle \rightsquigarrow \langle c_1, \sigma''\rangle}\ \text{IF2}$$

  Se deduce que $\langle b, \sigma\rangle \Downarrow \langle \textbf{false}, \sigma''\rangle$, pero $\Downarrow$ es determinista $\overset{\textbf{a)}}{\Longrightarrow}$ absurdo. Luego, debió aplicarse **IF1** para $\alpha \rightsquigarrow \gamma$:

$$\frac{\langle b, \sigma\rangle \Downarrow \langle \mathbf{true}, \sigma''\rangle}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma\rangle \rightsquigarrow \langle c_0, \sigma''\rangle}\ \text{IF1}$$

Pero como $\Downarrow$ es determinista, asi que $\sigma'' = \sigma'$, y por tanto $\gamma = \langle c_0, \sigma''\rangle = \langle c_0, \sigma'\rangle = \beta$

- **IF2**.
  Análogo al caso **IF1**
- **REPEAT**.
  Se tiene:
  a) $\alpha = \langle \mathbf{repeat}\ c\ \mathbf{until}\ b, \sigma\rangle$
  b) $\beta = \langle c;\ \mathbf{if}\ b\ \mathbf{then\ skip\ else\ repeat}\ c\ \mathbf{until}\ b, \sigma\rangle$

  Analizamos $\alpha \rightsquigarrow \gamma$: la regla aplicada debe ser **REPEAT** por la forma de α. Es decir, $\alpha \rightsquigarrow \gamma$ tiene la forma

$$\frac{}{\langle \mathbf{repeat}\ c\ \mathbf{until}\ b, \sigma\rangle \rightsquigarrow \langle c;\ \mathbf{if}\ b\ \mathbf{then\ skip\ else\ repeat}\ c\ \mathbf{until}\ b, \sigma\rangle}\ \text{REPEAT}$$

  Luego, $\gamma = \langle c;\ \mathbf{if}\ b\ \mathbf{then\ skip\ else\ repeat}\ c\ \mathbf{until}\ b, \sigma\rangle = \beta$.
- **SEQ2**.
  Sabemos entonces:
  a) $\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0', \sigma'\rangle$
  b) $\alpha = \langle c_0; c_1, \sigma\rangle$
  c) $\beta = \langle c_0'; c_1, \sigma'\rangle$

  HI: Suponemos que para toda subderivación de $\alpha \rightsquigarrow \beta$, de la forma $\alpha' \rightsquigarrow \beta'$, si $\alpha' \rightsquigarrow \beta'$ y $\alpha' \rightsquigarrow \beta''$, entonces $\beta' = \beta''$.
  Por la forma de α, conocemos la forma de $\alpha \rightsquigarrow \gamma$, pues sólo **SEQ2** pudo aplicarse:

$$\frac{\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0'', \sigma''\rangle}{\langle c_0; c_1, \sigma\rangle \rightsquigarrow \langle c_0''; c_1, \sigma''\rangle}\ \text{SEQ2}$$

  Luego, $\gamma = \langle c_0''; c_1, \sigma''\rangle \overset{\textbf{(HI)}}{=} \langle c_0'; c_1, \sigma'\rangle = \beta$.

1.6 Ejercicio 6

En el archivo Eval1.hs

1.7 Ejercicio 7

En el archivo Eval2.hs

1.8 Ejercicio 8

En el archivo Eval3.hs

1.9 Ejercicio 9

Se agregan las siguientes reglas de evaluación de paso chico:

$$\frac{\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0', \sigma'\rangle}{\langle \textbf{catch } c_0 \textbf{ with } c1, \sigma\rangle \rightsquigarrow \langle \textbf{catch } c_0' \textbf{ with } c1, \sigma\rangle} \text{ CATCH1}$$

$$\frac{}{\langle \textbf{catch skip with } c1, \sigma\rangle \rightsquigarrow \langle \textbf{skip}, \sigma\rangle} \text{ CATCHSKIP}$$

$$\frac{}{\langle \textbf{catch } err_c \textbf{ with } c1, \sigma\rangle \rightsquigarrow \langle c1, \sigma\rangle} \text{ CATCHERR}$$

Por otro lado, se añade a la sintaxis concreta la siguiente regla de producción:

$$comm \ ::= \text{'catch'}\ comm\ \text{'with'}\ comm$$