Análisis de Lenguajes de Programación TP1

Belmonte Marina

Ejercicio 1

Sintaxis Abstracta

$$
\begin{aligned}
intexp \ &::= nat \mid var \mid \ -_u \ intexp \\
&\mid \ intexp \ + \ intexp \\
&\mid \ intexp \ -_b \ intexp \\
&\mid \ intexp \ \times \ intexp \\
&\mid \ intexp \ \div \ intexp \\
&\mid \ boolexp \ ? \ intexp \ : \ intexp \\
boolexp \ &::= \mathbf{true} \mid \mathbf{false} \\
&\mid \ intexp \ == \ intexp \\
&\mid \ intexp \ \neq \ intexp \\
&\mid \ intexp \ < \ intexp \\
&\mid \ intexp \ > \ intexp \\
&\mid \ boolexp \ \wedge \ boolexp \\
&\mid \ boolexp \ \vee \ boolexp \\
&\mid \ \neg \ boolexp \\
comm \ &:= \mathbf{skip} \\
&\mid \ var = intexp \\
&\mid \ comm; \ comm \\
&\mid \ \mathbf{if} \ boolexp \ \mathbf{then} \ comm \ \mathbf{else} \ comm \\
&\mid \ \mathbf{while} \ boolexp \ \mathbf{do} \ comm
\end{aligned}
$$

Sintaxis Concreta

$$
\begin{array}{rcl}
\mathit{digit} & ::= & \textbf{'0'} \mid \textbf{'1'} \mid \ldots \mid \textbf{'9'} \\
\mathit{letter} & ::= & \textbf{'a'} \mid \ldots \mid \textbf{'Z'} \\
\mathit{nat} & ::= & \mathit{digit} \mid \mathit{digit}\ \mathit{nat} \\
\mathit{var} & ::= & \mathit{letter} \mid \mathit{letter}\ \mathit{var} \\
\mathit{intexp} & ::= & \mathit{nat} \\
& \mid & \mathit{var} \\
& \mid & \textbf{'-'}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'+'}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'-'}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'*'}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'/'}\ \mathit{intexp} \\
& \mid & \textbf{'('}\ \mathit{intexp}\ \textbf{')'} \\
& \mid & \mathit{boolexp}\ \textbf{'?'}\ \mathit{intexp}\ \textbf{':'}\ \mathit{intexp} \\
\mathit{boolexp} & ::= & \textbf{'true'} \mid \textbf{'false'} \\
& \mid & \mathit{intexp}\ \textbf{'=='}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'!='}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'<'}\ \mathit{intexp} \\
& \mid & \mathit{intexp}\ \textbf{'>'}\ \mathit{intexp} \\
& \mid & \mathit{boolexp}\ \textbf{'\&\&'}\ \mathit{boolexp} \\
& \mid & \mathit{boolexp}\ \textbf{'||'}\ \mathit{boolexp} \\
& \mid & \textbf{'!'}\ \mathit{boolexp} \\
& \mid & \textbf{'('}\ \mathit{boolexp}\ \textbf{')'} \\
\mathit{comm} & := & \boldsymbol{skip} \\
& \mid & \mathit{var}\ \textbf{'='}\ \mathit{intexp} \\
& \mid & \mathit{comm}\ \textbf{';'}\ \mathit{comm} \\
& \mid & \textbf{'if'}\ \mathit{boolexp}\ \textbf{'\{'}\ \mathit{comm}\ \textbf{'\}'} \\
& \mid & \textbf{'if'}\ \mathit{boolexp}\ \textbf{'\{'}\ \mathit{comm}\ \textbf{'\}'}\ \textbf{'else'}\ \textbf{'\{'}\ \mathit{comm}\ \textbf{'\}'} \\
& \mid & \textbf{'while'}\ \textbf{'\{'}\ \mathit{boolexp}\ \textbf{'\}'}
\end{array}
$$

Ejercicio 2

Se encuentra en el archivo `src/AST.hs`.

Ejercicio 3

Se encuentra en el archivo `src/Parser.hs`.

Ejercicio 4

$$
\frac{\langle p, \sigma\rangle \Downarrow_{exp} \textbf{true} \qquad \langle e_0, \sigma'\rangle \Downarrow_{exp} n_0}{\langle p\ ?\ e_0\ :\ e_1, \sigma\rangle \Downarrow_{exp} n_0}\ \text{ECond}_1
\qquad
\frac{\langle p, \sigma\rangle \Downarrow_{exp} \textbf{false} \qquad \langle e_1, \sigma'\rangle \Downarrow_{exp} n_1}{\langle p\ ?\ e_0\ :\ e_1, \sigma\rangle \Downarrow_{exp} n_1}\ \text{ECond}_2
$$

Ejercicio 5

Vamos a probar que la relación $\rightsquigarrow$ es determinista, es decir, si $t \rightsquigarrow t_1$ y $t \rightsquigarrow t_2$ entonces $t_1 = t_2$. Para probarlo usamos inducción sobre la derivación $t \rightsquigarrow t_1$.

- Consideremos que la última regla utilizada fue ASS, entonces:
  - $\langle e, \sigma \rangle \Downarrow_{exp} n$
  - $t = \langle v = e, \sigma \rangle$
  - $t_1 = \langle \mathbf{skip}, [\sigma' | v : n] \rangle$

  Como $\Downarrow_{exp}$ es determinista y la única regla que se le puede aplicar a t es ASS, entonces $t_1 = t_2$

- Consideremos que la última regla utilizada fue SEQ1, entonces:
  - $t = \langle \mathbf{skip}; c_1, \sigma \rangle$
  - $t_1 = \langle c_1, \sigma \rangle$

  Como la única regla que se le puede aplicar a t es SEQ1, entonces $t_1 = t_2$

- Consideremos que la ultima regla utilizada fue SEQ2, entonces:
  - $\langle c_0, \sigma \rangle \rightsquigarrow \langle c_0', \sigma' \rangle$
  - $t = \langle c_0; c_1, \sigma \rangle$
  - $t_1 = \langle c_0'; c_1, \sigma' \rangle$

  Como la única regla que se le puede aplicar a t es SEQ2 y por hipótesis inductiva la derivación $\langle c_0, \sigma \rangle \rightsquigarrow \langle c_0', \sigma' \rangle$ es determinista, no es posible aplicar SEQ2 con una hipótesis distinta. Por lo tanto $t_1 = t_2$

- Consideremos que la última regla utilizada fue IF1, entonces:
  - $\langle b, \sigma \rangle \Downarrow_{exp} \mathbf{true}$
  - $t = \langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma \rangle$
  - $t_1 = \langle c_0, \sigma \rangle$

  Como $\Downarrow_{exp}$ es determinista y la única regla que se le puede aplicar a t es IF1, entonces $t_1 = t_2$

- Consideremos que la última regla utilizada fue IF2, entonces:
  - $\langle b, \sigma \rangle \Downarrow_{exp} \mathbf{false}$
  - $t = \langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma \rangle$
  - $t_1 = \langle c_1, \sigma \rangle$

  Como $\Downarrow_{exp}$ es determinista y la única regla que se le puede aplicar a t es IF2, entonces $t_1 = t_2$

- Consideremos que la última regla utilizada fue WHILE1, entonces:
  - $\langle b, \sigma \rangle \Downarrow_{exp} \mathbf{true}$
  - $t = \langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma \rangle$
  - $t_1 = \langle c; \mathbf{while}\ b\ \mathbf{do}\ c, \sigma \rangle$

  Como $\Downarrow_{exp}$ es determinista y la única regla que se le puede aplicar a t es WHILE1, entonces $t_1 = t_2$

- Consideremos que la última regla utilizada fue WHILE2, entonces:
  - $\langle b, \sigma \rangle \Downarrow_{exp}$ **false**
  - $t = \langle \textbf{while } b \textbf{ do } c, \sigma \rangle$
  - $t_1 = \langle \textbf{skip}, \sigma \rangle$

  Como $\Downarrow_{exp}$ es determinista y la única regla que se le puede aplicar a t es WHILE2, entonces $t_1 = t_2$

Queda así demostrado que la relación de evaluación en un paso $\rightsquigarrow$ es determinista.

Ejercicio 6

Para la demostración utilizaré la clausura reflexivo-transitiva de $\rightsquigarrow$, donde:

$$\frac{t \rightsquigarrow t'}{t \rightsquigarrow^* t'}\ \mathrm{T}_1 \qquad\qquad \frac{t \rightsquigarrow^* t' \qquad t' \rightsquigarrow^* t''}{t \rightsquigarrow^* t''}\ \mathrm{T}_2$$

A:

$a = \textbf{while } x > 0 \textbf{ do} := x - y$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x, [[\sigma|x:1]|y:2]\rangle \Downarrow_{exp} 1}\ \text{VAR} \qquad \dfrac{}{\langle y, [[\sigma|x:1]|y:2]\rangle \Downarrow_{exp} 2}\ \text{VAR}}{\langle x > y, [[\sigma|x:1]|y:2]\rangle \Downarrow_{exp} 1 > 2}\ \text{GT} \qquad \dfrac{}{\langle y, [[\sigma|x:1]|y:2]\rangle \Downarrow_{exp} 2}\ \text{VAR}}{\langle x > y \ ?\ 2 * y : y, [[\sigma|x:1]|y:2]\rangle \Downarrow_{exp} 2}\ \text{ECOND}_2}{\langle x := x > y \ ?\ 2 * y : y, [[\sigma|x:1]|y:2]\rangle \rightsquigarrow \langle \textbf{skip}, [[\sigma|x:2]|y:2]\rangle}\ \text{ASS}}{\langle x := x > y \ ?\ 2 * y : y;\ a, [[\sigma|x:1]|y:2]\rangle \rightsquigarrow \langle \textbf{skip};\ a, [[\sigma|x:2]|y:2]\rangle}\ \text{SEQ}_2}{\langle x := x > y \ ?\ 2 * y : y;\ a, [[\sigma|x:1]|y:2]\rangle \rightsquigarrow^* \langle \textbf{skip}; \textbf{while } x > 0 \textbf{ do} := x - y, [[\sigma|x:2]|y:2]\rangle}\ \text{T}_1$$

B:

$$\dfrac{\dfrac{}{\langle \textbf{skip}; \textbf{while } x > 0 \textbf{ do} := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle \textbf{while } x > 0 \textbf{ do} := x - y, [[\sigma|x:2]|y:2]\rangle}\ \text{SEQ}_1}{\langle \textbf{skip}; \textbf{while } x > 0 \textbf{ do} := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \textbf{while } x > 0 \textbf{ do} := x - y, [[\sigma|x:2]|y:2]\rangle}\ \text{T}_1$$

C:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 2}\ \text{VAR} \qquad \dfrac{}{\langle 0, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 0}\ \text{NVAL}}{\langle x > 0, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 2 > 0}\ \text{GT}}{\langle \textbf{while } x > 0 \textbf{ do } x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle x := x - y; \textbf{while } x > 0 \textbf{ do } x := x - y, [[\sigma|x:2]|y:2]\rangle}\ \text{WHILE}_1}{\langle \textbf{while } x > 0 \textbf{ do } x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle x := x - y; \textbf{while } x > 0 \textbf{ do } x := x - y, [[\sigma|x:2]|y:2]\rangle}\ \text{T}_1$$

D:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 2}\,\text{VAR} \qquad \dfrac{}{\langle y, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 2}\,\text{VAR}}{\langle x - y, [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} 2 \text{ - } 2}\,\text{MINUS}}{\langle x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle \textbf{skip},\ [[\sigma|x:0]|y:2]\rangle}\,\text{ASS}}{\langle x := x - y; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle}\,\text{SEQ}_2}{\langle x := x - y; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle}\,\text{T}_1$$

E:

$$\dfrac{\dfrac{}{\langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle \rightsquigarrow \langle \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle}\,\text{SEQ}_1}{\langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle \rightsquigarrow^* \langle \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle}\,\text{T}_1$$

F:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x, [[\sigma|x:0]|y:2]\rangle \Downarrow_{exp} 0}\,\text{VAR} \qquad \dfrac{}{\langle 0, [[\sigma|x:0]|y:2]\rangle \Downarrow_{exp} 0}\,\text{NVAL}}{\langle x > 0, [[\sigma|x:0]|y:2]\rangle \Downarrow_{exp} 0 > 0}\,\text{GT}}{\langle \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle \rightsquigarrow \langle \textbf{skip},\ [[\sigma|x:0]|y:2]\rangle}\,\text{WHILE}_2}{\langle \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle \rightsquigarrow^* \langle \textbf{skip},\ [[\sigma|x:0]|y:2]\rangle}\,\text{T}_1$$

G:

$$\dfrac{A \qquad B}{\langle x := x > y\ ?\ 2\ *\ y : y;\ \ \textbf{while}\ x > 0\ \textbf{do} := x - y, [[\sigma|x:1]|y:2]\rangle \rightsquigarrow^* \langle \textbf{while}\ x > 0\ \textbf{do} := x - y, [[\sigma|x:2]|y:2]\rangle}\,\text{T}_2$$

H:

$$\dfrac{C \qquad D}{\langle \textbf{while}\ x > 0\ \textbf{do}\ x := x - y, [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle}\,\text{T}_2$$

I:

$$\dfrac{E \qquad F}{\langle \textbf{skip}; \textbf{while}\ x > 0\ \textbf{do}\ x := x - y,\ [[\sigma|x:0]|y:2]\rangle \rightsquigarrow^* \langle \textbf{skip},\ [[\sigma|x:0]|y:2]\rangle}\,\text{T}_2$$

J:

$a = \textbf{while}\ x > 0\ \textbf{do} := x - y$

$$\frac{G \quad H}{\langle x := x > y\ ?\ 2\ *\ y : y;\ a, [[\sigma|x:1]|y:2]\rangle \leadsto^* \langle \textbf{skip}; a,\ [[\sigma|x:0]|y:2]\rangle}\ \mathrm{T}_2$$

Demostración:

$$\frac{J \quad I}{\langle x := x > y\ ?\ 2\ *\ y : y;\ \textbf{while}\ x > 0\ \textbf{do} := x - y, [[\sigma|x:1]|y:2]\rangle \leadsto^* \langle \textbf{skip},\ [[\sigma|x:0]|y:2]\rangle}\ \mathrm{T}_2$$

Ejercicio 7

Se encuentra en el archivo `src/Eval1.hs`.

Ejercicio 8

Se encuentra en el archivo `src/Eval2.hs`.

Ejercicio 9

Se encuentra en el archivo `src/Eval3.hs`.

Ejercicio 10

Gramática abstracta

$$\begin{aligned}
\mathit{intexp}\ &::= \mathit{nat} \mid \mathit{var} \mid\ -_u\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ +\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ -_b\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ \times\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ \div\ \mathit{intexp} \\
&\mid\ \mathit{boolexp}\ ?\ \mathit{intexp}\ :\ \mathit{intexp} \\
\mathit{boolexp}\ &::= \textbf{true} \mid \textbf{false} \\
&\mid\ \mathit{intexp}\ ==\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ \neq\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ <\ \mathit{intexp} \\
&\mid\ \mathit{intexp}\ >\ \mathit{intexp} \\
&\mid\ \mathit{boolexp}\ \wedge\ \mathit{boolexp} \\
&\mid\ \mathit{boolexp}\ \vee\ \mathit{boolexp} \\
&\mid\ \neg\ \mathit{boolexp} \\
\mathit{comm}\ &:= \textbf{skip} \\
&\mid\ \mathit{var} = \mathit{intexp} \\
&\mid\ \mathit{comm};\ \mathit{comm} \\
&\mid\ \textbf{if}\ \mathit{boolexp}\ \textbf{then}\ \mathit{comm}\ \textbf{else}\ \mathit{comm} \\
&\mid\ \textbf{while}\ \mathit{boolexp}\ \textbf{do}\ \mathit{comm} \\
&\mid\ \textbf{repeat}\ \mathit{comm}\ \textbf{until}\ \mathit{boolexp}
\end{aligned}$$

Semántica operacional para comandos

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

$$\frac{}{\langle \mathbf{skip}; c_1, \sigma\rangle \rightsquigarrow \langle c_1, \sigma\rangle}\ \text{SEQ}_1 \qquad \frac{\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0', \sigma'\rangle}{\langle c_0; c_1, \sigma\rangle \rightsquigarrow \langle c_0'; c_1, \sigma'\rangle}\ \text{SEQ}_2$$

$$\frac{\langle b, \sigma\rangle \Downarrow_{exp} \mathbf{true}}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma\rangle \rightsquigarrow \langle c_0, \sigma\rangle}\ \text{IF}_1 \qquad \frac{\langle b, \sigma\rangle \Downarrow_{exp} \mathbf{false}}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1, \sigma\rangle \rightsquigarrow \langle c_1, \sigma\rangle}\ \text{IF}_2$$

$$\frac{\langle b, \sigma\rangle \Downarrow_{exp} \mathbf{true}}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \rightsquigarrow \langle c; \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle}\ \text{WHILE}_1 \qquad \frac{\langle b, \sigma\rangle \Downarrow_{exp} \mathbf{false}}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \rightsquigarrow \langle \mathbf{skip}, \sigma\rangle}\ \text{WHILE}_2$$

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