# Trabajo Práctico 1
# Análisis de Lenguajes de Programación


## Alumnas:

**Cipullo, Inés**

**Sullivan, Katherine**

Universidad Nacional de Rosario


2021

# Ejercicio 1

## Gramática Abstracta

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

### Gramática Concreta

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

## Ejercicio 2

Realizado en archivo `src/AST.hs`.

## Ejercicio 3

Realizado en archivo `src/Parser.hs`.

# Ejercicio 4

### Semántica Operacional Big-Step para Expresiones

$$\frac{}{\langle nv, \sigma\rangle \Downarrow_{exp} \langle \mathbf{nv}, \sigma\rangle}\ \text{NVAL} \qquad \frac{}{\langle x, \sigma\rangle \Downarrow_{exp} \langle \sigma x, \sigma\rangle}\ \text{VAR}$$

$$\frac{\langle e, \sigma\rangle \Downarrow_{exp} \langle n, \sigma'\rangle}{\langle -_u e, \sigma\rangle \Downarrow_{exp} \langle -n, \sigma'\rangle}\ \text{UMINUS} \qquad \frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 + e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 + n_1, \sigma''\rangle}\ \text{PLUS}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 - e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 - n_1, \sigma''\rangle}\ \text{BMINUS}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 * e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 * n_1, \sigma''\rangle}\ \text{MULT}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle \quad n_1 \neq 0}{\langle e_0 \div e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 \div n_1, \sigma''\rangle}\ \text{DIV}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 > e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 > n_1, \sigma''\rangle}\ \text{GT}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 < e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 < n_1, \sigma''\rangle}\ \text{LT}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0\ != e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 \neq n_1, \sigma''\rangle}\ \text{NOTEQ}$$

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \quad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0 == e_1, \sigma\rangle \Downarrow_{exp} \langle n_0 = n_1, \sigma''\rangle}\ \text{EQ} \qquad \frac{}{\langle bv, \sigma\rangle \Downarrow_{exp} \langle bv, \sigma\rangle}\ \text{BVAL}$$

$$\frac{\langle p, \sigma\rangle \Downarrow_{exp} \langle b, \sigma'\rangle}{\langle \neg p, \sigma\rangle \Downarrow_{exp} \langle \neg b, \sigma'\rangle}\ \text{NOT} \qquad \frac{\langle p_0, \sigma\rangle \Downarrow_{exp} \langle b_0, \sigma'\rangle \quad \langle p_1, \sigma'\rangle \Downarrow_{exp} \langle b_1, \sigma''\rangle}{\langle e_0 \vee e_1, \sigma\rangle \Downarrow_{exp} \langle b_0 \vee b_1, \sigma''\rangle}\ \text{OR}$$

$$\frac{\langle p_0, \sigma\rangle \Downarrow_{exp} \langle b_0, \sigma'\rangle \quad \langle p_1, \sigma'\rangle \Downarrow_{exp} \langle b_1, \sigma''\rangle}{\langle e_0 \wedge e_1, \sigma\rangle \Downarrow_{exp} \langle b_0 \wedge b_1, \sigma''\rangle}\ \text{AND}$$

Llamamos IASS a la asignación como expresión y ISEQ a la secuencialización de expresiones con el operador **,**

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

$$\frac{\langle e_0, \sigma\rangle \Downarrow_{exp} \langle n_0, \sigma'\rangle \qquad \langle e_1, \sigma'\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}{\langle e_0, e_1, \sigma\rangle \Downarrow_{exp} \langle n_1, \sigma''\rangle}\ \text{ISEQ}$$

# Ejercicio 5

### Determinismo de la relación de evaluación en un paso $\rightsquigarrow$

Queremos probar que dado un comando $c$ y un estado $\sigma$

$$\langle c, \sigma\rangle \rightsquigarrow \langle c', \sigma'\rangle \ \wedge\ \langle c, \sigma\rangle \rightsquigarrow \langle c'', \sigma''\rangle \implies c' = c'' \wedge \sigma' = \sigma'' \tag{1}$$

Para probarlo haremos inducción estructural sobre c

<u>**Caso 1**</u>:

Supongamos

$$c := \textbf{skip}$$

Como un comando de la forma **skip** termina la ejecución, este no deriva en un paso a ningún otro comando, por lo tanto, por premisas falsas en una implicancia resulta verdadero (1).

<u>**Caso 2**</u>:

Supongamos

$$c := v = e$$

Por la forma de $c$ la única regla de semántica operacional que se le puede aplicar resulta

$$\frac{\langle e, \sigma\rangle \Downarrow_{exp} \langle n, \sigma_1\rangle}{\langle c, \sigma\rangle \rightsquigarrow \langle \textbf{skip}, [\sigma_1|v : n]\rangle}\ \text{ASS}$$

De lo que entonces concluimos que necesariamente

$c' = c'' = \textbf{skip}$

$\sigma' = \sigma'' = \sigma_1$

por lo que vale (1).

<u>**Caso 3**</u>:

Supongamos

$$c := c_1; c_2$$

donde $c_1$ y $c_2$ son comandos.

De aquí tenemos dos opciones: que $c_1$ sea el comando **skip** o no.

Si $c_1$ es **skip**, luego, como **skip** no deriva en un paso a ningún otro comando, la única regla que podemos usar dada la estructura de $c$ es

$$\frac{}{\langle \mathbf{skip}; c_2, \sigma\rangle \rightsquigarrow \langle c_2, \sigma\rangle}\ \mathrm{SEQ}_1$$

De lo que entonces concluimos que necesariamente

$c' = c'' = c_2$

$\sigma' = \sigma'' = \sigma$

por lo que vale (1).

Ahora bien, si $c_1$ no es **skip** resulta que la única regla que podemos usar es

$$\frac{\langle c_1, \sigma\rangle \rightsquigarrow \langle c_1', \sigma_1\rangle}{\langle c_1; c_2, \sigma\rangle \rightsquigarrow \langle c_1'; c_2, \sigma_1\rangle}\ \mathrm{SEQ}_2$$

Por HI, sabemos que $c_1'$ es al único comando que puede derivar en un paso $c_1$. Por lo tanto, sabemos que necesariamente

$c' = c'' = c_1'; c_2$

$\sigma' = \sigma'' = \sigma$

por lo que vale (1).

<u>**Caso 4**</u>:

Supongamos

$$c := \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else} c_2$$

Tenemos dos opciones:

$$\langle b, \sigma\rangle \Downarrow_{exp} \langle \text{‘}\mathbf{true}\text{’}, \sigma_1\rangle \tag{2}$$

$$\langle b, \sigma\rangle \Downarrow_{exp} \langle \text{‘}\mathbf{false}\text{’}, \sigma_1\rangle \tag{3}$$

Como la relación $\Downarrow_{exp}$ es determinista si vale (2) no vale (3) y viceversa.

Supongamos primero que vale (2). Luego por esto y por la forma de $c$ la única regla de evaluación que se puede aplicar es

$$\frac{\langle b, \sigma\rangle \Downarrow_{exp} \langle \mathbf{true}, \sigma_1\rangle}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2, \sigma\rangle \rightsquigarrow \langle c_1, \sigma_1\rangle}\ \mathrm{IF}_1$$

De lo que resulta que necesariamente

$c' = c'' = c_1$

$\sigma' = \sigma'' = \sigma_1$

por lo que vale (1).

Ahora, supongamos que en cambio vale (3), luego la única regla de la que podemos hacer uso es

$$\frac{\langle b, \sigma\rangle \Downarrow_{exp} \langle \mathbf{false}, \sigma_1\rangle}{\langle \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2, \sigma\rangle \rightsquigarrow \langle c_2, \sigma_1'\rangle}\ \mathrm{IF}_2$$

De lo que resulta que necesariamente

$c' = c'' = c_2$

$\sigma' = \sigma'' = \sigma_1$

por lo que vale (1).

<u>**Caso 5**</u>:

Supongamos

$$c := \textbf{repeat } c_1 \textbf{ until } b$$

Por la forma de $c$ la única regla que puede ser utilizada es

$$\frac{}{\langle \textbf{repeat } c_1 \textbf{ until } b, \sigma\rangle \rightsquigarrow \langle c_1; \textbf{if } b \textbf{ then skip else repeat } b \textbf{ until } c_1, \sigma\rangle}\ \text{REPEAT}$$

De donde resulta necesariamente

$c' = c'' = c_1; \textbf{if } b \textbf{ then skip else repeat } b \textbf{ until } c_1$

$\sigma' = \sigma'' = \sigma$

por lo que vale (1).

Queda así demostrado el determinismo de la relación de evalución en un paso $\rightsquigarrow$.

## Ejercicio 6

Al hablar de la clausura transitiva de $\rightsquigarrow$ tenemos las siguientes reglas

$$\frac{\langle c, \sigma\rangle \rightsquigarrow \langle c', \sigma'\rangle}{\langle c, \sigma\rangle \rightsquigarrow^* \langle c', \sigma'\rangle}\ \text{T}_1 \qquad \frac{\langle c, \sigma\rangle \rightsquigarrow^* \langle c', \sigma'\rangle \quad \langle c', \sigma'\rangle \rightsquigarrow^* \langle c'', \sigma''\rangle}{\langle c, \sigma\rangle \rightsquigarrow^* \langle c'', \sigma''\rangle}\ \text{T}_2$$

**A:**

$a := \textbf{repeat } x = x - y \textbf{ until } x == 0$

$$\frac{\frac{}{\langle \textbf{skip; a}, \ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow \langle \textbf{a}, \ [[\sigma|x:1]|y:1]\rangle}\ \text{SEQ}_1}{\langle \textbf{skip; a}, \ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow^* \langle \textbf{a}, \ [[\sigma|x:1]|y:1]\rangle}\ \text{T}_1$$

**B:**

$a := \textbf{repeat } x = x - y \textbf{ until } x == 0$

$$\frac{\frac{}{\langle \textbf{a}, \ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow \langle x = x - y; \textbf{if } x == 0 \textbf{ then skip else a}, \ [[\sigma|x:1]|y:1]\rangle}\ \text{REPEAT}}{\langle \textbf{a}, \ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow^* \langle x = x - y; \textbf{if } x == 0 \textbf{ then skip else a}, \ [[\sigma|x:1]|y:1]\rangle}\ \text{T}_1$$

**C:**

$c := \textbf{if } x == 0 \textbf{ then skip else repeat } x = x - y \textbf{ until } x == 0$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x,\ [[\sigma|x:1]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{1},\ [[\sigma|x:1]|y:1]\rangle}\,\text{VAR} \qquad \dfrac{}{\langle y,\ [[\sigma|x:1]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{1},\ [[\sigma|x:1]|y:1]\rangle}\,\text{VAR}}{\langle x-y,\ [[\sigma|x:1]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{0},\ [[\sigma|x:1]|y:1]\rangle}\,\text{MINUS}}{\langle x=x-y,\ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow \langle \mathbf{skip},\ [[\sigma|x:0]|y:1]\rangle}\,\text{ASS}}{\langle x=x-y;\ \mathbf{c},\ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow \langle \mathbf{skip};\ \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle}\,\text{SEQ}_2}{\langle x=x-y;\ \mathbf{c},\ [[\sigma|x:1]|y:1]\rangle \rightsquigarrow^* \langle \mathbf{skip};\ \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_1$$

**D:**

$a :=$ **repeat** $x = x - y$ **until** $x == 0$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\langle 1,\ [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} \langle \mathbf{1},\ [[\sigma|x:2]|y:2]\rangle}\,\text{NVAL}}{\langle y=1,\ [[\sigma|x:2]|y:2]\rangle \Downarrow_{exp} \langle \mathbf{1},\ [[\sigma|x:2]|y:1]\rangle}\,\text{IASS}}{\langle x=y=1,\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle \mathbf{skip},\ [[\sigma|x:1]|y:1]\rangle}\,\text{ASS}}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow \langle \mathbf{skip};\ \mathbf{a},\ [[\sigma|x:1]|y:1]\rangle}\,\text{SEQ}_2}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \mathbf{skip};\ \mathbf{a},\ [[\sigma|x:1]|y:1]\rangle}\,\text{T}_1$$

**E:**

$c :=$ **if** $x == 0$ **then skip else repeat** $x = x - y$ **until** $x == 0$

$$\dfrac{\dfrac{}{\langle \mathbf{skip};\ \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle \rightsquigarrow \langle \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle}\,\text{SEQ}_1}{\langle \mathbf{skip};\ \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle \rightsquigarrow^* \langle \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_1$$

**F:**

$c :=$ **if** $x == 0$ **then skip else repeat** $x = x - y$ **until** $x == 0$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x,\ [[\sigma|x:0]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{0},\ [[\sigma|x:0]|y:1]\rangle}\,\text{VAR} \qquad \dfrac{}{\langle 0,\ [[\sigma|x:0]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{0},\ [[\sigma|x:0]|y:1]\rangle}\,\text{NVAL}}{\langle x==0,\ [[\sigma|x:0]|y:1]\rangle \Downarrow_{exp} \langle \mathbf{true},\ [[\sigma|x:0]|y:1]\rangle}\,\text{EQ}}{\langle \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle \rightsquigarrow \langle \mathbf{skip},\ [[\sigma|x:0]|y:1]\rangle}\,\text{IF}_1}{\langle \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle \rightsquigarrow^* \langle \mathbf{skip},\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_1$$

**DEM:**

$a :=$ **repeat** $x = x - y$ **until** $x == 0$

$c :=$ **if** $x == 0$ **then skip else repeat** $x = x - y$ **until** $x == 0$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{D \quad A}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \mathbf{a},\ [[\sigma|x:1]|y:1]\rangle}\,\text{T}_2 \quad B}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle x=x-y;\ \mathbf{c}\ [[\sigma|x:1]|y:1]\rangle}\,\text{T}_2 \quad C}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \mathbf{skip};\ \mathbf{c},\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_2 \quad E}{\langle x=y=1;\ \mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \mathbf{c}\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_2 \quad F}{\langle x=y=1;\mathbf{a},\ [[\sigma|x:2]|y:2]\rangle \rightsquigarrow^* \langle \mathbf{skip},\ [[\sigma|x:0]|y:1]\rangle}\,\text{T}_2$$

# Ejercicio 7

Realizado en el archivo `src/Eval1.hs`.

# Ejercicio 8

Realizado en el archivo `src/Eval2.hs`.

# Ejercicio 9

Realizado en el archivo `src/Eval3.hs`.

# Ejercicio 10

## Nueva gramática abstracta

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

## Nueva semántica operacional para comandos

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

$$\frac{}{\langle \textbf{skip}; c_1, \sigma \rangle \rightsquigarrow \langle c_1, \sigma \rangle} \text{ SEQ}_1$$

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

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

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

$$\frac{\langle e_1, \sigma \rangle \Downarrow_{exp} \langle n_1, \sigma' \rangle}{\langle \textbf{for } (e_1; e_2; e_3)\ c, \sigma \rangle \rightsquigarrow \langle \textbf{if } e_2 \textbf{ then } c;\ \textbf{for } (e_3; e_2; e_3)\ c \textbf{ else skip}, \sigma' \rangle} \text{ FOR}$$