Trabajo Práctico 1

Mellino, Natalia   Farizano, Juan Ignacio

Ejercicio 1

A continuación extendemos las sintaxis abstracta y concreta para incluir asignaciones de variables como expresiones enteras y para escribir una secuencia de expresiones enteras.

Sintaxis Abstracta

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

Sintaxis Concreta

$$
\begin{aligned}
digit &::= \text{'0'} \mid \text{'1'} \mid \ \dots \ \mid \text{'9'} \\
letter &::= \text{'a'} \mid \ \dots \ \mid \text{'Z'} \\
nat &::= digit \mid digit\ nat \\
var &::= letter \mid letter\ var \\
intexp &::= nat \\
&\mid var \\
&\mid \text{'-'}\ intexp \\
&\mid\ intexp\ \text{'+'}\ intexp \\
&\mid\ intexp\ \text{'-'}\ intexp \\
&\mid\ intexp\ \text{'*'}\ intexp \\
&\mid\ intexp\ \text{'/'}\ intexp \\
&\mid\ \text{'('}\ intexp\ \text{')'} \\
&\mid\ var\ \text{'='}\ intexp \\
&\mid\ intexp\ \text{','}\ intexp \\
boolesxp &::= \text{'true'} \mid \text{'false'} \\
&\mid\ intexp\ \text{'=='}\ intexp \\
&\mid\ intexp\ \text{'!='}\ intexp \\
&\mid\ intexp\ \text{'<'}\ intexp \\
&\mid\ intexp\ \text{'>'}\ intexp \\
&\mid\ boolexp\ \text{'\&\&'}\ boolexp \\
&\mid\ boolexp\ \text{'||'}\ boolexp \\
&\mid\ \text{'!'}\ boolexp \\
&\mid\ \text{'('}\ boolexp\ \text{')'} \\
com &::= \mathbf{skip} \\
&\mid\ var\ \text{'='}\ intexp \\
&\mid\ comm\ \text{';'}\ comm \\
&\mid\ \text{'if'}\ \ boolexp\ \ \text{'\{'}\ \ comm\ \ \text{'\}'} \\
&\mid\ \text{'if'}\ \ boolexp\ \ \text{'\{'}\ \ comm\ \ \text{'\}'}\ \ \text{'else'}\ \ \text{'\{'}\ \ comm\ \ \text{'\}'} \\
&\mid\ \text{'while'}\ \ boolexp\ \ \text{'\{'}\ \ comm\ \ \text{'\}'}
\end{aligned}
$$

Ejercicio 4:

Ejercicio 5:

Asumimos que la relación $\Downarrow_{exp}$ es determinista y procedemos por inducción en la última regla de derivación. Queremos probar : $c \rightsquigarrow c',\ c \rightsquigarrow c'' \Rightarrow c' = c''$

• Si $c \rightsquigarrow c'$ usando como última regla ASS:
c tiene la forma $\langle v = e, \sigma \rangle$ y tenemos la premisa $\langle e, \sigma \rangle\ \Downarrow_{exp}\ \langle n, \sigma' \rangle$, inmediatamente debido a la regla ASS obtenemos que $c' = \langle \mathbf{skip}, [\sigma' \mid v : n] \rangle$.

Supongamos entonces, que esta relación no es determinista, es decir que $c' \neq c''$. Por la forma que tiene c observemos que la única regla que podemos usar en la derivación $c \rightsquigarrow c''$ es la regla *ASS*, entonces tenemos que: $\langle v = e, \sigma\rangle \rightsquigarrow \langle \mathbf{skip}, [\sigma'' \,|v : n']\rangle$ con la premisa $\langle e, \sigma\rangle \Downarrow_{exp} \langle n', \sigma''\rangle$ Como $c' \neq c''$ vemos que $\sigma' \neq \sigma'' \vee n \neq n''$. Esto es una contradicción ya que por determinismo de la relación $\Downarrow_{exp}$ necesariamente debe ocurrir que $\sigma' = \sigma'' \wedge n = n''$.

$\therefore\ c' = c''$

- Si $c \rightsquigarrow c'$ usando como última regla *SEQ1*:

c tiene la forma $\langle \mathbf{skip}; c_1, \sigma\rangle$ y tenemos que $c' = \langle c_1, sigma\rangle$. Por la forma que tiene c, observemos que no es posible aplicar ninguna otra regla de inferencia, por lo tanto en la derivación $c \rightsquigarrow c''$ se tiene que $c'' = \langle c_1, \sigma\rangle$.

$\therefore\ c' = c''$

- Si $c \rightsquigarrow c'$ usando como última regla *SEQ2*:

Tenemos entonces que c tiene la forma: $\langle c_0; c_1, \sigma\rangle$ y además: $\langle c_0,\ \sigma\rangle \rightsquigarrow \langle c_0', \sigma'\rangle$. Ahora, analicemos que pasa en $c \rightsquigarrow c''$: ¿qué reglas podemos aplicar?

- No podemos aplicar *SEQ1* ya que esto nos diría que $c_0 = \mathbf{skip}$ y esto nos contradice $\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0',\ \sigma\rangle$ ya que si $c_0 = skip$ el estado σ no debería pasar a ser σ'.
- En cuanto a las demás reglas, es claro que por la forma que tiene c no es posible aplicarlas.

Por lo tanto, la única regla que podemos aplicar es *SEQ2*. Si suponemos que $c' \neq c''$ tenemos entonces que c tiene la forma: $\langle c_0; c_1, \sigma\rangle$ y además: $\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0'', \sigma''\rangle$. Luego sigue que c'' tiene la forma: $\langle c_0''; c1, \sigma''\rangle$. Sin embargo, observemos que $\langle c_0, \sigma\rangle \rightsquigarrow \langle c_0',\ \sigma'\rangle$ es una subderivación y por lo tanto vale nuestra Hipótesis Inductiva, es decir necesariamente:

$$\langle c_0,\ \sigma\rangle \rightsquigarrow \langle c_0', \sigma'\rangle = \langle c_0,\ \sigma\rangle \rightsquigarrow \langle c_0'',\ \sigma''\rangle$$

Por lo tanto, $c_0' = c_0''$ y $\sigma' = \sigma''$.

$\therefore\ c' = c''$

- Si $c \rightsquigarrow c'$ usando como última regla *IF1*:

c tiene la forma $\langle \mathbf{if}\ \text{b}\ \mathbf{then}\ c_0\ \mathbf{else}\ c_1,\ \sigma\rangle$ y además $\langle b, \sigma\rangle \Downarrow_{exp} \langle \mathbf{true}, \sigma'\rangle$. Entonces c' tiene la forma $\langle c_0, \sigma'\rangle$. ¿Qué pasa en el caso de $c \rightsquigarrow c''$. Sabemos que por la forma que tiene c y por el hecho de que $\langle b, \sigma\rangle \Downarrow_{exp} \langle \mathbf{true}, \sigma'\rangle$ sólo podemos aplicar la regla *IF1*. Recordemos que la relación $\Downarrow_{exp}$ es determinista y por eso podemos decir que sólo podamos aplicar *IF1*. inmediatamente de esta conclusión surge que $c' = c''$.

- Si $c \rightsquigarrow c'$ usando como última regla *IF2*:

(este caso es análogo a la regla *IF1*)

- Si $c \rightsquigarrow c'$ usando como última regla *WHILE1*: tenemos entonces que
  - c tiene la forma $\langle \mathbf{while}\ \text{b}\ \mathbf{do}\ c_0, \sigma\rangle$.
  - $\langle b, \sigma\rangle \Downarrow_{exp} \langle \mathbf{true}, \sigma'\rangle$
  - $c' = \langle c0; \mathbf{while}\ \text{b}\ \mathbf{do}\ c_0, \sigma'\rangle$

Luego en $c \rightsquigarrow c''$ ocurre que por la forma de c y por el determinismo de $\Downarrow_{exp}$, la única regla que podemos usar es *WHILE1*. Por estas dos cosas surge inmediatamente que $c' = c''$.

- Si $c \rightsquigarrow c'$ usando como última regla *WHILE2*: tenemos entonces que
  - c tiene la forma $\langle \textbf{while } \mathrm{b} \textbf{ do } c_0,\ \sigma \rangle$.
  - $\langle b, \sigma \rangle \Downarrow_{exp} \langle \textbf{false}, \sigma' \rangle$
  - $c' = \langle \textbf{skip}, \sigma' \rangle$

Luego en $c \rightsquigarrow c''$ ocurre que por la forma de c y por el determinismo de $\Downarrow_{exp}$, la única regla que podemos usar es *WHILE1*. Por estas dos cosas surge inmediatamente que $c' = c''$.

Por lo tanto, concluimos que la relación $\rightsquigarrow$ es determinista.