EJERCICIOS SEMANTICA OPERACIONAL EXTENSIONES DE WHILE

Ejercicio 3.9

Denotemos por $B \equiv$ begin D_V ; P ; Q, B_ℓ end, con P y Q las dos declaraciones de p y q , y B_ℓ el bloque local.

Trabajaremos con env_P "cualquiera", cuyos valores no se van a utilizar.

Entonces, partiendo de s_0 la aplicación de $[var_{ns}]$ genera s_D con $s_D\,x = 0$. Ahora al aplicar la parte de procedimientos de $[block_{ns}]$ $upd_P(P;Q,\ env_P)$ genera env'_P que luego examinaremos.

Nos queda "calcular" $env'_P \vdash \langle B_\ell,\ s_D \rangle \rightarrow s_F$.

Denotemos por $B_\ell \equiv$ begin D_ℓ ; P' ; C end con P' la "declaración local" y $C \equiv$ call q ; y:= x .

Al aplicar de nuevo $[block_{ns}]$, $[var_{ns}]$ genera s_ℓ con $s_\ell\,x = 5$, y a continuación obtenemos $env''_P = upd_P(P',\ env'_P)$ con $env''_P\,p = C_\ell$ siendo $C_\ell \equiv (x:= x+1)$, y finalmente tenemos $env''_P \vdash \langle C ;\ s_\ell \rangle$, donde la aplicación de $[call^{rec}_{ns}]$ nos requiere revisar el valor de $env'_P\,q$, que vemos es (call p).

Aplicamos entonces de nuevo $[call^{rec}_{ns}]$, lo que requiere el valor de $env''_P\,p$ que "sigue siendo" C_ℓ, de modo que obtenemos s_1 con $s_1\,x = s_\ell\,x + 1 = 5+1 = 6$. Y aplicando $[ass_{ns}]$ obtendríamos $\boxed{s_2\,y = s_1\,x = 6}$.

Ejercicio 3.11

Veamos cuando hemos aplicado $[call^{rec}_{ns}]$ arriba, y apliquemos en su lugar $[call_{ns}]$. La aplicación al evaluar C nos requiere $env''_P\,q = (\text{call } p\ ;\ env'''_P)$ siendo $env'''_P = upd'_P(P\ ;\ env_P)$ el entorno producido por la upd'_P "revisada" tras declarar sólo P.

Vemos que $env'''_P\,p = (x:=x*2\ ;\ env_P)$, con lo que al aplicar $[call_{ns}]$ utilizamos $[ass_{ns}]$ para llegar a s'_1 con $s'_1\,x = 2\cdot s_\ell\,x = 10$ y aplicando de nuevo $[ass_{ns}]$ para terminar la evaluación de C

llegamos a $\boxed{s'_2\ y = s'_1\ x = 10}$

Evidentemente, en este caso el resultado habría sido el mismo si hubiésemos aplicado [call$^{rec}_{ns}$] ya que cuando hemos utilizado el valor de env'''_p no hemos tenido que utilizar $env'''_p\ q$, ni cuando hemos utilizado env_p hemos "vuelto a llamar" a p, pues P no es un "procedimiento recursivo.

<u>Ejercicio 3.13</u>

Incorporamos ahora el tratamiento estático de las variables. Observad que al no haber ningún condicional (ni bucle) en B el "flujo" va a ser exactamente el mismo que en el ejercicio anterior, pero los estados s'_1 y s'_2 podrían variar, si las variables manejadas ahora fueran distintas. Hemos entonces ver cómo se crea y accede a la memoria para ver quiénes serán los correspondientes s''_1 y s''_2.

El que s_0 no se use en realidad nunca, se corresponde ahora con que "podemos suponer" que partimos de una memoria "vacía".

Entonces al aplicar [block$_{ns}$], la aplicación de $\to_D$ genera

$x \longrightarrow \overset{d^1_x}{\boxed{0}}$ cuando hemos "ejecutado" D_V

$env'_V\ x = d^1_x$

Mientras que al "ejecutar" D_ℓ generamos $x \longrightarrow \overset{d^2_x}{\boxed{5}}$

$env''_V\ x = d^2_x$

Ello nos lo encontramos a lo largo de la evaluación de

$$env_V,\ env_P \vdash \langle B, sto\rangle \to sto_F .$$

Al "evaluar" B_ℓ, la "llamada" en C a q nos "llevaba" a hacer una nueva "llamada" a p (manejando $env'''_p\ p$) que sigue teniendo como "cuerpo" $x := x+2$, pero ahora hemos de utilizar el correspondiente entorno de variables "guardado" en $env_p\ p$ que según la correspondiente definición de upd''_p es env'_V, con lo que al aplicar [ass ns] para "ejecutar" $x := x * 2$ obtenemos sto' con $sto'\ d^1_x = (sto\ d^1_x) * 2 = 0 * 2 = 0$, de modo que $sto' = sto$.

y al aplicar [ass ns] para terminar de "evaluar" C_1 "utilizamos" la dirección de y en env''_V, que obviamente es la misma que en env'_V y que en env_V $\quad y \longrightarrow \boxed{?}^{\,d_y}$ $\quad \begin{cases} env_V\ y = d_y \\ sto\ d_y = ? \end{cases}$

con lo que obtenemos sto'' con $sto''\ d_y = sto'\ \underline{\underline{d^2_x}} = 5$, $\boxed{5}^{\,d_y}$

de modo que el correspondiente "estado compuesto" s'' sería

$$s''\ y = (sto'' \circ env''_V)\ y = sto''\ d_y = 5$$
$$s''\ x = (sto'' \circ env''_V)\ x = sto''\ d^2_x = 5$$

Aunque también "mantendríamos" $sto''\ d^1_x = 0$, que "volvería a ser accesible" si "detrás" de B_ℓ "añadiésemos" más instrucciones dentro del "cuerpo" de B, mientras se habría "perdido" definitivamente el acceso a d^2_x, ya que "al salirnos" del bloque "local" B_ℓ "recuperaríamos" el entorno env'_V, y aunque la memoria se maneja de forma "dinámica" (sto'' sigue totalmente vigente) el "efecto" es como si hubiese sido estático ya que d^2_x sigue ahí, y conteniendo 5, pero "como si nada".
