DOMINIOS RECURSIVOS

El dominio Ω que vimos, satisface el siguiente isomorfismo:

$$(*) \qquad \Omega \approx (\Sigma' + Z \times \Omega)_\perp$$

$\phi \in \Omega \to (\Sigma' + Z \times \Omega)_\perp$
$\psi \in (\Sigma' + Z \times \Omega)_\perp \to \Omega$

ϕ y ψ continuas tales que:

$$\phi\ x = \begin{cases} \perp & \text{si } x = \langle\rangle \\ L0\ y & \text{si } x = \langle y\rangle \text{ con } y \in \Sigma' \\ L1\ \langle n, w\rangle & \text{si } x = \langle n\rangle +\!+\ w \end{cases}$$

$$\psi\ x = \begin{cases} \langle\rangle & \text{si } x = \perp \\ \langle y\rangle & \text{si } x = L0\ y \\ \langle n\rangle +\!+\ w & \text{si } x = L1\ \langle n, w\rangle \end{cases}$$

ϕ y ψ son morfismos (preservan todo) y son inversas mutuas. Son isomorfismos.

Notación:

$$\begin{array}{l} \Sigma \xrightarrow{Lnorm} \Sigma' \\ \Sigma \xrightarrow{Labnorm} \Sigma' \\ \Sigma' \xrightarrow{L0} \Sigma' + Z \times \Omega \\ Z\times\Omega \xrightarrow{L1} \Sigma' + Z \times \Omega \\ \Sigma' + Z \times \Omega \xrightarrow{Lbottom} (\Sigma' + Z \times \Omega)_\perp \xrightarrow{\psi} \Omega \end{array}$$

$Lterm = \psi\ .\ Lbottom\ .\ L0\ .\ Lnorm \in \Sigma \to \Omega$
$Labort = \psi\ .\ Lbottom\ .\ L0\ .\ Labnorm \in \Sigma \to \Omega$
$Lout = \psi\ .\ Lbottom\ .\ L1 \in Z \times \Omega \to \Omega$
$\perp_\Omega = \psi(\perp) \in \Omega$

Ecuaciones:

$[[skip]]s = Lterm\ s$
$[[fail]]s = Labort\ s$
$[[v := e]]s = Lterm\ [s \mid v : [[e]]s]$

$$[[\text{if } b \text{ then } c0 \text{ else } c1]]s = \begin{cases} [[c0]]s & \text{si } [[b]]s \\ [[c1]]s & \text{c.c.} \end{cases}$$

$[[c0;c1]]s = [[c1]]_* ([[c0]]s)$
donde, dada una $f \in \Sigma \to \Omega$, denotamos por f_* la siguiente extensión de f a Ω. Así, $f_* \in \Omega \to \Omega$ continua definida por:

$$f_*\ x = \begin{cases} \perp_\Omega & \text{si } x = \perp_\Omega \\ f\ s & \text{si } x = Lterm\ s \\ Labort\ s & \text{si } x = Labort\ s \\ Lout\ \langle n, f_*\ w\rangle & \text{si } x = Lout\ \langle n,w\rangle \end{cases}$$

En efecto, la clase pasada definimos así:

$$f_*\ x = \begin{cases} \langle n1,..,nk\rangle +\!+\ f\ s & \text{si } x = \langle n1,..,nk,s\rangle \\ x & \text{c.c.} \end{cases}$$

o

$$f_*\ x = \begin{cases} \langle n1,..,nk\rangle & \text{si } x = \langle n1,..,nk\rangle \\ \langle n1,..,nk\rangle +\!+\ f\ s & \text{si } x = \langle n1,..,nk,s\rangle \\ \langle n1,..,nk,\langle abort,s\rangle\rangle & \text{si } x = \langle n1,..,nk,\langle abort,s\rangle\rangle \\ \langle n1,..,nk,...\rangle & \text{si } x = \langle n1,..,nk,...\rangle \end{cases}$$

Esta también se puede escribir así, f_* continua tal que:

$$f_*\ x = \begin{cases} \langle\rangle & \text{si } x = \langle\rangle \\ f\ s & \text{si } x = \langle s\rangle \\ \langle\langle abort,s\rangle\rangle & \text{si } x = \langle\langle abort,s\rangle\rangle \\ \langle n\rangle +\!+\ f_*\ w & \text{si } x = \langle n\rangle +\!+\ w \end{cases}$$

$[[\text{catchin } c0 \text{ with } c1]]s = [[c1]]_\clubsuit ([[c0]]s)$

donde, dada una $f \in \Sigma \to \Omega$, denotamos por $f_♣$ la siguiente extensión de f a Ω. Así, $f_♣ \in \Omega \to \Omega$ continua definida por:

$$f_♣\ x = \begin{cases} \bot_\Omega & \text{si } x = \bot_\Omega \\ \text{Lterm } s & \text{si } x = \text{Lerm } s \\ f\ s & \text{si } x = \text{Labort } s \\ \text{Lout } \langle n, f_♣\ w\rangle & \text{si } x = \text{Lout } \langle n,w\rangle \end{cases}$$

$[[\text{while } b \text{ do } c]] = \sup'(F^i \bot')$

$$\text{donde } F\ w\ s = \begin{cases} w_*\ ([[c]]s) & \text{si } [[b]]s \\ \text{Lterm } s & \text{c.c.} \end{cases}$$

$[[\text{newvar } v:= e \text{ in } c]]s = (\lambda s'\in\Sigma.\ [s' \mid v : s\ v])_\dagger\ ([[c]][s \mid v : [[e]]s])$

donde, dada una $f \in \Sigma \to \Sigma$, denotamos por $f_\dagger$ la siguiente extensión de f a Ω. Así, $f_\dagger \in \Omega \to \Omega$ continua definida por:

$$f_\dagger\ x = \begin{cases} \bot_\Omega & \text{si } x = \bot_\Omega \\ \text{Lterm } (f\ s) & \text{si } x = \text{Lterm } s \\ \text{Labort } (f\ s) & \text{si } x = \text{Labort } s \\ \text{Lout } \langle n, f_\dagger\ w\rangle & \text{si } x = \text{Lout } \langle n,w\rangle \end{cases}$$

$[[!e]]s = \text{Lout } \langle [[e]]s, \text{Lterm } s\rangle$

INPUT

Agregamos el comando

<comm> ::= ?<var>

Todas las ecuaciones como (*) que usen +, x, $()_\bot$ y -> tienen solución. Puede considerarse Ω definido por (*). Para contemplar también el input, vamos a modificar dicha definición.

Agregamos al isomorfismo:

$\Omega \approx (\Sigma' + Z \times \Omega + (Z \to \Omega))_\bot$

$\phi \in \Omega \to (\Sigma' + Z \times \Omega + (Z \to \Omega))_\bot$
$\psi \in (\Sigma' + Z \times \Omega + (Z \to \Omega))_\bot \to \Omega$

Notación:

$$\begin{tikzcd} \Sigma \arrow[r, "Lnorm"] & \Sigma' \arrow[r, "L0"] & \Sigma' + Z \times \Omega + (Z \to \Omega) \arrow[r, "Lbottom"] & (\Sigma' + Z \times \Omega + (Z \to \Omega))_\bot \arrow[r, "\psi"] & \Omega \\ \Sigma \arrow[ur, "Labnorm"'] & Z\times\Omega \arrow[ur, "L1"] & & & \\ & (Z \to \Omega) \arrow[uur, "L2"'] & & & \end{tikzcd}$$

$\text{Lterm} = \psi \ .\ \text{Lbottom} \ .\ \text{L0} \ .\ \text{Lnorm} \in \Sigma \to \Omega$
$\text{Labort} = \psi \ .\ \text{Lbottom} \ .\ \text{L0} \ .\ \text{Labnorm} \in \Sigma \to \Omega$
$\text{Lout} = \psi \ .\ \text{Lbottom} \ .\ \text{L1} \in Z \times \Omega \to \Omega$
$\text{Lin} = \psi \ .\ \text{Lbottom} \ .\ \text{L2} \in (Z \to \Omega) \to \Omega$
$\bot_\Omega = \psi(\bot) \in \Omega$

Ecuaciones:

$[[\text{skip}]]s = \text{Lterm } s$
$[[\text{fail}]]s = \text{Labort } s$
$[[v:= e]]s = \text{Lterm } [s \mid v : [[e]]s]$

$$[[\text{if } b \text{ then } c0 \text{ else } c1]]s = \begin{cases} [[c0]]s & \text{si } [[b]]s \\ [[c1]]s & \text{c.c.} \end{cases}$$

$[[c0;c1]]s = [[c1]]_*\ ([[c0]]s)$

donde, dada una f ∈ Σ -> Ω, denotamos por f_* la siguiente extensión de f a Ω. Así, f_* ∈ Ω -> Ω continua definida por:

```
        ⎧⊥_Ω              si x = ⊥_Ω
        |f s               si x = Lterm s
f_* x = {Labort s          si x = Labort s
        |Lout <n,f_* w>    si x = Lout <n,w>
        ⎩Lin (f_* . g)     si x = Lin g
```

```
[[catchin c0 with c1]]s = [[c1]]_♣ ([[c0]]s)
```

donde, dada una f ∈ Σ -> Ω, denotamos por f_♣ la siguiente extensión de f a Ω. Así, f_♣ ∈ Ω -> Ω continua definida por:

```
         ⎧⊥_Ω                 si x = ⊥_Ω
         |Lterm s             si x = Lerm s
f_♣ x =  {f s                  si x = Labort s
         |Lout <n,f_♣ w>       si x = Lout <n,w>
         ⎩Lin (f_♣ . g)        si x = Lin g
```

```
[[while b do c]] = sup'(F^i ⊥')

              ⎧w_* ([[c]]s)     si [[b]]s
donde F w s = {
              ⎩Lterm s          c.c.
```

```
[[newvar v:= e in c]]s = (λs'∈Σ. [s' | v : s v])_† ([[c]][s | v : [[e]]s])
```

donde, dada una f ∈ Σ -> Σ, denotamos por f_† la siguiente extensión de f a Ω. Así, f_† ∈ Ω -> Ω continua definida por:

```
        ⎧⊥_Ω                si x = ⊥_Ω
        |Lterm (f s)         si x = Lterm s
f_† x = {Labort (f s)        si x = Labort s
        |Lout <n,f_† w>      si x = Lout <n,w>
        ⎩Lin (f_† . g)       si x = Lin g
```

```
[[!e]]s = Lout <[[e]]s, Lterm s>
[[?v]]s = Lin (λn∈Z. Lterm [s | v : n])
```