Examen de sémantique des langages de programmation

Avril 2005

1 lambda-calcul

2 lambda-calcul

1. $x\ x\ \lambda x.x$,

2. Une première solution est de réutiliser la notion de paires fournies en cours et de considérer que $n\ x\ y$ devrait être la paire de T et de la paire de x et y, tandis que l est n'importe quelle paire dont le premier élément est F. La fonction dec est alors la première projection et retourne donc bien T pour $n\ x\ y$, tandis que la les fonctions π_1 et π_2 appliquent d'abord la seconde projection à leur argument, puis la projection qui permet d'atteindre la bonne branche. Ceci donne les fonctions suivantes:

   - $n \equiv \lambda xyz.z\ T \lambda t.t\ x\ y$,
   - $l \equiv \lambda z.z\ F\ F$,
   - $dec \equiv \lambda t.t\ \lambda uv.u$,
   - $\pi_1 \equiv \lambda t.t(\lambda uv.v)(\lambda uv.u)$, item $\pi_2 \equiv \lambda t.t(\lambda uv.v)(\lambda uv.v)$.

   Une deuxième solution est de construire directement des triplets au lieu d'imbriquer des paires:

   - $n \equiv \lambda xyz.z\ T\ x\ y$,
   - $l \equiv \lambda x.x\ F$,
   - $dec \equiv \lambda t.t \lambda xyz.x$,
   - $\pi_1 \equiv \lambda t.t \lambda xyz.y$,
   - $\pi_2 \equiv \lambda t.t \lambda xyz.z$.

Il faut vérifier la correction de ces solutions en faisant les réductions d'usage.

3 lambda calcul typé

3. $\lambda f : t_1 \to t_2.\lambda g : t_2 \to t_3.\lambda h : t_1 \to t_2 \to t_2 \to t_3 \to t_4.\lambda x : t_1.h\, x\, (f\, x)(f\, x)(g\, (f\, x))$
   $\lambda f : t_1 \to t_2 \to t_3.\lambda g : t_2 \to t_3 \to t_4.lambdax : t_1.\lambda y : t_2.g\ y\ (f\ x\ y)$

4 Sémantique naturelle

4. Posons $\rho_0 \equiv (x,1)\cdot(y,0)\cdot\emptyset$, $\rho_1 \equiv (x,2)\cdot(y,0)\cdot\emptyset$, $\rho_2 \equiv (x,2)\cdot(y,4)\cdot\emptyset$, $\rho_3 \equiv (x,0)\cdot(y,4)\cdot\emptyset$, $I \equiv$ `while(x >0){x:=x+1;{y:=x+2;x:=x-2}}`. La dérivation est décomposée en plusieurs sous-dérivations:

$$D_1 \equiv \dfrac{\dfrac{\overline{\rho_1 \vdash \mathtt{x} \to 2} \quad \overline{\rho_1 \vdash 2 \to 2}}{\rho_1 \vdash \text{x+2} \to 4} \qquad \dfrac{\overline{(y,0)\cdot\emptyset \vdash y,4 \mapsto (y,4)\cdot\emptyset} \quad (x \neq y)}{\rho_1 \vdash y,4 \mapsto \rho_2}}{\rho_1 \vdash \text{y:=x+2} \leadsto \rho_2}$$

$$D_2 \equiv \dfrac{\dfrac{\overline{\rho_2 \vdash x \to 2} \quad \overline{\rho_2 \vdash -2 \to -2}}{\rho_2 \vdash \mathtt{x\ +\ (-2)} \to 0} \qquad \overline{\rho_2 \vdash x,0 \mapsto \rho_3}}{\rho_2 \vdash \text{x:=x-2} \leadsto \rho_3}$$

$$D_3 \equiv \dfrac{D_1 \qquad D_2}{\rho \vdash \mathtt{y{:}=\ x+2;\ x{:}=\ x-2} \leadsto \rho_3}$$

$$D_4 \equiv \dfrac{\dfrac{\overline{\rho_0 \vdash x \to 1} \quad \overline{\rho_0 \vdash 1 \to 1}}{\rho_0 \vdash \text{x+1} \to 2} \qquad \overline{\rho_0 \vdash x,2 \mapsto \rho_1}}{\rho_0 \vdash \text{x:=x+1} \leadsto \rho_1}$$

$$D_5 \equiv \dfrac{D_4 \qquad D_3}{\rho_0 \vdash \text{x:=x+1;\{y:=x+2;x:=x-2\}} \leadsto \rho_3}$$

$$\dfrac{\dfrac{\overline{\rho_0 \vdash x \to 1 \rho_0 \vdash 0 \to 0} \quad (1 > 0)}{\rho_0 \vdash \mathtt{x>0} \leadsto \mathtt{true}} \qquad D_5 \qquad \dfrac{\dfrac{\overline{\rho_3 \vdash x \to 0} \quad \overline{\rho_3 \vdash 0 \to 0} \quad (0 \leq 0)}{\rho_3 \vdash \mathtt{x} > 0 \to \mathtt{false}}}{\rho_3 \vdash I \leadsto \rho_3}}{\rho_0 \vdash I \leadsto \rho_3}$$

5 Sémantique dénotationelle

5. Un ordre partiel est un ensemble de couples. Cet ensemble contient toujours $\{(a,a),(b,b),(c,c)\}$. Cet ensemble est lui-même un ordre partiel. Par ailleurs, il y a six possibilités où un seul couple (x,y) est ajouté. Voici un exemple: $\{(a,a),(b,b),(c,c),(a,b)\}$, il faut ensuite ajouter trois cas où l'un des élement est minimal, mais les deux autres ne sont pas comparables, voici un exemple $\{(a,a),(b,b),(c,c),(a,b),(a,c)\}$, et trois cas où l'un des éléments est maximal mais les deux autres ne sont pas comparables , voici un exemple $\{(a,a),(b,b),(c,c)(a,c),(b,c)\}$. Enfin il faut ajouter 6 cas où l'ordre est total, voici un exemple $\{(a,a),(b,b),(c,c),(a,b),(b,c),(a,c)\}$. En tout on trouve donc 1+6+3+3+3+6 ordres partiels différents.

6. $f = \lambda x.\bot$ et $g = \lambda x.1$, $f_n(x) = \bot$ si $n \leq x$ et $f_n(x) = 1$ si $x < n$.

6 Sémantique axiomatique

7. Voici le programme

```
{0 <= x}
y:=0;
while(y*y*y <= x) do
 {invariant (y-1)^3 <= x}
 y := y+1;
done;
y := y-1
{y^3 <= x < (y+1)^3}
```

Les conditions de vérifications engendrées par `vc` pour ce programme sont:

```
0 <= x => (0-1)^3 <= x

(y-1)^3 <= x /\ y*y*y <= x => ((y+1)-1)^3 <= x

(y-1)^3 <= x /\ not(y*y*y <= x) => (y-1)^3 <= x < ((y-1)+1)^3
```

Voici une solution sans multiplications dans le code, qui réutilise un exercice vu en cours.

```
{0 <= x}
y := 0;
z := 0;
t := 1;
u := 6;
while(z <= x) do
 { invariant z = y^3 /\ t = 3*y^2+3*y+1 /\
             u = 6*y+6 /\ (y-1)^3 <= x }
 z:= z+t;
 t:= t+u;
 u:= u+6;
 y:= y+1
done
y := y - 1
{y^3 <= x <= (y+1)^3}
```

Les conditions de vérification sont les suivantes:

```
0 <= x => 0=0^3 /\ 1=3*0^2+3*0+1 /\ 6=6*0+6 /\ (0-1)^3 <= x

(z = y^3 /\ t = 3*y^2+3*y+1 /\ u = 6*y+6 /\ (y-1)^3 <= x)/\
z <= x
```

```
=>
 z+t = (y+1)^3 /\ t+u = 3*(y+1)^2+3*(y+1)+1 /\
 u+6 = 6(y+1) + 6 /\ ((y+1)-1)^3 <= x

z = y^3 /\ t=3*y^2+3*y+1 /\ u = 6*y+6 /\ (y-1)^3 <= x /\
not(z <= x)
=>
(y-1)^3 <= x < ((y-1)+1)^3
```