

Travaux Dirigés de Programmation Fonctionnelle

«Compilation»

I Combinateurs

Démontrer les égalités suivantes entre des expressions applicatives utilisant des combinateurs :

1. $S K K 1 = I 1$
2. $S (K (+ 1)) (K 2) 3 = K (+ 1 2) 3$
3. $S (S (K +) (K 2)) I 3 = + 2 3$
4. $Apply (Pair (Curry Snd) (Quote 2) (Quote 1)) = 2$

On rappelle ci-dessous la définition des combinateurs S, K et I et des combinateurs catégoriques :

$$\begin{aligned} S g d x &= (g x) (d x) \\ K x y &= x \\ I x &= x \\ Fst (x, y) &= x \\ Snd (x, y) &= y \\ (Quote x) y &= x \\ (Curry f) y z &= f (y, z) \\ (Pair x y) z &= (x z, y z) \\ Apply (x, y) &= x y \end{aligned}$$

II Compilation

Compiler les λ -expression suivantes (on devra au préalable utiliser la notation de De Bruijn).

1. $\lambda x. (+ 1 x) 2$
2. $\lambda x. \lambda y. x 1 2$
3. $\lambda x. (x 4 (\lambda x. x 3)) +$

On rappelle ci-dessous la définition de la fonction, *comb*, de traduction d'une λ -expression en code pour la CAM :

$$\begin{aligned}
comb(Ln) &= \underbrace{\text{Fst} (\text{Fst} (\dots (\text{Fst} \text{Snd}) \dots))}_{n \times} \\
comb(k) &= \text{Quote } k \\
comb(f) &= \text{Curry} (\text{Snd } f) \\
comb(M N) &= \text{Apply} (\text{Pair} (comb(M)) (comb(N))) \\
comb(\lambda.M) &= \text{Curry} (comb(M))
\end{aligned}$$

III Séquence d'instructions

Traduire les expressions applicatives de l'exercice II en une séquence d'instructions pour la CAM. On rappelle ci-dessous les instructions correspondantes à chaque combinateur :

Combinateur	Notation
Fst	[Fst]
Snd	[Snd]
Quote x	[' x]
Curry f	[$\wedge(f)$]
Pair $x y$	[<@ x [,]@ y @>]
Apply c	c @[app]

IV Exécution du code

Dérouler l'exécution par la CAM des séquences d'instructions de l'exercice III. On rappelle ci-dessous la fonction de transition d'état de la CAM :

$$\begin{array}{llll}
((s, t) & \text{Fst} :: C & S) & \Rightarrow (s & C & S) \\
((s, t) & \text{Snd} :: C & S) & \Rightarrow (t & C & S) \\
(s & 'c :: C & S) & \Rightarrow (c & C & S) \\
(s & \wedge(C) :: C_1 & S) & \Rightarrow ((C s) & C_1 & S) \\
(s & \langle :: C & S) & \Rightarrow (s & C & s :: S) \\
(t & , :: C & s :: S) & \Rightarrow (s & C & t :: S) \\
(t & \rangle :: C & s :: S) & \Rightarrow ((s, t) & C & S) \\
((C s, t) & \text{app} :: C_1 & S) & \Rightarrow ((s, t) & C@C_1 & S) \\
(m & + :: C & S) & \Rightarrow (+ m & C & S) \\
((m, n) & + :: C & S) & \Rightarrow (\text{add}(m, n) & C & S)
\end{array}$$