

TD de *Sémantique des langages de programmation* n° 3

Lambda calcul

Programmation

Exercice 1 Arithmétique dans le λ -calcul:

Soit $\bar{n} = \lambda x \lambda y \underbrace{(x(x \dots (xy) \dots))}_{n \times}$ une représentation des entiers naturels en λ -calcul.

Définir:

- Un terme S qui réalise la fonction successeur, c.à.d. tel que pour tout entier naturel n , $S\bar{n} \rightarrow_{\beta}^* \overline{n+1}$.
- Un terme A pour la somme de deux entiers ($(A\bar{n})\bar{m} \rightarrow_{\beta}^* \overline{n+m}$).
- Un terme M pour la multiplication de deux entiers.
- Un terme E pour l'exponentiation ($(E\bar{n})\bar{m} \rightarrow_{\beta}^* \overline{m^n}$).

Exercice 2 Couples et iteration

- Pour M et N λ -termes donnés, définir des termes $\langle M, N \rangle$, π_1 et π_2 tels que $\pi_1 \langle M, N \rangle \rightarrow_{\beta}^* M$ et $\pi_2 \langle M, N \rangle \rightarrow_{\beta}^* N$.
- Définir un codage des booléens et du *if_then_else* en λ -calcul.
- Définir un terme P qui réalise la fonction prédécesseur, c.à.d. tel que pour tout entier naturel n ,

$$P\bar{n} \rightarrow_{\beta}^* \begin{cases} \bar{0} & \text{si } n = 0 \\ \overline{n-1} & \text{sinon} \end{cases}$$

- Définir un terme F qui réalise la fonction de Fibonacci, c.à.d. tel que pour tout entier naturel n , $F\bar{n} \rightarrow_{\beta}^* \overline{fib(n)}$, où

$$fib(n) = \begin{cases} 1 & \text{si } n \leq 1 \\ fib(n-1) + fib(n-2) & \text{sinon} \end{cases}$$