

LIFPF – Programmation fonctionnelle

TD1 – introduction au λ -calcul

Licence informatique UCBL – Printemps 2025

Exercice 1 : Typage et évaluation en λ -calcul

- Pour chacune des λ -expressions suivantes, donner son type en supposant que le seul type primitif est **number** puis l'évaluer *en explicitant chaque β -réduction*. Si l'expression n'est pas typable, expliquer pourquoi. On supposera que l'addition notée $+$ a pour type **number** \rightarrow **number** \rightarrow **number** et se réduit par calcul arithmétique usuel, de même que pour la multiplication notée $*$. Par exemple, $(\lambda x. x + 3) 2$ a pour type **number** et peut se réduire en $(\lambda x. x + 3) 2 \xrightarrow{\beta} 2 + 3 \rightsquigarrow 5$
 - $((\lambda x. \lambda y. (x + y) \ 3) 4)$
 - $((\lambda x. \lambda y. (x \ y) \ 3) 4)$
 - $\lambda x. (x \ x)$
 - $((\lambda x. \lambda y. (y \ (x + 2)) \ 5) \ \lambda z. (z * 3))$
 - $(\lambda x. (\lambda y. (x + y) \ x) \ 5)$
 - $(\lambda x. ((x \ \lambda y. (y + 2)) + (x \ \lambda z. (z * 3)))) \ (\lambda u. \lambda w. (w \ u) \ 5))$
 - $(\lambda x. \lambda y. (x \ y) \ \lambda z. z)$
 - $(\lambda x. (x + 3) \ \lambda y. y)$
 - $\lambda x. \lambda y. x$
 - $(\lambda x. \lambda y. x \ 3)$
 - $((\lambda x. \lambda y. (y \ (3 + x)) \ 4) \ \lambda z. (z * 2))$
 - $(\lambda x. (\lambda y. (y * x) \ x) \ 5)$
- Donner deux séquences de réductions *différentes* de l'expression $(\lambda x. (x + 5) \ (\lambda z. z \ 2))$

Exercice 2 : Pour aller plus loin : codage des booléens dans le λ -calcul

Dans cet exercice on va s'intéresser au codage des booléens en λ -calcul pur appelé *booléens de Church*. Dans ce codage, les valeurs de vérités sont *des fonctions* et les fonctions booléennes des *fonctions d'ordre supérieur*. On définit les termes suivants qui représentent « vrai », « faux » et la fonction « et » :

- **T** = $\lambda x. \lambda y. x$
- **F** = $\lambda x. \lambda y. y$
- **AND** = $\lambda x. \lambda y. ((x \ y) \ \mathbf{F})$.

- Vérifier les réductions suivantes

$$\mathbf{AND}(\mathbf{T})(\mathbf{T}) \rightsquigarrow \mathbf{T}$$

$$\mathbf{AND}(\mathbf{T})(\mathbf{F}) \rightsquigarrow \mathbf{F}$$

$$\mathbf{AND}(\mathbf{F})(\mathbf{T}) \rightsquigarrow \mathbf{F}$$

$$\mathbf{AND}(\mathbf{F})(\mathbf{F}) \rightsquigarrow \mathbf{F}$$

2. Quelle est la fonction booléenne associée à **XXX** $= \lambda x. \lambda y. ((x\ y)\ x)$?
3. Qu'est ce qui se passe quand on évalue $((\mathbf{AND}\ x)\ y)$ où x ou y n'est *pas* la représentation d'un booléen ?
4. Donner une représentation de la fonction booléenne « non ».
5. Donner une représentation de la fonction booléenne « ou ».
6. Montrer que le terme **IF** $= \lambda x. x$ permet de représenter l'instruction conditionnelle **IF** *cond* **THEN** *M* **ELSE** *N*.