Introduction au lambda-calcul

version du 6 novembre 2002 – 12 h 13

Les fonctions comme citoyens de première classe

On peut faire que les preuves soient citoyens de première classe, mais pourquoi pas les fonctions ?

Quelques dates

autour de 1870 un Italien[a] s'oppose à Cantor sur le point de savoir quel est le concept de base des mathématiques prétendant que ça devrait être les fonctions.

1920 *Schönfinkel* initie la logique combinatoire,

1925 *Haskell Curry* crée la logique combinatoire,

1936 *Alonso Church* crée le λ-calcul,

1970-... Explosion du λ-calcul due à l'informatique (Barendregt, Berry, Boehm, de Bruijn, Girard, Hindley, Klop, Krivine, Levy, Plotkin, Scott, mais aussi Curien, Statmann etc.)

[a] dont j'ai oublié le nom.

Des notations différentes, un même concept

en maths	$x \mapsto x$	$f \mapsto (x \mapsto f(f(x)))$
en CAML	`fun x -> x`	`fun f -> (fun x -> (f (f x)))`
en λ-calcul	$\lambda x.x$	$\lambda f.(\lambda x.(f(fx)))$

La syntaxe

La classe Λ est la plus petite classe qui contient

1. x si x est une variable,
2. $\lambda x.M$ si $M \in \Lambda$,
3. (MN) si $M \in \Lambda$ et $N \in \Lambda$.

La syntaxe

La classe Λ est la plus petite classe qui contient

1. x si x est une variable,
2. $\lambda x.M$ si $M \in \Lambda$, abstraction
3. $(M N)$ si $M \in \Lambda$ et $N \in \Lambda$. application

Qu'y a-t-il derrière la syntaxe ?

On peut voir les termes comme des abstractions des fonctions ou des programmes fonctionnels.

Dans $\lambda x.M$, on dit que M est le corps de la fonction ou du programme.

Dans $(M N)$, on peut voir M comme une fonction que l'on applique au paramètre N. La valeur va s'obtenir par «réduction» (approche intentionnelle).

Le lambda-calcul décrit les fonctions par leur comportement.

L’anecdote derrière la syntaxe ?

Au début Church voulait écrire $\hat{\mathrm{x}}$.

Mais au temps des machines à écrire on ne savait écrire que $\hat{}\,\mathrm{x}$.

Ce qui a donné $\Lambda \mathrm{x}$, puis $\lambda \mathrm{x}$.

Exemples de termes

$I \equiv \lambda x..x$

$K \equiv \lambda x(\lambda y.x)$

$S \equiv \lambda x.(\lambda y.(\lambda z.((xz)(yz))))$

$B \equiv \lambda x.(\lambda y.(\lambda z.(x(yz))))$

Convention

1. Au lieu de $\lambda x_1(...(\lambda x_n.M)...)$

   on écrit $\lambda x_1...x_n.M$.

   Par exemple : $\lambda xy.x$.

2. Au lieu de $(...(MN_1)...N_p)$

   on écrit $MN_1...N_p$ ou $M\vec{N}$, si $\vec{N} = (N_1...N_p)$.

   Par exemple, $\lambda xyz.xz(yz)$

   à la place de $\lambda x.(\lambda y.(\lambda z.((xz)(yz))))$.

Convention

Par exemple : $((\lambda x.x)y)y)$ donne $(\lambda x.x)yy$,

«La fonction identité appliquée à y, puis le résultat est appliqué à y».

En revanche, $\lambda x.xyy$ correspond à $\lambda x.(xy)y$.

«La fonction qui à x fait correspondre le résultat de x appliqué à y puis à y».

Les mêmes termes avec conventions

$I \equiv \lambda x.x$

$K \equiv \lambda xy.x$

$S \equiv \lambda xyz.xz(yz)$

$B \equiv \lambda xyz.x(yz)$

Les mêmes termes avec conventions

$I \equiv \lambda x.x$	I est la fonction identité
$K \equiv \lambda xy.x$	Kc est la fonction constante c
$S \equiv \lambda xyz.xz(yz)$	$Sabc$ distribue c
$B \equiv \lambda xyz.x(yz)$	B permute l'effet des parenthèses

Variables et substitutions

Les variables liées

$$\begin{aligned} BV(x) &= \emptyset \\ BV(\lambda x.M) &= BV(M) \cup \{x\} \\ BV(MN) &= BV(M) \cup BV(N) \end{aligned}$$

Les variables liées

$$\begin{aligned} BV(x) &= \emptyset \\ BV(\lambda x.M) &= BV(M) \cup \{x\} \\ BV(MN) &= BV(M) \cup BV(N) \end{aligned}$$

$\boxed{\text{exemples}}$

$$\begin{aligned} BV(\lambda x.x) &= \{x\} \\ BV(\lambda fx.f(fx)) &= \{f, x\} \\ BV(\lambda fx.f(fxy)y) &= \{f, x\} \end{aligned}$$

Les variables libres 1/2

$$
\begin{aligned}
FV(x) &= \{x\} \\
FV(\lambda x.M) &= FV(M) - \{x\} \\
FV(M\,N) &= FV(M) \cup FV(N)
\end{aligned}
$$

Les variables libres 1/2

$$
\begin{aligned}
FV(x) &= \{x\} \\
FV(\lambda x.M) &= FV(M) - \{x\} \\
FV(MN) &= FV(M) \cup FV(N)
\end{aligned}
$$

exemples

$$
\begin{aligned}
FV(\lambda x.x) &= \emptyset \\
FV(\lambda fx.f(fx)) &= \emptyset \\
FV(\lambda fx.f(fxy)y) &= \{y\} \\
FV(\lambda x.f(fx)) &= \{f\}
\end{aligned}
$$

Les variables libres 2/2

Un terme qui n’a pas de variable libre est dit clos ou fermé ou est appelé un combinateur.

Les variables libres 2/2

Un terme qui n'a pas de variable libre est dit clos ou fermé ou est appelé un combinateur.

ATTENTION !

Une variable peut être à la fois libre et liée dans un terme.

Par exemple : $x(\lambda x.x)$.

Le produit cartésien et la curryfication

Il n'y a pas de produit cartésien dans le λ-calcul simple.

Si on veut écrire :

$$\lambda(x,y).f(x,y) \quad \text{ou} \quad (x,y) \mapsto f(x,y)$$

on le remplace par

$$\lambda xy.fxy$$

C'est la curryfication (nommée après Haskell Curry).

Substitution

Substituer une variable par un terme ne consiste pas simplement à remplacer toutes les occurrences de la variable par ce terme, à cause du phénomène de capture.

Quand on écrit $M[x := P]$ on ne remplace pas simplement les occurrences de x dans M par P.

Substitution

Ainsi

$$\begin{aligned} x(\lambda x.x)[x := y] &\neq y(\lambda x.y) \\ (\lambda y.x)[x := y] &\neq \lambda y.y \end{aligned}$$

donc il faut être prudent.

Substitution avec renommage

1. $x[x := P] = P$
2. $y[x := P] = y$
3. $(\lambda x.M)[x := P] = \lambda x.M$
4. $(\lambda y.M)[x := P] = \lambda y.(M[x := P])$

   si $x \notin FV(M)$ ou $y \notin FV(P)$
5. $(\lambda y.M)[x := P] = \lambda z.(M[y := z][x := P])$

   si $x \in FV(M)$ et $y \in FV(P)$

   et z est une nouvelle variable
6. $(M_1 M_2)[x := P] = M_1[x := P] M_2[x := P]$

La convention de Barendregt

C’est une convention sur les variables libres d’un terme dans un énoncé mathématique.

Il n’existe aucun sous-terme dans lequel une variable apparaît à la fois libre et liée.

L'α-conversion (règles structurelles)

$$\lambda x.N \equiv_\alpha \lambda y.(N[x := y]) \qquad y \notin FV(N) \qquad base$$

$$\frac{M_1 \equiv_\alpha N_1 \qquad M_2 \equiv_\alpha N_2}{M_1 M_2 \equiv_\alpha N_1 N_2}\,\alpha APP$$

$$\frac{M \equiv_\alpha N}{\lambda z.M \equiv_\alpha \lambda z.N}\,\alpha ABS$$

$$x \equiv_\alpha x \quad \alpha var$$

Réflexivité de l'α-conversion

Lemme : Pour tout $M \in \Lambda$, on a $M \equiv_\alpha M$.

Par induction structurelle sur M.

M **est la variable** x dans ce cas on applique l'axiome 'a Var.

M **est une application** $M_1 M_2$**.** Alors par induction on a $M_1 \equiv_\alpha M_1$ et $M_2 \equiv_\alpha M_2$ donc on peut appliquer la règle α *APP* pour obtenir $M_1 M_2 \equiv_\alpha M_1 M_2$.

M **est une abstraction *'l x . P*.** Par induction $P \equiv_\alpha P$. Donc par la règle α ABS on a $\lambda x.P \equiv_\alpha \lambda x.P$.

L'α-conversion (règles de congruence)

$$\frac{M \equiv_\alpha N}{N \equiv_\alpha N}\ \alpha\textit{symétrie}$$

$$\frac{M \equiv_\alpha N \qquad N \equiv_\alpha P}{M \equiv_\alpha P}\ \alpha\textit{transitivité}$$

L'α-conversion (règles de congruence)

$$\frac{M \equiv_\alpha N}{N \equiv_\alpha N}\ \alpha\textit{symétrie}$$

$$\frac{M \equiv_\alpha N \qquad N \equiv_\alpha P}{M \equiv_\alpha P}\ \alpha\textit{transitivité}$$

L'α-conversion est une relation d'équivalence, stable par passage au contexte, on dit que c'est une congruence.

L'α-conversion

L'α-conversion ne change pas la «signification» des termes.

La convention de Barendregt

- On suppose que dans tout théorème que l'on énonce, on suit la convention de Barendregt.
- Si l'on a un terme qui ne satisfait pas la convention de Barendregt, on s'y ramène par α-conversion

Substitution et convention de Barendregt

Avec la convention de Barendregt, la définition des substitutions devient beaucoup plus simple.

- $x[x := P] = P$
- $y[x := P] = y$
- $(\lambda y.M)[x := P] = \lambda y.M[x := P]$
- $(M_1 M_2)[x := P] = M_1[x := P]M_2[x := P]$

La β-réduction et les autres réductions

La β-contraction

Les fonctions sont faites pour calculer !

Les réductions d'un terme représentent son calcul.

La β-contraction en est l'étape élémentaire.

$$(\lambda x.M)P \xrightarrow[\beta]{} M[x := P]$$

R-réduction

On se donne un ensemble R de règles, c-à-d de paires de termes, par exemple β.

$M \xrightarrow[R]{} N$ signifie que M se réduit à N par R en une étape.

R-réduction

(contraction) $$\frac{(M, N) \in R}{M \xrightarrow[R]{} N}$$ (ξ) $$\frac{M \xrightarrow[R]{} N}{\lambda x M \xrightarrow[R]{} \lambda x N}$$

(congruence gauche) $$\frac{M \xrightarrow[R]{} N}{M P \xrightarrow[R]{} N P}$$ (congruence droite) $$\frac{M \xrightarrow[R]{} N}{P M \xrightarrow[R]{} P N}$$

Exercice

Réduire

- $\lambda y.(\lambda x.x)z$
- $(\lambda fx.f(fx))(\lambda x.x)$
- $(\lambda fx.f(fx))(\lambda fx.fx)$

D’autres exemples de réductions

La contraction η

Pour tout $M \in \Lambda$ et $x \notin FV(M)$,

$$\lambda x.Mx \quad \xrightarrow[\eta]{} \quad M.$$

L’expansion η

Pour tout $M \in \Lambda$ et $x \notin FV(M)$,

$$M \quad \xrightarrow[\eta_{exp}]{} \quad \lambda x.Mx.$$

D’autres exemples de réductions

La contraction par β et η

$$\xrightarrow[\beta\eta]{} \quad = \quad \xrightarrow[\beta]{} \cup \xrightarrow[\eta]{} .$$

On s’intéresse évidemment à la $\beta\eta$-réduction $\underset{\beta\eta}{\twoheadrightarrow}$ qui est la fermeture transitive et réflexive de $\xrightarrow[\beta\eta]{}$.

Fermeture transitive et réflexive

(cas de base) $\dfrac{M \xrightarrow[R]{} N}{M \xrightarrow[R]{}\!\!\!\!\to N}$ (réflexivité) $M \xrightarrow[R]{}\!\!\!\!\to M$

(transitivité) $\dfrac{M \xrightarrow[R]{}\!\!\!\!\to N \quad N \xrightarrow[R]{}\!\!\!\!\to L}{M \xrightarrow[R]{}\!\!\!\!\to L}$

Proposition Préservation par abstraction et application de $\xrightarrow[\beta]{}\!\!\!\!\!\twoheadrightarrow$

$$\frac{M \underset{\beta}{\twoheadrightarrow} N}{\lambda x.M \underset{\beta}{\twoheadrightarrow} \lambda x.N}$$

$$\frac{M \underset{\beta}{\twoheadrightarrow} N \qquad P \underset{\beta}{\twoheadrightarrow} Q}{M P \underset{\beta}{\twoheadrightarrow} N Q}$$

Proposition | Préservation par abstraction et application de $\underset{\beta}{\twoheadrightarrow}$

Cas $$\frac{M \underset{\beta}{\twoheadrightarrow} N}{\lambda x.M \underset{\beta}{\twoheadrightarrow} \lambda x.N}$$

On doit montrer que

– sous l'hypothèse $M \underset{\beta}{\twoheadrightarrow} N$

– on a la conclusion $\lambda x.M \underset{\beta}{\twoheadrightarrow} \lambda x.N$.

La démonstration est par induction sur la taille de l'arbre de preuve de $M \underset{\beta}{\twoheadrightarrow} N$ (qui utilise les règles de la définition de $\underset{\beta}{\twoheadrightarrow}$ et celle de $\underset{\beta}{\rightarrow}$).

Proposition $\boxed{\text{Préservation par abstraction et application de } \xrightarrow[\beta]{} \!\!\!\!\!\!\to}$

Trois cas se présentent :

1. $M \xrightarrow[\beta]{} N$, on a utilisé le «cas de base», alors par (ξ), $\lambda x.M \xrightarrow[\beta]{} \lambda x.N$ et on conclut par le «cas de base».
2. $M \equiv N$, alors $\lambda x.M \equiv \lambda x.N$ et conclut par «réflexivité».
3. Il existe P tel que $M \twoheadrightarrow_\beta P$ et $P \twoheadrightarrow_\beta N$. Par induction, on tire,
   - $\lambda x.M \twoheadrightarrow_\beta \lambda x.P$
   - et $\lambda x.P \twoheadrightarrow_\beta \lambda x.N$,

   et par «transitivité» $\lambda x.M \twoheadrightarrow_\beta \lambda x.N$.

Fermeture transitive, réflexive et symétrique

Avec les règles

(cas de base) $$\frac{M \xrightarrow[R]{} N}{M \overset{*}{\underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow}} N}$$ (réflexivité) $$M \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} M$$

(transitivité) $$\frac{M \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} N \quad N \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} L}{M \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} L}$$ (symétrie) $$\frac{M \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} N}{N \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} M}$$

on obtient la fermeture transitive, réflexive et symétrique

de $\xrightarrow[R]{}$.

Fermeture transitive, réflexive et symétrique

– Elle s'écrit $=_R$ ou $\underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow}$,

– elle se dit R-égal ou R-convertible ou R-équivalent .

Égalité extensionnelle

L'égalité extensionnelle

$\underset{\beta}{\twoheadleftarrow\!\!\!\twoheadrightarrow}$ décrit l'égalité intentionnelle.

Autrement dit, si $M \underset{\beta}{\twoheadleftarrow\!\!\!\twoheadrightarrow} N$ si M et N ont le même «comportement».

Deux termes sont extensionnellement équivalents s'ils prennent les mêmes «valeurs» quand on les appliquent au même terme.

L’égalité extensionnelle

On aussi la règle

$$\text{(ext)} \quad \frac{M x =_{\text{ext}} N x}{M =_{\text{ext}} N} \qquad x \notin FV(M N)$$

L’égalité extensionnelle

On aussi la règle

$$\text{(ext)} \quad \frac{Mx =_{\text{ext}} Nx}{M =_{\text{ext}} N} \qquad x \notin FV(MN)$$

Par les règles cas de base pour β + réflexivité + transitivité + symétrie + ext on définit une relation $=_{\text{ext}}$.

Proposition $\boxed{=_{\text{ext}} \quad = \quad \overset{}{\underset{\beta\eta}{\twoheadleftarrow\!\!\!\twoheadrightarrow}} \; .}$

Proposition $\boxed{=_{\text{ext}} \;=\; \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow}\,.}$

Pour montrer que $\underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow} \subseteq =_{\text{ext}}$

il suffit de montrer que si $P \xrightarrow[\eta]{} Q$ alors $P =_{\text{ext}} Q$

donc il suffit de montrer que pour $x \notin FV(M)$ $(\lambda x.Mx) =_{\text{ext}} M$

En fait, il suffit de montrer qu'on a $(\lambda x.Mx)x \underset{\beta}{\twoheadleftarrow\!\!\twoheadrightarrow} Mx$.

Par le *cas de base* et la définition de $\xrightarrow[\beta]{}$, cela vient de

$(\lambda x.Mx)x \xrightarrow[\beta]{} (Mx)[x := x] \equiv Mx$.

Proposition $\boxed{=_{\text{ext}} \;=\; \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow}\,.}$

Pour montrer que $=_{\text{ext}} \;\subseteq\; \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow}$,

il suffit de montrer que ext est une règle dérivée dans $\underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow}$.

Supposons $Mx \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow} Nx$ avec $x \notin FV(MN)$,

alors $\lambda x.Mx \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow} \lambda x.Nx$ par ξ,

donc par η appliquée deux fois, on a

$$M \xleftarrow[\eta]{} \lambda x.Mx \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow} \lambda x.Nx \xrightarrow[\eta]{} N$$

soit

$$M \underset{\beta\eta}{\twoheadleftarrow\!\!\twoheadrightarrow} N.$$

Quelques résultats de stabilité

Contexte

Un contexte $C[\,]$ est défini ainsi

1. $[\,]$ est un contexte,
2. si $M \in \Lambda$ et si $C[\,]$ est un contexte alors $MC[\,]$ et $C[\,]M$ sont des contextes,
3. si $C[\,]$ est un contexte alors $\lambda x.C[\,]$ est un contexte.

Contexte

Définition Si $C[\,]$ est un contexte et $A \in \Lambda$ alors $C[A]$ est défini par induction sur $C[\,]$.

- $[A] = A$,
- si $C[\,] = \lambda x.D[\,]$ alors $C[A] = \lambda x.D[A]$,
- si $C[\,] = MD[\,]$ alors $C[A] = MD[A]$,
- si $C[\,] = D[\,]M$ alors $C[A] = D[A]M$,

Stabilité par contexte

Proposition $\xrightarrow[R]{}$, $\overset{*}{\underset{R}{\twoheadrightarrow}}$ et $\underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow}$ sont stables par contexte.

$\underset{R}{\twoheadrightarrow}$ et $\underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow}$ sont stables par substitutions.

$\xrightarrow[R]{}$ est stable par contexte.

Démonstration : $M \xrightarrow[R]{} N$ alors $C[M] \xrightarrow[R]{} C[N]$

Par induction sur la structure de $C[\,]$, sachant que $M \xrightarrow[R]{} N$.

1. $C[\,] = [\,]$ alors $C[M] = M$ et $C[N] = N$, évident.
2. $C[\,] = AD[\,]$,
   - par induction $D[M] \xrightarrow[R]{} D[N]$,
   - d'autre part, $C[M] = AD[M]$ et $C[N] = AD[N]$,
   - donc par congruence à droite $C[M] \xrightarrow[R]{} C[N]$.
3. $C[\,] = D[\,]A$, comme 2 en changeant «droite» en «gauche».
4. $C[\,] = \lambda x.D[\,]A$, par induction $D[M] \xrightarrow[R]{} D[N]$, d'où la conclusion par (ξ).

Stabilité par substitution

Proposition

$M \xrightarrow[R]{} \!\!\!\!\!\rightarrow N$ implique $A[x := M] \xrightarrow[R]{} \!\!\!\!\!\rightarrow A[x := N]$.

Stabilité par substitution

Proposition

$M \xrightarrow[R]{} \!\!\!\!\!\!\rightarrow N$ implique $A[x := M] \xrightarrow[R]{} \!\!\!\!\!\!\rightarrow A[x := N]$.

Démonstration : L'hypothèse est $M \xrightarrow[R]{} \!\!\!\!\!\!\rightarrow N$.

La démonstration se fait par induction sur A.

- $A \equiv x$, alors $A[x := M] \equiv M \xrightarrow[R]{} \!\!\!\!\!\!\rightarrow N \equiv A[x := N]$.
- $A \equiv y$, alors $A[x := M] \equiv y \xrightarrow[R]{} \!\!\!\!\!\!\rightarrow y \equiv A[x := N]$ par réflexivité.

Stabilité par substitution (suite)

- $A \equiv \lambda y.B$,

  par induction $B[x := M] \underset{R}{\twoheadrightarrow} B[x := N]$,

  donc $A[x := M] \equiv \lambda y.B[x := M]$ et

  $A[x := N] \equiv \lambda y.B[x := N]$,

  par (ξ) pour $\underset{R}{\twoheadrightarrow}$, on a

  $\lambda y.B[x := M] \underset{R}{\twoheadrightarrow} \lambda y.B[x := N]$.

Stabilité par substitution (suite)

– $A \equiv B\,B'$

par induction

$$\begin{array}{ccc} B[x := M] & \xrightarrow[R]{}\!\!\!\!\!\rightarrow & B[x := N] \\ B'[x := M] & \xrightarrow[R]{}\!\!\!\!\!\rightarrow & B'[x := N] \end{array}$$

par définition de la substitution

$$\begin{array}{rcl} A[x := M] & \equiv & B[x := M]\,B'[x := M] \\ A[x := N] & \equiv & B[x := N]\,B'[x := N] \end{array}$$

Stabilité par substitution (suite)

par congruence et transitivité de $\xrightarrow[R]{}\!\!\!\!\twoheadrightarrow$ on a

$$\dfrac{\dfrac{B[x := M] \twoheadrightarrow B[x := N]}{B[x := M]B'[x := M] \twoheadrightarrow B[x := N]B'[x := M]} \qquad \dfrac{B'[x := M] \twoheadrightarrow B'[x := N]}{B[x := N]B'[x := M] \twoheadrightarrow B[x := N]B'[x := N]}}{B[x := M]B'[x := M] \twoheadrightarrow B[x := N]B'[x := N]}$$

Exercice

Montrez que ça ne peut pas marcher pour $\xrightarrow[R]{}$,

c-à-d qu'on n'a pas :

$M \xrightarrow[R]{} N$ implique $A[x := M] \xrightarrow[R]{} A[x := N]$.

Redex et formes normales

Quelques définitions

- Un R-redex est un terme M tel que $(M, N) \in R$.
- N et le R-contracté de M.
- Un terme M est R-irréductible si M ne contient aucun R-redex.
- Un terme N est une forme normale de M, si N est R-irréductible et si $M \underset{R}{\twoheadleftarrow\!\!\twoheadrightarrow} N$.

Formes normales

On n'affirme

- ni l'existence (cf Ω),
- ni l'unicité, il y a unicité pour β et $\beta \cup \eta_{ext}$, mais il faut le prouver.

  La forme normale si elle existe est la valeur intentionnelle.

Exercice

Lesquels de ces termes sont des formes normales ?

$(\lambda x.x)$

$((\lambda xy.x)v)w$

$(\lambda xy.xv)w$

$\lambda xy.xvw$

Des termes

Ω et les autres

$$\omega \equiv \lambda x.xx$$

$$\Omega \equiv (\lambda x.xx)(\lambda x.xx)$$

$$\mathsf{Y} \equiv \lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$$

$$\mathsf{W}_F \equiv (\lambda x.F(xx))$$

Exercice

Montrez que Ω se réécrit vers un unique terme. Lequel ?

Plus précisément, montrez qu'il existe un terme unique $M \in \Lambda$ tel que $\Omega \xrightarrow[\beta]{} M$.

Exercice

1. Montrez que $\mathsf{Y}F \underset{\beta}{\twoheadrightarrow} F(\mathsf{W}_F\mathsf{W}_F)$.
2. Montrez que $F(\mathsf{Y}F) \underset{\beta}{\twoheadrightarrow} F(\mathsf{W}_F\mathsf{W}_F)$.
3. Conclure que $\mathsf{Y}F \underset{\beta}{\twoheadleftarrow\!\!\twoheadrightarrow} F(\mathsf{Y}F)$.

Les entiers de Church

Entiers de Church

- $(\lambda f x . f (f x)) \equiv$ **2** correspond au nombre entier deux.
- $(\lambda f x . x) \equiv T$ correspond à zéro.
- $(\lambda f x . f x) \equiv$ **1** correspond à un.
- Plus généralement l'entier **n** est le terme $(\lambda f x . f^n x)$.

Exercice

1. Montrez que

$$\mathbf{1} \xrightarrow[\eta]{} I \equiv \lambda x.x.$$

2. Écrivez l'opération «successeur».
3. Écrivez les opérations d'«addition» et de «multiplication».
4. Calculez
   - **12**,
   - **21**,
   - **22**,
5. A quoi correspond **mn** ?

Corrigé

En effet,

$$
\begin{array}{rcl}
\mathbf{1} & \equiv & \lambda f x . f x \\
& \xrightarrow[\eta]{} & \lambda f . f \quad \equiv \quad I .
\end{array}
$$

Corrigé

Le successeur est **succ** $\equiv \lambda nfx.n\ f\ (f\ x)$,

l'addition est **add** $\equiv \lambda mnfx.\ m\ f\ (n\ f\ x)$,

tandis que la multiplication est **mult** $\equiv \lambda mnf.\ m\ (n\ f)$.

Corrigé

$$
\begin{array}{lcl}
\mathbf{2}^{\mathbf{2}} & \equiv & (\lambda fx.f(f\ x))\ (\lambda gy.g(g\ y)) \\
 & \longrightarrow & \lambda x.(\lambda gy.g(g\ y))\ ((\lambda hz.h(h\ z))\ x) \\
 & \longrightarrow & \lambda x.(\lambda gy.g(g\ y))\ (\lambda z.x(x\ z)) \\
 & \longrightarrow & \lambda xy.(\lambda z.x(x\ z))\ ((\lambda w.x(x\ w))y) \\
 & \longrightarrow & \lambda xy.(\lambda z.x(x\ z))\ (x\ (x\ y)) \\
 & \longrightarrow & \lambda xy.x(x\ (x\ (x\ y))) \\
 & \equiv & \lambda fx.f(f\ (f\ (f\ x))) \quad \equiv \quad \mathbf{4}
\end{array}
$$

Corrigé

Appelons $\mathbf{p} \equiv \lambda mn.m\ n$ cette opération.

On remarque que

$$\begin{array}{lcl} \mathbf{p}\ \mathbf{0}\ \mathbf{n} & \xrightarrow[\beta]{} & (\lambda fx.x)\ \mathbf{n} \\ & \xrightarrow[\beta]{} & \lambda x.x \\ & \xleftarrow[\eta]{} & \mathbf{1} \end{array}$$

Corrigé

et que

$\mathbf{p}\ (\mathbf{succ\ m})\ \mathbf{n}\ x$	$\xrightarrow[\beta]{}$	$\mathbf{succ\ m\ n}\ x$	Défintion de **p**
	$\xrightarrow[\beta]{}$	$(\mathbf{m\ n})\ (\mathbf{n}\ x)$	Défintion de **succ**
	$\xleftarrow[\beta]{}$	$\mathbf{mult}\ (\mathbf{m\ n})\ x$	Défintion de **mult**
	$\xleftarrow[\beta]{}$	$\mathbf{mult}\ (\mathbf{p\ m\ n})\ \mathbf{n}\ x$	Défintion de **p**.

Corrigé

On a donc

$$\begin{aligned} \mathbf{p\ 0\ n} &=_{\beta\eta} \mathbf{1} \\ \mathbf{p}\ \big(\mathbf{succ\ m}\big)\ \mathbf{n} &=_{\beta\eta} \mathbf{mult}\ \big(\mathbf{p\ m\ n}\big)\ \mathbf{n} \end{aligned}$$

Corrigé

On a donc

$$\begin{aligned}\textbf{p 0 n} &=_{\beta\eta} \textbf{1}\\ \textbf{p}\ (\textbf{succ m})\ \textbf{n} &=_{\beta\eta} \textbf{mult}\ (\textbf{p m n})\ \textbf{n}\end{aligned}$$

Or on a

$$\begin{aligned}n^0 &= 1\\ n^{m+1} &= n^m \cdot n.\end{aligned}$$

p est un bon candidat pour représenter l'exponentielle.

Il y a simplement un problème : on applique un entier à un entier.

Lambda calcul et cohérence

Le lambda calcul est maximalement cohérent

Supposons que l'on ajoute au lambda calcul l'identité $\mathbf{S} = \mathbf{K}$.

Notons que d'une part

$$\begin{array}{rcl} \mathbf{S}\,\mathbf{I}\,(\mathbf{K}\,P)\,\mathbf{I} & \xrightarrow[\beta]{} & \mathbf{I}\,(\mathbf{K}\,P)\,\mathbf{I} \\ & \xrightarrow[\beta]{} & \mathbf{K}\,P\,\mathbf{I} \\ & \xrightarrow[\beta]{} & P \end{array}$$

Le lambda calcul est maximalement cohérent

et d'autre part

$$\begin{array}{lll} \mathbf{K}\,\mathbf{I}\,(\mathbf{K}\,P)\,\mathbf{I} & \xrightarrow[\beta]{} & \mathbf{I}\,\mathbf{I} \\ & \xrightarrow[\beta]{} & \mathbf{I} \end{array}$$

Donc de $\mathbf{S} = \mathbf{K}$ on déduit que pour tout terme P on a $P = \mathbf{I}$.

Le lambda calcul est maximalement cohérent

Autrement dit si on ajoute au lambda calcul l'unique égalité $P = \mathbf{I}$ on le rend incohérent en rendant tous les termes égaux.