λ-calcul

Éléments d'informatique pour les technologies quantiques

Introduction

Langage

- Symboles

Langage

- Symboles (A,B,C,D....)

Langage

- Symboles (A,B,C,D....)
- Mots

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, LKJDSLKSDZ, UUU...)

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, LKJDSLKSDZ, UUU...)
- Syntaxe

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, LKJDSLKSDZ, UUU...)
- Syntaxe
- Mots corrects

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, ~~LKJDSLKSDZ,~~ UUU...)
- Syntaxe
- Mots corrects

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, ~~LKJDSLKSDZ,~~ UUU...)
- Syntaxe
- Mots corrects
- Sens des mots

Langage

- Symboles (A,B,C,D....)
- Mots (BABA, MAISON, ~~LKJDSLKSDZ,~~ UUU...)
- Syntaxe
- Mots corrects
- Sens des mots (Sémantique)

Langage

- Symboles (1,2,3,+,x,- ....)
- Mots (2+1, 2, ~~3---+x-1~~, 1+2+-3...)
- Syntaxe
- Mots corrects
- Sens des mots (Sémantique)

Définition des termes

terme *(s, t, u, w...)* =

- Un entier *(1,2,3...)*

Exemple : 1 est un terme
286 est un terme

Définition des termes

terme *(s, t, u, w...)* =

- Un entier *(1,2,3...)*
- Une addition *(u + v)*

Exemple : 1+2 est un terme
1+(1+5) est un terme

Définition des termes

terme *(s, t, u, w...)* =

- Un entier *(1,2,3...)*
- Une addition *(u + v)*
- Un moins *(- u)*

Exemple : -1 est un terme
-(2+(-6)) est un terme

Définition des termes

terme *(s, t, u, w…)* =

- Un entier *(1,2,3…)*
- Une addition *(u + v)*
- Un moins *(- u)*

Exemple : -1 est un terme
-(2+(-6)) est un terme
(qu'on notera -(2-6))

Sens des termes (valeur)

Exemple : ((3+2) - 1) a pour valeur 4

Sens des termes (valeur)

Exemple : ((3+2) - 1) a pour valeur 4

-(-(3+1)) a aussi pour valeur 4

Sens des termes (valeur)

Exemple : ((3+2) - 1) a pour valeur 4
-(-(3+1)) a aussi pour valeur 4
1+(1+(1+1)) vaut aussi 4

Sens des termes (valeur)

Exemple : $((3+2) - 1)$ a pour valeur 4
$-(-(3+1))$ a aussi pour valeur 4
$1+(1+(1+1))$ vaut aussi 4

CE SONT DES MOTS DIFFÉRENTS !

Sens des termes (valeur)

Exemple : ((3+2) - 1) → (5 – 1) → 4

-(-(3+1)) → -(-4) → 4

1+(1+(1+1)) → 1+(1+2) → 1+3 → 4

On trouve la valeur du mot par réécriture

Réécriture des termes

Pour tous entiers x,y :

1 • $(x + y) \to x+y$

Exemple : $5 + 1 \to 6$

Réécriture des termes

Pour tous entiers x,y :

1. (x + y) → x+y
2. -(-(x)) → x

Exemple : -(-2) → 2

Valeurs des termes

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Exemple :

- (1 + 2) + - (- 3)

Valeurs des termes

Exemple :

- (1 + 2) + - (- 3)
- 3 + - (-3)

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Valeurs des termes

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Exemple :

- (1 + 2) + - (- 3)
- 3 + - (-3)
- 3 + 3

Valeurs des termes

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Exemple :

- (1 + 2) + - (- 3)
- 3 + - (-3)
- 3 + 3
- 6

Valeurs des termes

Exemple :

- 1 + (– 2)

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Valeurs des termes

Exemple :

- 1 + (– 2)

Ne se réécrit pas !

Pour tous entiers x,y :

- (x + y) → x+y
- -(-(x)) → x

Cas des fonctions

Exemple :

- $(x \mapsto x + 3)\ 4$

Cas des fonctions

Exemple :

- $(x \mapsto x + 3)\ 4 \quad \to \quad 7$

Cas des fonctions

Cas général :

- $(x \mapsto f(x))$

Cas des fonctions

Cas général :

- $(x \mapsto f(x))\ y$

Cas des fonctions

Cas général :

- $(x \mapsto f(x))\ y \rightarrow f(y)$

Idée cruciale

Petit langage de fonctions

Idée cruciale

Petit langage de fonctions :

$$\lambda x.\ f(x) \quad \leftrightarrow \quad (x \mapsto f(x))$$

Historique

- Inventé par Alonso Church en 1930

Historique

- Inventé par Alonso Church en 1930
- Fonde les mathematiques ?

Historique

- Inventé par Alonso Church en 1930
- ~~Fonde les mathématiques~~
- Moins puissant que la théorie des ensembles

Historique

- Inventé par Alonso Church en 1930
- ~~Fonde les mathématiques~~
- Moins puissant que la théorie des ensembles
- Turing-complet

Définition

λ-calcul

Définition des λ-termes

λ-terme (*s*, *t*, *u*, *w*...) =

- Une variable (x, y, z ...)

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (*x*, *y*, *z* …)

Exemple : x est un λ-terme

y est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (*x*, *y*, *z* …)
- Une application *uv*

Définition des λ-termes

λ-terme *(s, t, u, w...)* =

- Une variable (*x*, *y*, *z* ...)
- Une application *uv*

u est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w...)* =

- Une variable (*x*, *y*, *z* ...)
- Une application *uv*

u est un λ-terme

v est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (*x*, *y*, *z* …)
- Une application *uv*

Exemple : xy est un λ-terme

y(xx) est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (x, y, z …)
- Une application uv
- Une abstraction $\lambda x \,.\, u$

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (x, y, z …)
- Une application uv
- Une abstraction $\lambda x \,.\, u$

x est une variable

Définition des λ-termes

λ-terme *(s, t, u, w...)* =

- Une variable (*x*, *y*, *z* ...)
- Une application *uv*
- Une abstraction $\lambda x \, . \, u$

x est une variable

u est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w...)* =

- Une variable (x, y, z ...)
- Une application uv
- Une abstraction $\lambda x . u$

Exemple : $\lambda x.x$ est un λ-terme

$\lambda y.\, yx$ est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (x, y, z …)
- Une application uv
- Une abstraction $\lambda x \,.\, u$

Exemple : δ = λx. (xx) est un λ-terme

Ω = δδ est un λ-terme

Définition des λ-termes

λ-terme *(s, t, u, w…)* =

- Une variable (x, y, z …)
- Une application uv
- Une abstraction $\lambda x \,.\, u$

Exemple : $\delta = \lambda x.\ xx$ est un λ-terme

$\Omega = \delta\delta$ est un λ-terme

Définition des λ-termes

Remarques :

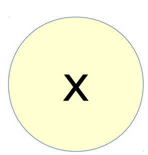
- Un λ-terme est un arbre

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre

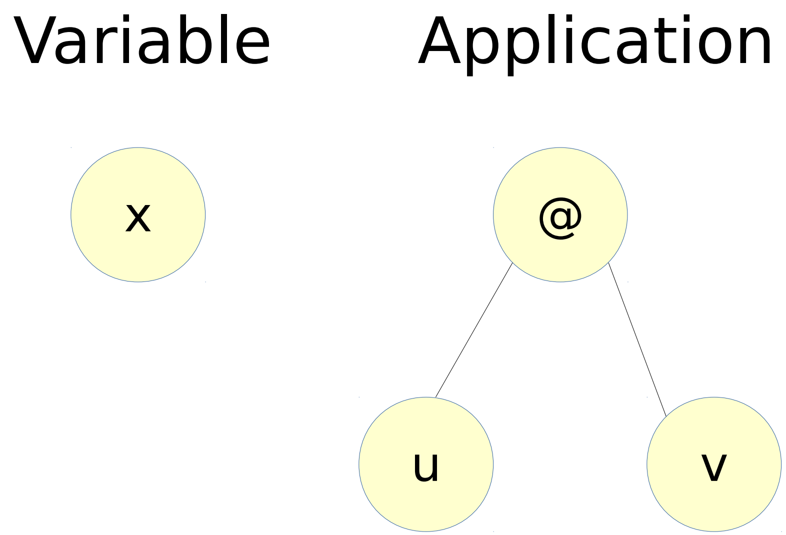
Variable



Définition des λ-termes

Remarques :

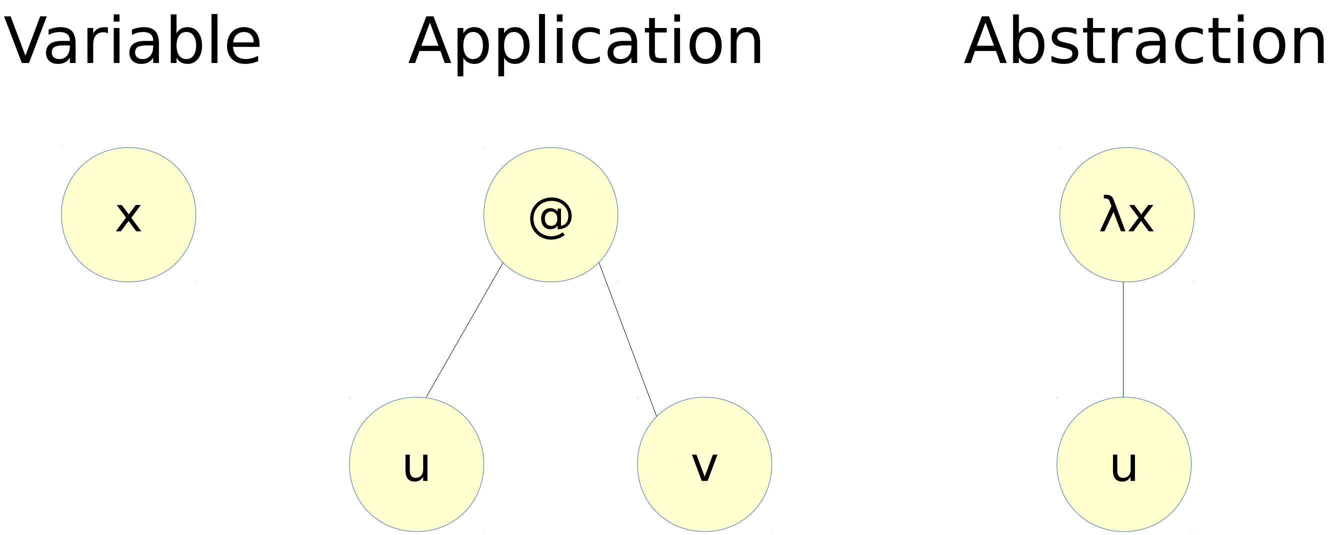
- Un λ-terme est un arbre



Définition des λ-termes

Remarques :

- Un λ-terme est un arbre



Définition des λ-termes

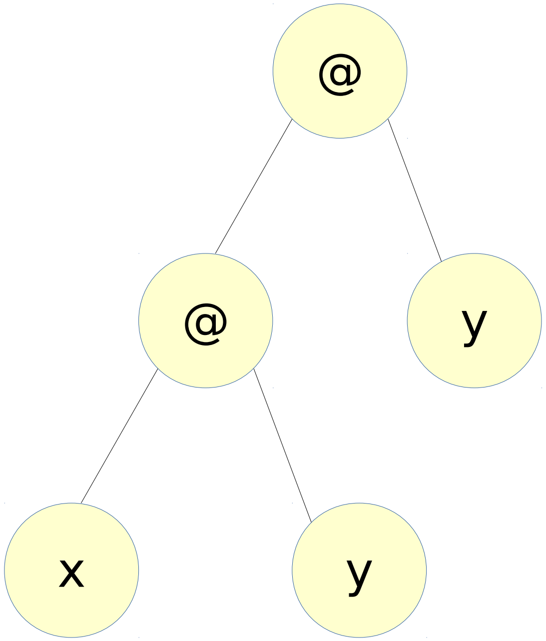
Remarques :

- Un λ-terme est un arbre
- Curryfication

Définition des λ-termes

Remarques :

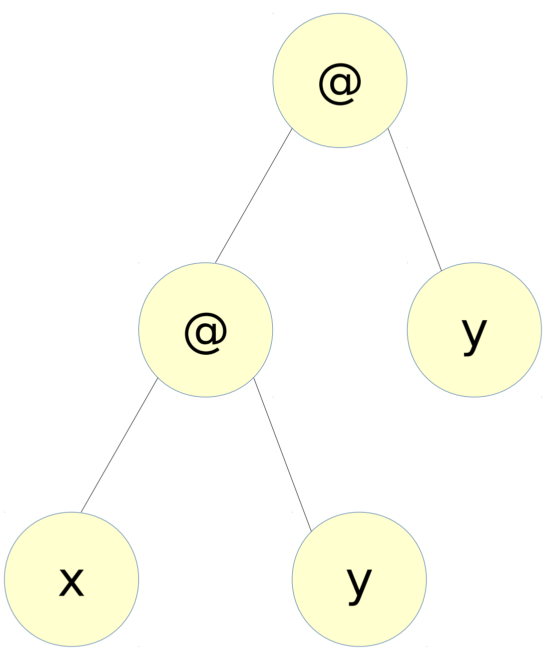
- Un λ-terme est un arbre
- Curryfication



Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication

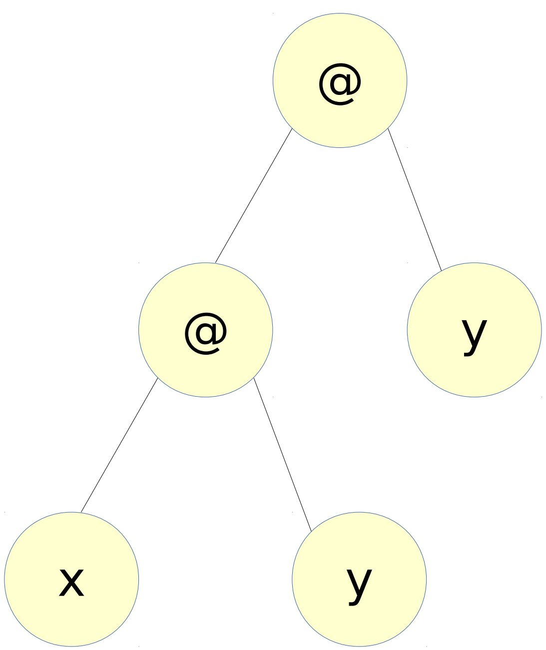


s'écrit: xyy

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication



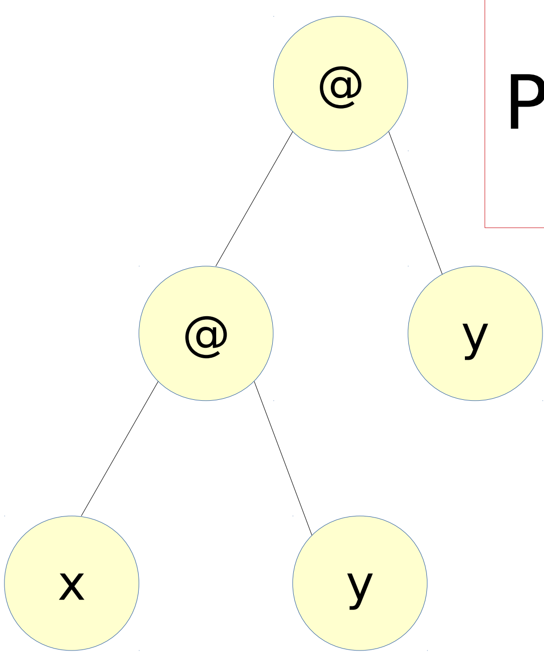
s’écrit: *xyy* et signifie (xy)y

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication

PENSER À LA CURRYFICATION EN CAML !



s'écrit: *xyy* et signifie (xy)y

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication
- Abstraction = fonction

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication
- Abstraction = fonction

$\lambda x.u$ = “ la fonction qui à x associe u”

Définition des λ-termes

Remarques :

- Un λ-terme est un arbre
- Curryfication
- Abstraction = fonction

“λx. λy. u” est souvent noté “λxy. u”

Réécritures

λ-calcul

Sémantique

Syntaxe = symboles autorisés

Sémantique

Syntaxe = symboles autorisés

Sémantique = sens des symboles

Variables libres et variables liées

$$\lambda x.\ x$$

$$\lambda y.\ y$$

Variables libres et variables liées

$\lambda x.\ x$

$\lambda y.\ y$

l'identité (id : $x \mapsto x$)

Variables libres et variables liées

$$\lambda x.\, ex$$

$$\neq$$

$$\lambda x.\, fx$$

Variables libres et variables liées

$$\lambda x.\ ex \neq \lambda x.\ fx$$

e et f sont des variables libres

Variables libres et variables liées

Exemple:

$$\text{terme} := y(\lambda x.\lambda y.xz)$$

Variables libres et variables liées

Exemple:

$$\text{terme} := y(\lambda x.\lambda y.xz)$$

- FV(terme) = {y,z} *(variables libres)*

Variables libres et variables liées

Exemple:

$$\text{terme} := y(\lambda x.\lambda y.xz)$$

- FV(terme) = {y,z} *(variables libres)*
- BV(terme) = {y,x} *(variables liées)*

Sémantique . α-équivalence

Idée :

- On peut renommer les variables liées sans problème

$$\lambda x.u \sim \lambda y.u[x:=y]$$

Sémantique . α-équivalence

Définition : Substitution partielle

- Soient u,v des λ-termes et x une variable

Sémantique . α-équivalence

Définition : Substitution partielle

- Soient u,v des λ-termes et x une variable tels que:
- $x \notin BV(u)$

$$u = y(\lambda x.\lambda y.xz)$$

Sémantique . α-équivalence

Définition : Substitution partielle

- Soient u,v des λ-termes et x une variable tels que:
- $x \notin BV(u)$
- $FV(v) \cap BV(u) = \emptyset$

$$u = y(\lambda x.\lambda y.xz)$$

Sémantique . α-équivalence

Définition : Substitution partielle

- Soient u,v des λ-termes et x une variable tels que:
- $x \notin BV(u)$
- $FV(v) \cap BV(u) = \emptyset$
- Alors on définit $u[x := v]$

  (on remplace x par v dans u)

Sémantique . α-équivalence

Idée :

- On peut renommer les variables liées sans problème

$$\lambda x.u \sim \lambda y.u[x:=y]$$

- On formalise donc ce renommage

Sémantique . α-équivalence

Définition : α-équivalence

$=_\alpha$ est la plus petite congruence contenant α, qui est la relation :

$$\lambda x.\ u \quad \alpha \quad \lambda y.\ u[x := y]$$

Sémantique . α-équivalence

Définition : α-équivalence

$=_\alpha$ est la plus petite congruence contenant α, qui est la relation :

$$\lambda x.\ u \quad \alpha \quad \lambda y.\ u[x := y]$$

- où y=x ou bien ...

Sémantique . α-équivalence

Définition : α-équivalence

$=_\alpha$ est la plus petite congruence contenant α, qui est la relation :

$$\lambda x.\, u \;\;\alpha\;\; \lambda y.\, u[x := y]$$

- où y=x ou bien $\begin{cases} x \notin BV(u) \\ y \notin BV(u) \cup FV(u) \end{cases}$

Sémantique . α-équivalence

Définition : α-équivalence

$=_\alpha$ est la plus petite congruence contenant α, qui est la relation :

$$\lambda x.\ u \quad \alpha \quad \lambda y.\ u[x := y]$$

- où $y = x$ ou bien $\begin{cases} \boxed{x \notin BV(u)} \\ y \notin BV(u) \cup FV(u) \end{cases}$

Sémantique

Syntaxe = symboles autorisés

Sémantique = sens des symboles

Sémantique

Syntaxe = symboles autorisés

Sémantique = sens des symboles

$\lambda x.xz \sim \lambda y.yz$

Sémantique

Syntaxe = symboles autorisés

Sémantique = sens des symboles

$$\lambda x.xz \sim \lambda y.yz$$

$$(\lambda x.xz)u \sim uz$$

Sémantique

Syntaxe = symboles autorisés

Sémantique = sens des symboles

$$\lambda x.xz \sim \lambda y.yz \qquad (\alpha)$$

$$(\lambda x.xz)u \sim uz \qquad (\beta)$$

Sémantique . β-réduction

Définition : β-réduction

$\boxed{\text{Idée : étape de calcul}}$

$$(\lambda x.xz)u \rightarrow uz$$

Sémantique . β-réduction

Définition : β-réduction

$$(\lambda x.\ u)v \rightarrow u[x{:=}v]$$

Sémantique . β-réduction

Définition : β-réduction

$$(\lambda x.\ u)v \to u[x{:=}v]$$

(le calcul d'une fonction)

Sémantique . β-réduction

Définition : β-réduction

$$(\lambda x.\ u)v \rightarrow u[x:=v]$$

(le calcul d'une fonction)

$(x \mapsto u)v \ \rightarrow\ u[x \text{ remplacés par } v]$

Sémantique . β-réduction

Définition : β-réduction

$$(\lambda x.\ u)v \to u[x{:=}v]$$

(le calcul d'une fonction)

$(x \mapsto x\ x)v \ \to\ v\ v$

Sémantique . β-réduction

Définition : β-réduction

$$\underbrace{(\lambda x.\ u)v}_{\text{un redex}} \rightarrow u[x{:=}v]$$

Sémantique . β-réduction

Définition : β-réduction

$$\underbrace{(\lambda x.\ u)v}_{\text{un redex}} \rightarrow \underbrace{u[x:=v]}_{\text{le contractum associé}}$$

Sémantique . β-réduction

Définition : β-réduction

$$\underbrace{(\lambda x.\ u)v}_{\text{un redex}} \rightarrow \underbrace{u[x{:=}v]}_{\text{(le réduit)}}$$

Sémantique . β-réduction

Notations : β-réduction

$$U \to V$$

Sémantique . β-réduction

Notations : β-réduction

$$u \to v$$

$$u \to^* v$$

Sémantique . β-réduction

Notations : β-réduction

$$u \rightarrow v$$

$$u \rightarrow^{*} v$$

$$u \rightarrow u_1 \rightarrow \ldots \rightarrow u_n = v$$

Sémantique . β-réduction

Notations : β-réduction

$$u \to v$$

$$u \to^* v$$

$$u \not\to$$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\ xx$, $i = \lambda x.\ x$

- $\delta\ i \rightarrow$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\, xx\,,\ \ i = \lambda x.\, x$

- $\delta\, i \to ii \to i \not\to$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\, xx$, $i = \lambda x.\, x$

- $\delta\, i \to ii \to i \not\to$
- $x\, \delta\, i$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\, xx\ ,\ \ i = \lambda x.\, x$

- $\delta\, i \rightarrow ii \rightarrow i \not\rightarrow$
- $x\, \delta\, i \not\rightarrow$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\, xx$, $i = \lambda x.\, x$

- $\delta\, i \rightarrow ii \rightarrow i \not\rightarrow$
- $x\, \delta\, i \not\rightarrow$
- $i\, y\, (\lambda x.\lambda y.\, xz) \rightarrow y\, (\lambda x.\lambda y.\, xz) \not\rightarrow$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\ xx\ ,\ \ i = \lambda x.\ x$

- $\delta\ i \rightarrow ii \rightarrow i \not\rightarrow$
- $x\ \delta\ i \not\rightarrow$
- $i\ y\ (\lambda x.\lambda y.\ xz) \rightarrow\ y\ (\lambda x.\lambda y.\ xz) \not\rightarrow$

$\Omega = \delta\delta$

Sémantique . β-réduction

Exemples:

$\delta = \lambda x.\ xx\ ,\ \ i = \lambda x.\ x$

- $\delta\ i \rightarrow ii \rightarrow i \not\rightarrow$
- $x\ \delta\ i \not\rightarrow$
- $i\ y\ (\lambda x.\lambda y.\ xz) \rightarrow\ y\ (\lambda x.\lambda y.\ xz) \not\rightarrow$

$\Omega = \delta\delta$

- $\Omega \rightarrow \Omega \rightarrow \Omega \rightarrow \Omega \rightarrow \Omega \rightarrow \Omega \rightarrow \ldots$

Sémantique . β-réduction

Plusieurs choix possibles:

$$(\lambda x\,.\,(\lambda y\,.y)x)((\lambda z.zz)a)$$

Sémantique . β-réduction

Plusieurs choix possibles:

$$(\lambda x . (\lambda y . y)x)((\lambda z.zz)a)$$

$$(\lambda y . y)((\lambda z.zz)a)$$

Sémantique . β-réduction

Plusieurs choix possibles:

$$(\lambda x . (\lambda y . y)x)((\lambda z.zz)a)$$

$$(\lambda y . y)((\lambda z.zz)a)$$

Sémantique . β-réduction

Plusieurs choix possibles:

$$(\lambda x\,.\,(\lambda y\,.y)x)((\lambda z.zz)a)$$

$$(\lambda y\,.y)((\lambda z.zz)a)$$

$$(\lambda x\,.\,(\lambda y\,.y)x)aa$$

$$(\lambda x.x)((\lambda z.zz)a)$$

Sémantique . β-réduction

Plusieurs choix possibles:

$$(\lambda x\,.\,(\lambda y\,.y)x)((\lambda z.zz)a)$$

$$(\lambda y\,.y)((\lambda z.zz)a) \qquad\qquad (\lambda x\,.\,(\lambda y\,.y)x)aa$$

$$(\lambda x.x)((\lambda z.zz)a)$$

$$((\lambda z.zz)a)$$

$$aa$$

Sémantique . β-réduction

Plusieurs choix possibles:

$(\lambda x\ .\ (\lambda y\ .y)x)((\lambda z.zz)a)$

$(\lambda y\ .y)((\lambda z.zz)a)$

$(\lambda x\ .\ (\lambda y\ .y)x)aa$

$(\lambda x.x)((\lambda z.zz)a)$

$((\lambda z.zz)a)$

aa

λ-calcul

Récapitulons !

λ-calcul

- Simple (3 règles)

λ-calcul

- Simple (3 règles)
- Fonctions (α, β ...)

λ-calcul

- Simple (3 règles)
- Fonctions (α, β ...)
- Déterministe ?

Valeurs

λ-calcul

Sémantique . Normalisation

Définition :

Un terme u est en forme normale ssi:

- $u \not\rightarrow$

Sémantique . Normalisation

Définition :

Un terme u est en forme normale ssi:

- $u \not\to$

C'est à dire ssi :

- u ne contient pas de redex

Sémantique . Normalisation

Définition :

Un terme u est en forme normale ssi:

- u $\not\to$

C'est à dire ssi :

- u ne contient pas de redex

Exemple : y (λx.λy. xz) est en forme normale

x est en forme normale

Sémantique . Normalisation

Définition :

Un terme u est en forme normale ssi:

- $u \not\rightarrow$

C'est à dire ssi :

- u ne contient pas de redex

Idée : forme normale = valeur

Sémantique . Normalisation

Définition :

Un terme v est une forme normale de u ssi:

- $u \rightarrow^* v \not\rightarrow$

Sémantique . Normalisation

Définition :

Un terme v est une forme normale de u ssi:

- $u \rightarrow^{*} v \not\rightarrow$

Idée : v est une valeur possible de u

Sémantique . Normalisation

Définition :

Un terme v est une forme normale de u ssi:

- $u \rightarrow^* v \not\rightarrow$

Définition :

Un terme u est normalisable si

- $\exists v \mid u \rightarrow^* v \rightarrow$
- *i.e.* u a une forme normale

Sémantique . Terminaison

Définition :

u faiblement terminant

=

u normalisable

Sémantique . Terminaison

Définition :

u faiblement terminant

=

u normalisable

Définition :

u terminant

=

toutes ses β-réductions terminent

Sémantique . Terminaison

Définition :

u faiblement terminant

=

u normalisable

Définition :

u terminant

=

u fortement normalisable

Sémantique . Terminaison

Exemple :

u faiblement terminant

mais u non terminant

Sémantique . Terminaison

Exemple :

u faiblement terminant

mais u non terminant

$$u = (\lambda x.y)\Omega$$

Sémantique . Terminaison

Exemple :

u faiblement terminant

mais u non terminant

$$u = (\lambda x.y)\Omega$$

$$y$$

Sémantique . Terminaison

Exemple :

u faiblement terminant

mais u non terminant

$$u = (\lambda x.y)(\delta\delta)$$

$$y$$

Sémantique . Terminaison

Exemple :

u terminant

$$u = (\lambda x.y)\delta\delta$$

Sémantique . Terminaison

Exemple :

u terminant

$$u = (\lambda x.y)\delta\delta$$

$$y\delta$$

Sémantique . Confluence

Terminaison :

λ-calcul ne termine pas pour la β-réduction

Sémantique . Confluence

Terminaison :

λ-calcul = définir et caractériser les fonctions récursives

Sémantique . Confluence

Terminaison :

λ-calcul = définir et caractériser les fonctions récursives

```
def fact(x):
    if x == 0 : return 1
    else: return x * fact(x - 1)
```

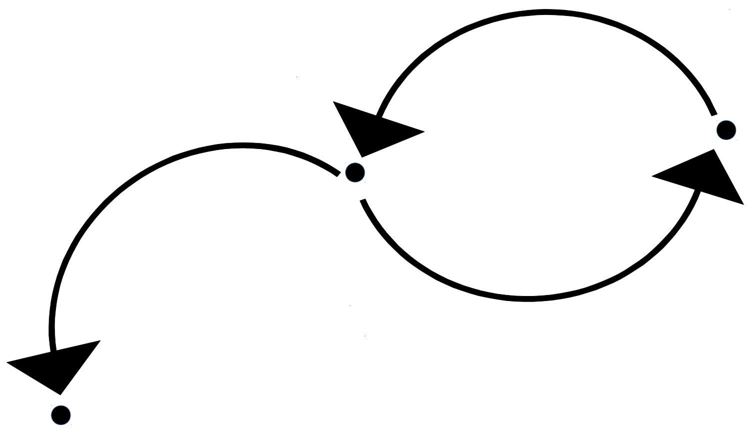
Sémantique . Confluence

λ-calcul ne termine pas pour la β-réduction

Résultat unique ?

Sémantique . Confluence

Système de transitions $(A, \rightarrow)$



Sémantique . Confluence

Système de transitions $(A, \rightarrow)$

Exemples:

- $(\lambda, \rightarrow)$
- $(\mathbb{N}, <)$
- $(\mathbb{N}, >)$
- $(\mathbb{N}^*, |)$
- $(\mathbb{N}, \rightarrow)$

Sémantique . Confluence

Définition :

(A,→) a la propriété de forme normale unique

=

∀**a**,**b**,**c** ∈ A, si

$$\begin{tikzcd} & \mathbf{a} \arrow[dl, "*"'] \arrow[dr, "*"] & \\ \mathbf{b} \arrow[d, "/" marking] & & \mathbf{c} \arrow[d, "/" marking] \\ {} & & {} \end{tikzcd}$$

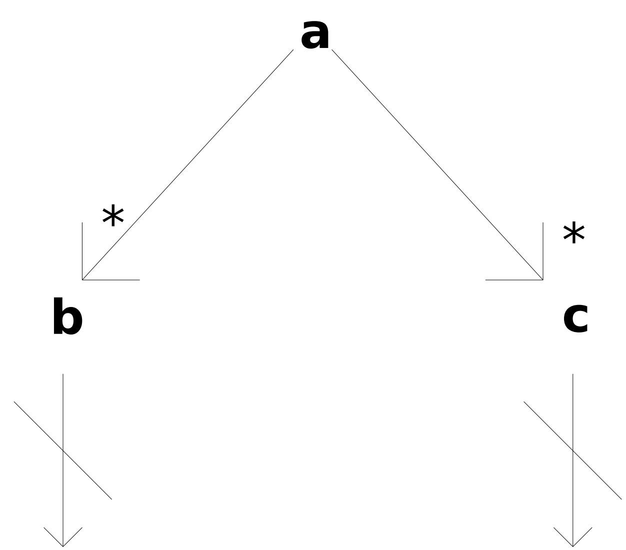
Sémantique . Confluence

Définition :

(A,→) a la propriété de forme normale unique

=

∀**a**,**b**,**c** ∈ A, si



alors **b** = **c**

Sémantique . Confluence

Exemples (forme normale unique):

- $(\mathbb{N}, <)$
- $(\mathbb{N}, >)$
- $(\mathbb{N}^*, |)$
- $(\mathbb{N}, \rightarrow)$
- $(\mathbb{N}, \textit{double\_ou\_plus\_un})$

Sémantique . Confluence

Exemples (forme normale unique):

- ($\mathbb{N}$, <)
- ($\mathbb{N}$, >)
- ($\mathbb{N}$*, |)
- ($\mathbb{N}$, ➔)   1456 ➔ 456 ➔ 45 ➔ 4
- ($\mathbb{N}$, *double_ou_plus_un*)

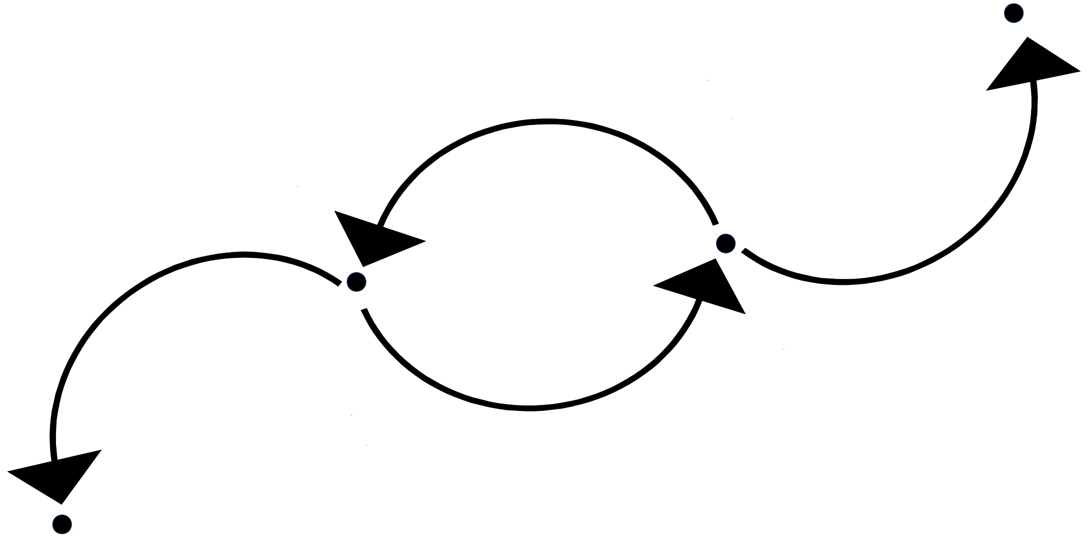
Sémantique . Confluence

Exemples (forme normale unique):

- ($\mathbb{N}$, $<$) V
- ($\mathbb{N}$, $>$) V
- ($\mathbb{N}^*$, $|$) V
- ($\mathbb{N}$, $\rightarrow$) F
- ($\mathbb{N}$, *double_ou_plus_un*) V

Sémantique . Confluence

Contre-exemple:



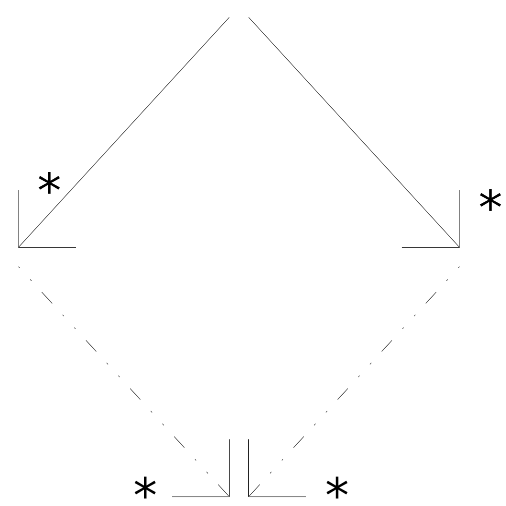
F

Sémantique . Confluence

Définition :

(A,→) confluent

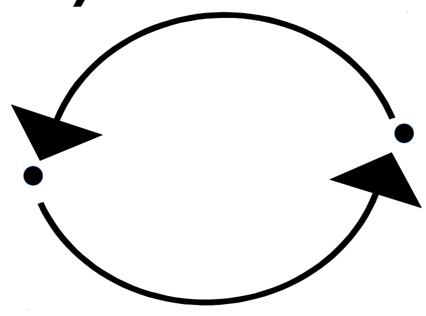
=



Sémantique . Confluence

Exemple :

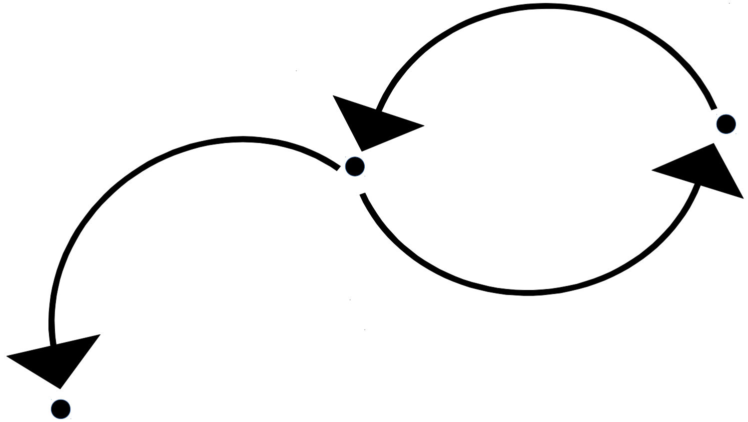
(A,→) confluent



Sémantique . Confluence

Exemple :

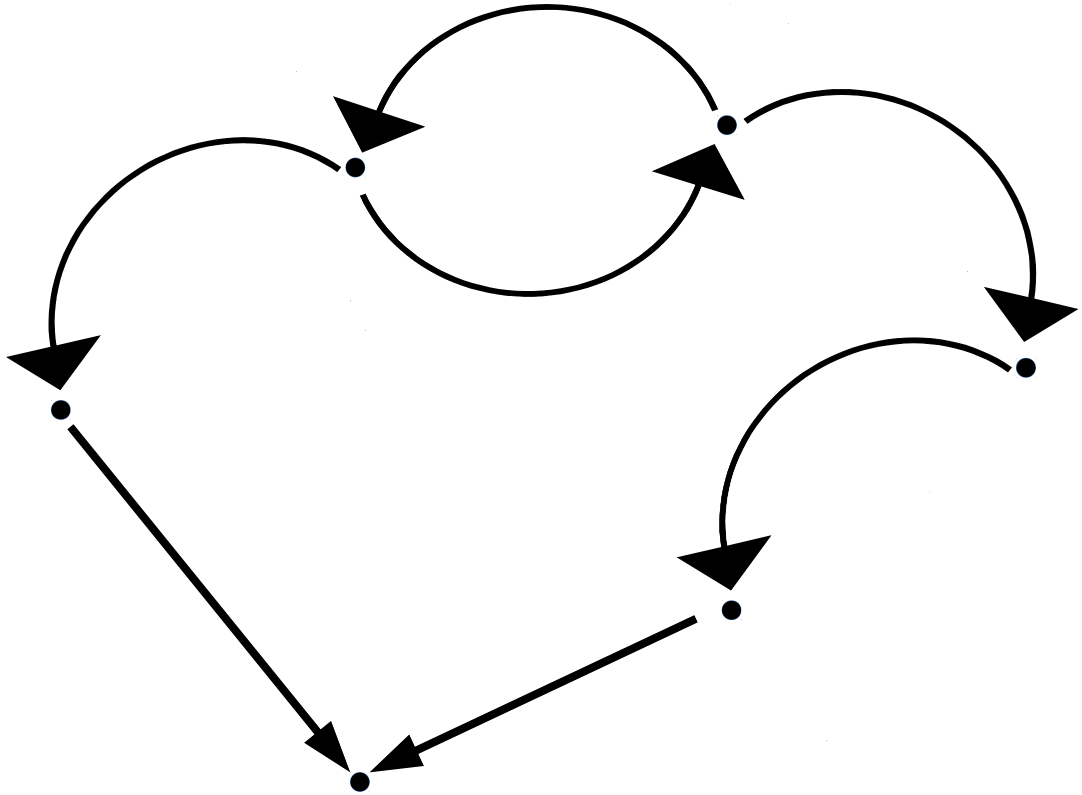
(A,→) confluent



Sémantique . Confluence

Exemple :

(A,→) confluent



Sémantique . Confluence

Exemples (confluence):

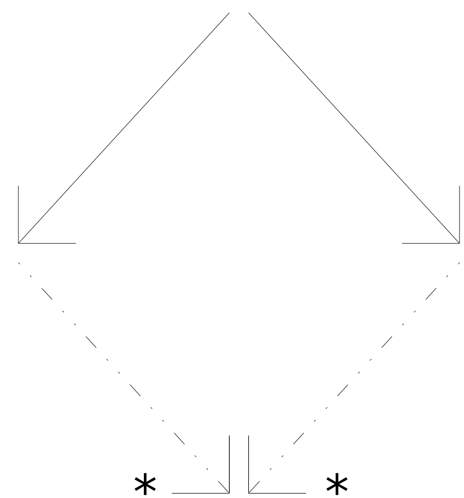
- $(\mathbb{N}, <)$ V
- $(\mathbb{N}, >)$ V
- $(\mathbb{N}^*, |)$ V
- $(\mathbb{N}, \rightarrow)$ F
- $(\mathbb{N}$, *double_ou_plus_un*) V

Sémantique . Confluence

Définition :

(A,→) localement confluent

=

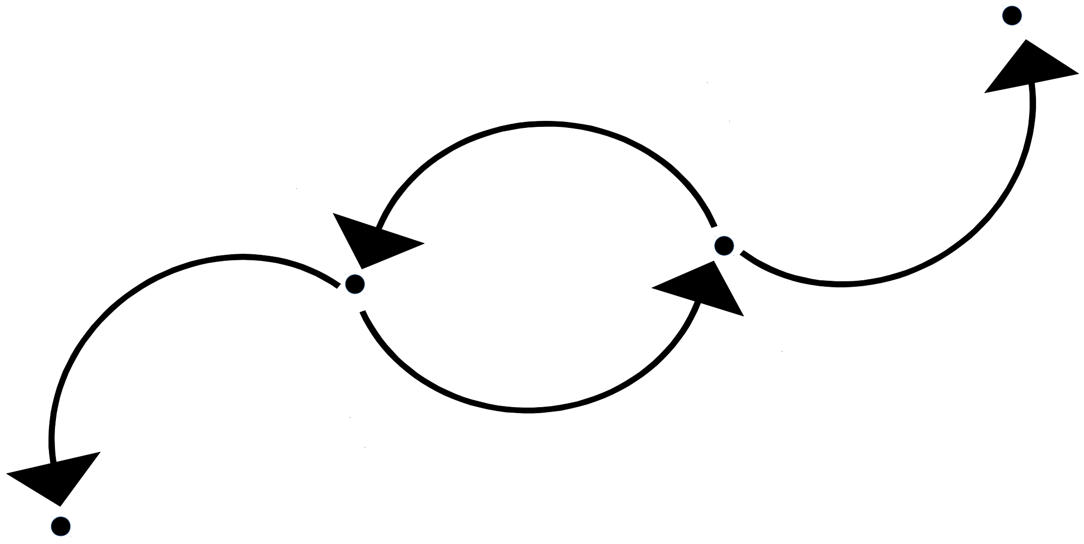


Sémantique . Confluence

Exemple :

(A,→) localement confluent

A :

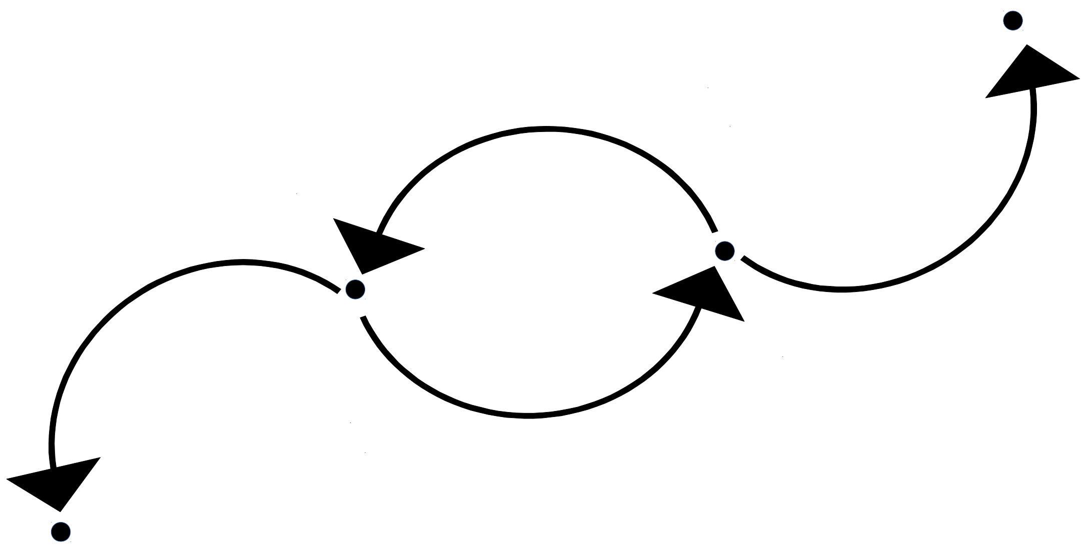


Sémantique . Confluence

Exemple :

(A,→) localement confluent

A :



A n’est pas confluent !

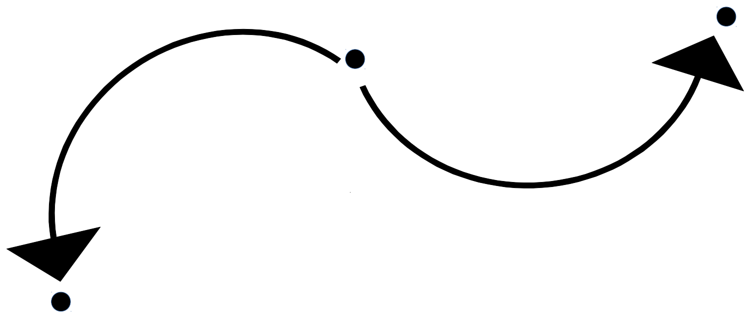
Sémantique . Confluence

Exemples (confluence locale):

- $(\mathbb{N}, <)$ V
- $(\mathbb{N}, >)$ V
- $(\mathbb{N}^*, |)$ V
- $(\mathbb{N}, \rightarrow)$ F
- $(\mathbb{N},$ *double_ou_plus_un*$)$ V

Sémantique . Confluence

Exemples (confluence locale):



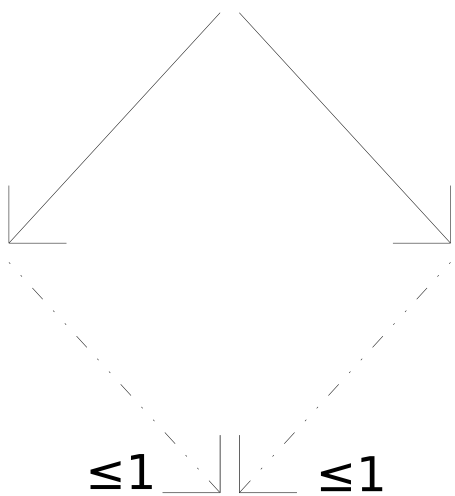
F

Sémantique . Confluence

Définition :

(A,→) fortement confluent

=



Sémantique . Confluence

Exemples (confluence forte):

- $(\mathbb{N}, <)$ V
- $(\mathbb{N}, >)$ V
- $(\mathbb{N}^*, |)$ V
- $(\mathbb{N}, \rightarrow)$ F
- $(\mathbb{N},$ *double_ou_plus_un*$)$ F

Sémantique . Confluence

Remarque :

$(A, \to)$ fortement confluent

implique

$(A, \to)$ confluent

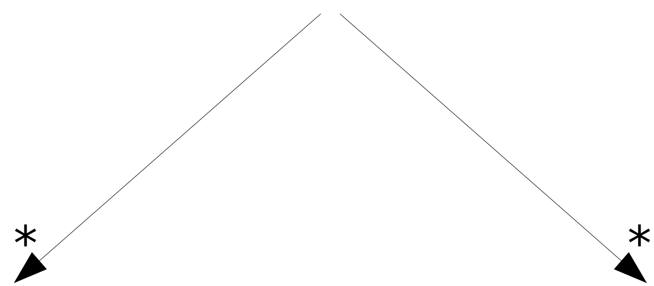
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



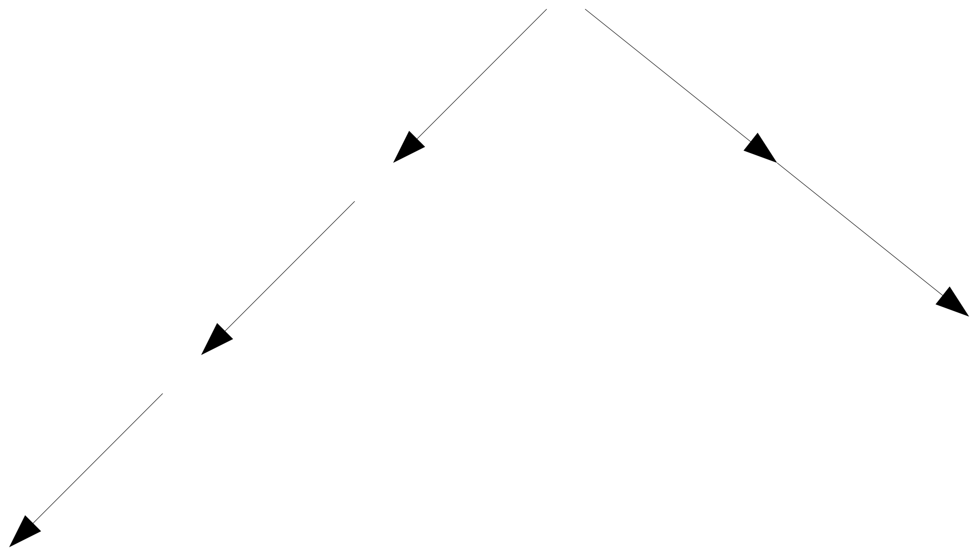
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



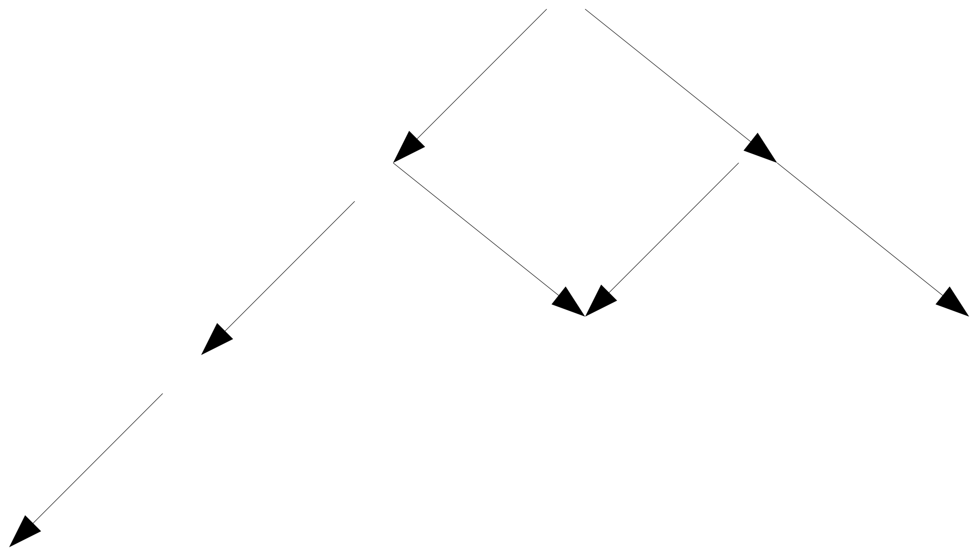
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



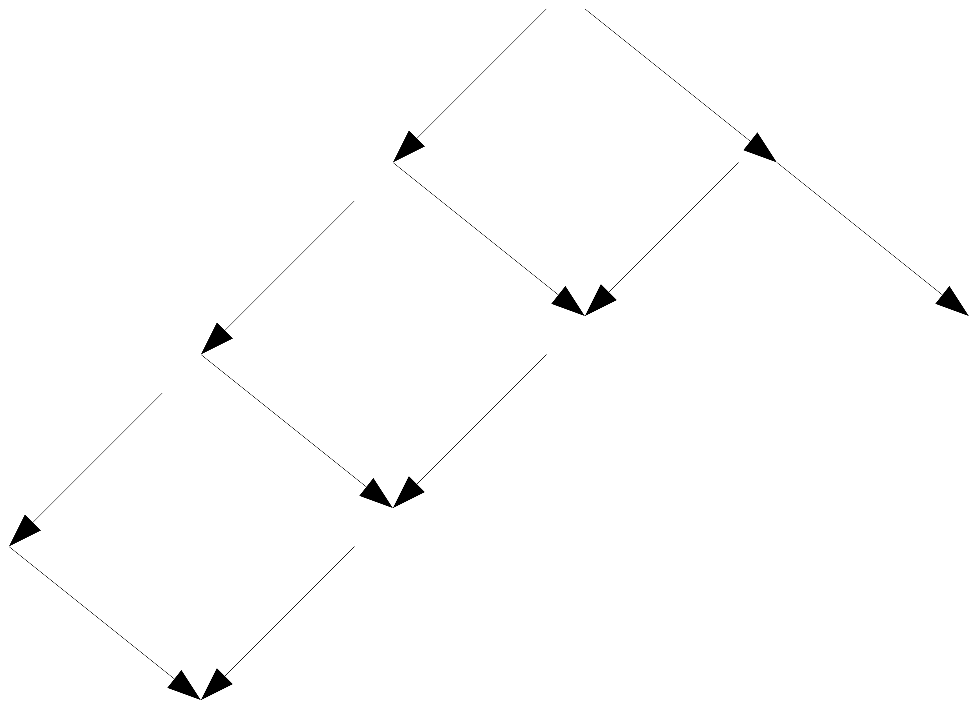
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



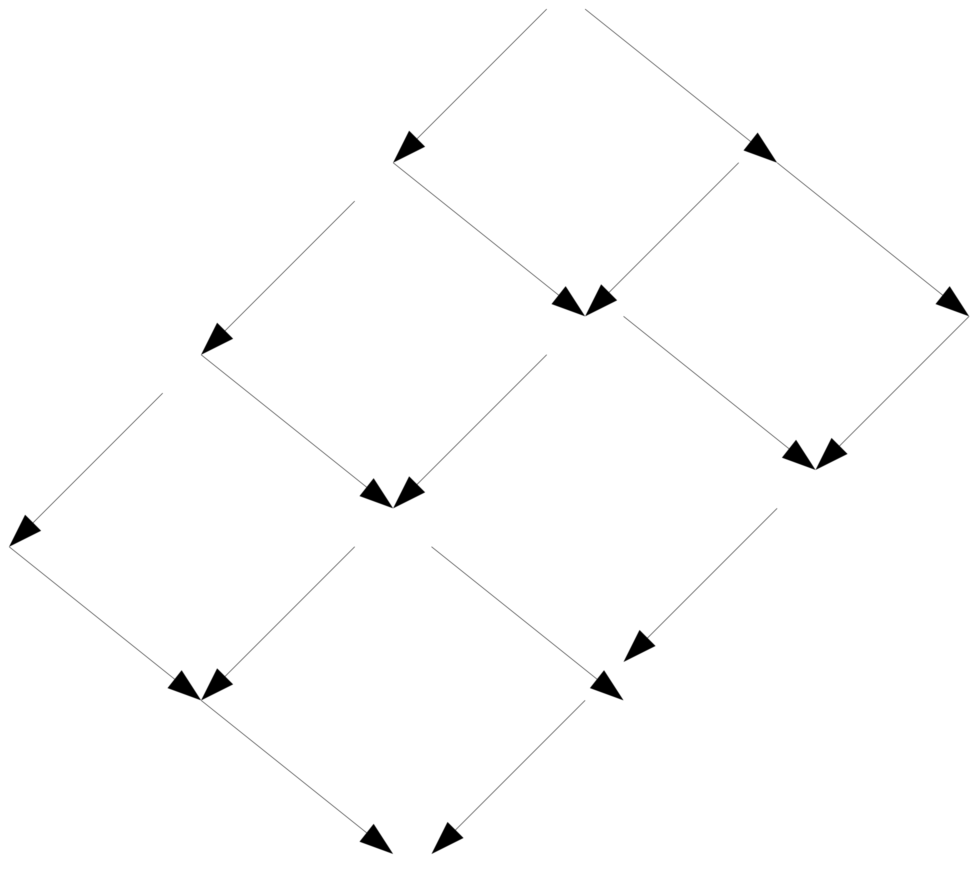
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



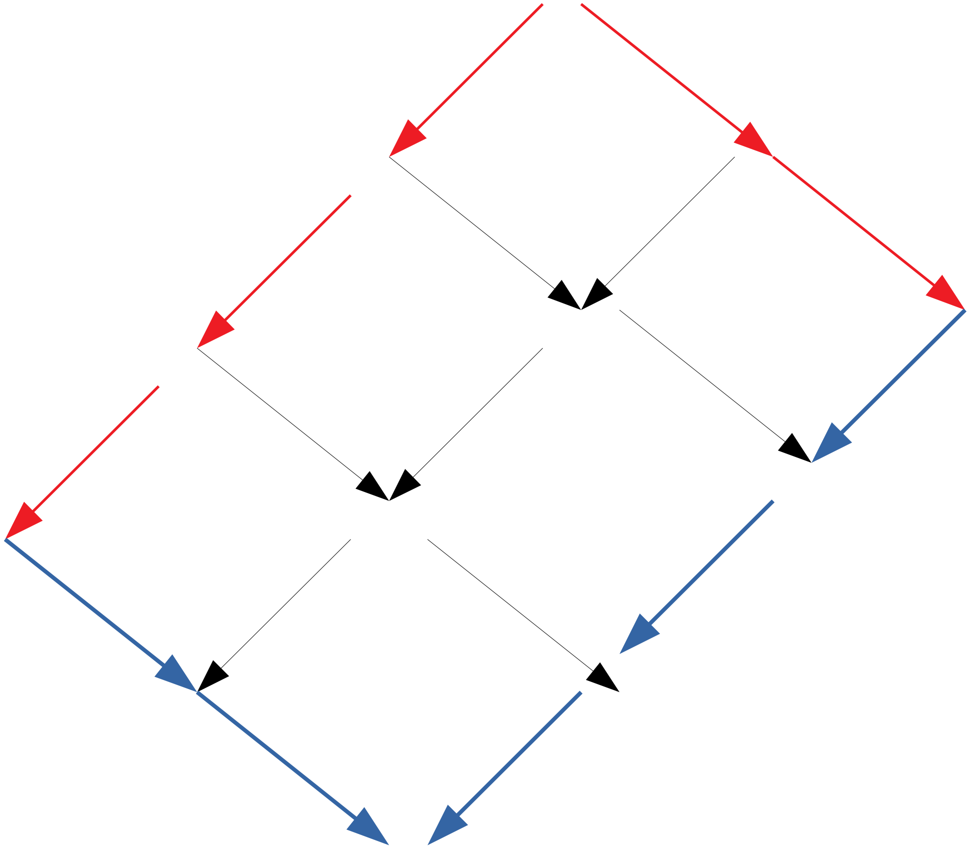
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



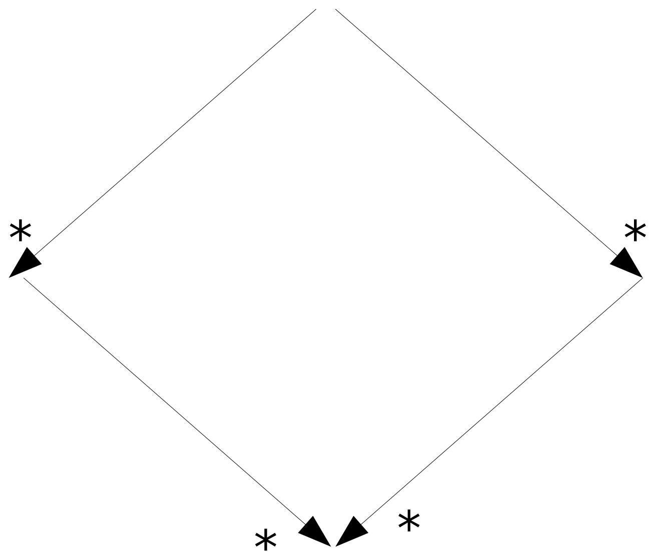
Sémantique . Confluence

(A,→) fortement confluent

implique

(A,→) confluent

Démonstration :



Sémantique . Confluence

Confluence forte de la β-réduction?

Sémantique . Confluence

Confluence forte de la β-réduction?

NON !

Sémantique . Confluence

Confluence forte de la β-réduction?

NON !

(λy.yyy) ((λz.zz) x)

Sémantique . Confluence

Confluence forte de la β-réduction?

NON !

$$(\lambda y.yyy)\ ((\lambda z.zz)\ x) \ \rightarrow\ ((\lambda z.zz)\ x)^3$$

Sémantique . Confluence

Confluence forte de la β-réduction?

NON !

$(\lambda y.yyy)\ ((\lambda z.zz)\ x) \rightarrow ((\lambda z.zz)\ x)((\lambda z.zz)\ x)(\ldots$

OU BIEN : $\searrow (\lambda y.yyy)\ (xx)$

Sémantique . Confluence

Confluence forte de la β-réduction?

NON !

$(\lambda y.yyy)\ ((\lambda z.zz)\ x) \;\rightarrow\; ((\lambda z.zz)\ x)^3$

$\searrow$

OU BIEN : $(\lambda y.yyy)\ (xx)$

Sémantique . Confluence

Comment prouver le déterminisme ?

Sémantique . Confluence

Comment prouver le déterminisme ?

→ nouveau type de réductions

β-réduction parallèle

Définition :

- $u \Rightarrow u$ pour tout terme u

β-réduction parallèle

Définition :

- $u \Rightarrow u$ pour tout terme u
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $uv \Rightarrow u'v'$

β-réduction parallèle

Définition :

- $u \Rightarrow u$ pour tout terme u
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $uv \Rightarrow u'v'$
- si $u \Rightarrow u'$ alors $\lambda x.u \Rightarrow \lambda x.u'$

β-réduction parallèle

Définition :

- $u \Rightarrow u$ pour tout terme u
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $uv \Rightarrow u'v'$
- si $u \Rightarrow u'$ alors $\lambda x.u \Rightarrow \lambda x.u'$
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $(\lambda x.u)v \Rightarrow u'[x:=v']$

β-réduction parallèle

Définition :

- $u \Rightarrow u$ pour tout terme u
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $uv \Rightarrow u'v'$
- si $u \Rightarrow u'$ alors $\lambda x.u \Rightarrow \lambda x.u'$
- si $u \Rightarrow u'$ et $v \Rightarrow v'$ alors $(\lambda x.u)v \Rightarrow u'[x:=v']$

$\Rightarrow$ est la plus petite relation binaire vérifiant ces conditions et le α-renommage

β-réduction parallèle

Théorème 1 :

> La β-réduction parallèle ($\Rightarrow$) est fortement confluente

β-réduction parallèle

Théorème 1 :

> La β-réduction parallèle ($\Rightarrow$) est fortement confluente

Idée :
Il y a une β-réduction parallèle “maximale”, qu’on peut atteindre après avoir fait une réduction quelconque

β-réduction parallèle

Théorème 1 :

La β-réduction parallèle (⇒) est fortement confluente

Exemple:

((λz.zz) x)((λz.zz) x)((λz.zz)x)

β-réduction parallèle

Théorème 1 :

La β-réduction parallèle (⇒) est fortement confluente

Exemple:

$$((\lambda z.zz)\ x)((\lambda z.zz)\ x)((\lambda z.zz)x)$$

$$\downarrow$$

$$xx(xx)(xx)$$

β-réduction parallèle

Théorème 1 :

La β-réduction parallèle (⇒) est fortement confluente

Exemple:

$$
\begin{tikzcd}
((\lambda z.zz)\ x)((\lambda z.zz)\ x)((\lambda z.zz)x) \arrow[dd] \arrow[dr] & \\
& xx\ ((\lambda z.zz)x)(xx) \arrow[dl] \\
xx(xx)(xx) &
\end{tikzcd}
$$

β-réduction parallèle

Théorème 2:

$$\rightarrow \;\subset\; \Rightarrow \;\subset\; \rightarrow^*$$

β-réduction parallèle

Théorème 2:

$$\rightarrow \;\subset\; \Rightarrow \;\subset\; \rightarrow^*$$

Si u → v alors u ⇒ v

Si u ⇒ v alors u → u1 → u2 … → v

β-réduction parallèle

Théorème 2:

$$\to \;\subset\; \Rightarrow \;\subset\; \to^*$$

Conséquence:

Si $u \to^* v$ alors $u \Rightarrow^* v$

$u \to^* w$ $\quad$ $u \Rightarrow^* w$

$$\begin{tikzcd}
 & u \arrow[dl, "*"'] \arrow[dr, "*"] & \\
v \arrow[dr, dashed, "*"'] & & w \arrow[dl, dashed, "*"] \\
 & s &
\end{tikzcd}$$

β-réduction parallèle

Théorème 2:

$$\rightarrow \;\subset\; \Rightarrow \;\subset\; \rightarrow^*$$

Conséquence:

Si $u \rightarrow^* v$ alors $u \Rightarrow^* v$

$u \rightarrow^* w$ $u \Rightarrow^* w$

Par confluence (de $\Rightarrow$) :

$\exists s \mid$ $v \Rightarrow^* s$

$w \Rightarrow^* s$

Ainsi, $\exists s \mid$ $v \rightarrow^* s$

$w \rightarrow^* s$

β-réduction parallèle

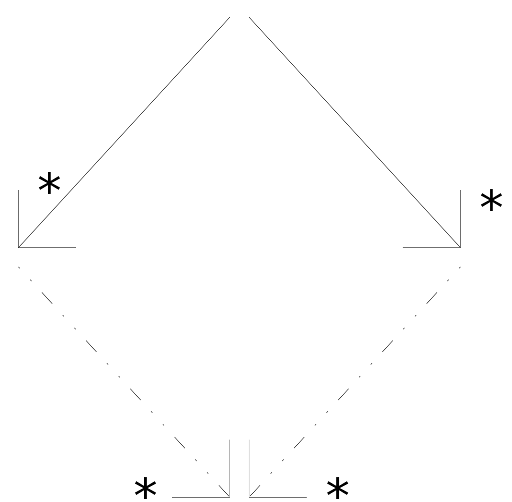
Théorème 3:

Le λ-calcul est confluent

β-réduction parallèle

Théorème 3:

Le λ-calcul est confluent



β-réduction parallèle

Théorème 3:

Le λ-calcul est confluent

Donc (λ, →) a la propriété
de forme normale unique !

β-réduction parallèle

Théorème 3:

Le λ-calcul est confluent

Donc aucun λ-terme n'a deux valeurs !

Conclusion

λ-calcul

λ-calcul - Conclusion

- Simple (3 règles)
- Fonctions (α, β ...)
- Déterministe (confluent)

λ-calcul - Conclusion

- Simple (3 règles)
- Fonctions (α, β ...)
- Déterministe (confluent)
- Turing-Complet

λ-calcul - Conclusion

- Des questions ?

Bonus – entiers de Church

- 0 : λf.λx.x
- 1 : λf.λx.fx
- 2 : λf.λx.f(fx)

  ...
- n : λf.λx.f^n(x)

Bonus – entiers de Church

- 0 : λfx.x
- 1 : λfx.fx
- 2 : λfx.f(fx)

  …
- n : λfx.f^n(x)

Successeur (+1) :

S : λn.(λf.λx.n f (f x)) = λnfx.nf(fx)

Exemple :

S 1 = (λn.(λf.λx.n f (f x))) 1
= λf.λx.1 f (f x)
= λf.λx.f(f(x))
= 2

Bonus – entiers de Church

- 0 : λfx.x
- 1 : λfx.fx
- 2 : λfx.f(fx)

  …
- n : $\lambda fx.f^{n}(x)$

Successeur (+1) :

S : λn.(λf.λx.n f (f x)) = λnfx.nf(fx)

Exemple :

S 1 = (λn.(λf.λx.n f (f x))) 1
= λf.λx.1 f (f x)
= λf.λx.f(f(x))
= 2

Addition :

+ : λnp.nSp → appliquer n fois la fonction S à p