

Curs 6

- În 1929-1932 Church a propus λ -calculul ca sistem formal pentru logica matematică. În 1935 a argumentat că orice funcție calculabilă peste numere naturale poate fi calculată în λ -calcul.
- În 1935, independent de Church, Turing a dezvoltat mecanismul de calcul numit astăzi Mașina Turing. În 1936 și el a argumentat că orice funcție calculabilă peste numere naturale poate fi calculată de o mașină Turing. De asemenea, a arătat echivalența celor două modele de calcul. Această echivalență a constituit o indicație puternică asupra "universalității" celor două modele, conducând la ceea ce numim astăzi "Teza Church-Turing".

Referințe

- Benjamin C. Pierce, Types and Programming Languages, The MIT Press 2002
- J.R. Hindley, J.P. Seldin, Lambda-Calculus and Combinators, an Introduction, Cambridge University Press, 2008
- R. Nederpelt, H. Geuvers, Type Theory and Formal Proof, an Introduction, Cambridge University Press 2014

λ -calcul: sintaxa

Lambda Calcul - sintaxă

$t =$ x (variabilă)
 $| \lambda x. t$ (abstractizare)
 $| t t$ (aplicare)

λ -calcul: sintaxa

Lambda Calcul - sintaxă

$$\begin{array}{ll} t = & x \quad \text{(variabilă)} \\ & | \lambda x. t \quad \text{(abstractizare)} \\ & | t \ t \quad \text{(aplicare)} \end{array}$$

λ -termeni

Fie $Var = \{x, y, z, \dots\}$ o mulțime infinită de variabile.
Mulțimea λT termenilor λT este definită inductiv astfel:

[Variabilă] $Var \subseteq \lambda T$

[Aplicare] dacă $t_1, t_2 \in \lambda T$ atunci $(t_1 \ t_2) \in \lambda T$

[Abstractizare] dacă $x \in Var$ și $t \in \lambda T$ atunci $(\lambda x. t) \in \lambda T$

Lambda termeni

λ -termeni: exemple

- x, y, z
- $(xy), (yx), (x(yx))$
- $(\lambda x.x), (\lambda x.(xy)), (\lambda z.(xy)), (\lambda x.(\lambda z.(xy)))$
- $((\lambda x.x)y), ((\lambda x.(xz))y), ((\lambda x.x)(\lambda y.y))$

Lambda termeni

λ -termeni: exemple

- x, y, z
- $(xy), (yx), (x(yx))$
- $(\lambda x.x), (\lambda x.(xy)), (\lambda z.(xy)), (\lambda x.(\lambda z.(xy)))$
- $((\lambda x.x)y), ((\lambda x.(xz))y), ((\lambda x.x)(\lambda y.y))$

Convenții:

- se elimină parantezele exterioare
- aplicarea este asociativă la stînga: $t_1 t_2 t_3$ este $(t_1 t_2) t_3$
- corpul abstractizării este extins la dreapta: $\lambda x.t_1 t_2$ este $\lambda x.(t_1 t_2)$ (nu $(\lambda x.t_1) t_2$)
- scriem $\lambda xyz.t$ în loc de $\lambda x.\lambda y.\lambda z.t$

Lambda termeni / Funcții anonime

λ -termeni: exemple

- x, y, z
- $(xy), (yx), (x(yx))$
- $(\lambda x.x), (\lambda x.(xy)), (\lambda z.(xy)), (\lambda x.(\lambda z.(xy)))$
- $((\lambda x.x)y), ((\lambda x.(xz))y), ((\lambda x.x)(\lambda y.y))$

λ -termeni/ funcții anonime în Haskell

În Haskell, \backslash e folosit în locul simbolului λ și \rightarrow în locul punctului.

$\lambda x.x * x$ este $\backslash x \rightarrow x * x$

$\lambda x.x > 0$ este $\backslash x \rightarrow x > 0$

Variabile libere și legate

Apariții libere și legate

Pentru un termen $\lambda x.t$ spunem că:

- aparițiile variabilei x în t sunt legate (*bound*)
- λx este legătura (*binder*), iar t este domeniul (*scope*) legării
- o apariție a unei variabile este liberă (*free*) dacă apare într-o poziție în care nu e legată.

Un termen fără variabile libere se numește închis (*closed*).

Exemplu:

- $\lambda x.x$ este un termen închis
- $\lambda x.xy$ nu este termen închis, x este legată, y este liberă
- în termenul $x(\lambda x.xy)$ prima apariție a lui x este liberă, a doua este legată.

Variabile libere

Mulțimea variabilelor libere $FV(t)$

Pentru un λ -termen t mulțimea variabilelor libere este definită astfel:

$$[\text{Variabilă}] \quad FV(x) = x$$

$$[\text{Aplicare}] \quad FV(t_1 t_2) = FV(t_1) \cup FV(t_2)$$

$$[\text{Abstractizare}] \quad FV(\lambda x.t) = FV(t) \setminus \{x\}$$

Exemplu:

$$\begin{aligned} FV(\lambda x.xy) &= FV(xy) \setminus \{x\} \\ &= (FV(x) \cup FV(y)) \setminus \{x\} \\ &= (\{x\} \cup \{y\}) \setminus \{x\} \\ &= \{y\} \end{aligned}$$

Variabile libere

Mulțimea variabilelor libere $FV(t)$

Pentru un λ -termen t mulțimea variabilelor libere este definită astfel:

$$[\text{Variabilă}] \quad FV(x) = x$$

$$[\text{Aplicare}] \quad FV(t_1 t_2) = FV(t_1) \cup FV(t_2)$$

$$[\text{Abstractizare}] \quad FV(\lambda x.t) = FV(t) \setminus \{x\}$$

Exemplu:

$$\begin{aligned} FV(\lambda x.xy) &= FV(xy) \setminus \{x\} \\ &= (FV(x) \cup FV(y)) \setminus \{x\} \\ &= (\{x\} \cup \{y\}) \setminus \{x\} \\ &= \{y\} \end{aligned}$$

$$FV(x\lambda x.xy) =$$

Variabile libere

Mulțimea variabilelor libere $FV(t)$

Pentru un λ -termen t mulțimea variabilelor libere este definită astfel:

$$[\text{Variabilă}] \quad FV(x) = x$$

$$[\text{Aplicare}] \quad FV(t_1 t_2) = FV(t_1) \cup FV(t_2)$$

$$[\text{Abstractizare}] \quad FV(\lambda x.t) = FV(t) \setminus \{x\}$$

Exemplu:

$$\begin{aligned} FV(\lambda x.xy) &= FV(xy) \setminus \{x\} \\ &= (FV(x) \cup FV(y)) \setminus \{x\} \\ &= (\{x\} \cup \{y\}) \setminus \{x\} \\ &= \{y\} \\ FV(x\lambda x.xy) &= \{x, y\} \end{aligned}$$

Substituții

Fie t un λ -termen $x \in Var$.

Definiție intuitivă

Pentru un λ -termen u vom nota prin $[u/x]t$ rezultatul înlocuirii tuturor aparițiilor libere ale lui x cu u în t .

Exemple: Dacă x, y, z sunt variabile distincte atunci

- $[y/x]\lambda z.x = \lambda z.y$
- $[(\lambda z.zw)/x](\lambda y.x) = \lambda y.\lambda z.zw$

Definirea substituției

Rezultatul substituirii lui x cu u în t este definit astfel:

[Variabilă] $[u/x]x = u$

[Variabilă] $[u/x]y = y$ dacă $x \neq y$

[Aplicare] $[u/x](t_1 t_2) = [u/x]t_1 [u/x]t_2$

[Abstractizare] $[u/x]\lambda y.t = \lambda y.[u/x]t$ unde
 $y \neq x$ și $y \notin FV(u)$

Substituții

Exemple: Dacă x, y, z sunt variabile distincte atunci

- $[y/x]\lambda z.x = \lambda z.y$
- Cine este $[y/x]\lambda y.x$?

Substituții

Exemple: Dacă x, y, z sunt variabile distincte atunci

□ $[y/x]\lambda z.x = \lambda z.y$

□ Cine este $[y/x]\lambda y.x$?

Dacă folosim definiția intuitivă obținem

$[y/x]\lambda y.x = \lambda y.y$ ceea ce este greșit!

Substituții

Exemple: Dacă x, y, z sunt variabile distincte atunci

□ $[y/x]\lambda z.x = \lambda z.y$

□ Cine este $[y/x]\lambda y.x$?

Dacă folosim definiția intuitivă obținem

$[y/x]\lambda y.x = \lambda y.y$ ceea ce este greșit!

Cum procedăm pentru a repara greșeala? Observăm că $\lambda y.x$ desemnează o funcție constantă, aceeași funcție putând fi reprezentată prin $\lambda z.x$. Aplicarea corectă a substituției este:

$$[y/x]\lambda y.x = [y/x]\lambda z.x = \lambda z.y$$

Substituții

Exemple: Dacă x, y, z sunt variabile distincte atunci

□ $[y/x]\lambda z.x = \lambda z.y$

□ Cine este $[y/x]\lambda y.x$?

Dacă folosim definiția intuitivă obținem

$[y/x]\lambda y.x = \lambda y.y$ ceea ce este greșit!

Cum procedăm pentru a repara greșeala? Observăm că $\lambda y.x$ desemnează o funcție constantă, aceeași funcție putând fi reprezentată prin $\lambda z.x$. Aplicarea corectă a substituției este:

$$[y/x]\lambda y.x = [y/x]\lambda z.x = \lambda z.y$$

Avem libertatea de a redenumi variabilele legate!

α -conversie (α -echivalență)

α -conversia $=_\alpha$

[Reflexivitate] $t =_\alpha t$

[Simetrie] $t_1 =_\alpha t_2$ implică $t_2 =_\alpha t_1$

[Tranzitivitate] $t_1 =_\alpha t_2$ și $t_2 =_\alpha t_3$ implică $t_1 =_\alpha t_3$

[Redenumire] $\lambda x.t =_\alpha \lambda y.[y/x]t$ dacă $y \notin FV(t)$

[Compatibilitate] $t_1 =_\alpha t_2$ implică

$tt_1 =_\alpha tt_2, t_1t =_\alpha t_2t$ și $\lambda x.t_1 =_\alpha \lambda x.t_2$

α -conversie (α -echivalență)

α -conversia $=_{\alpha}$

[Reflexivitate] $t =_{\alpha} t$

[Simetrie] $t_1 =_{\alpha} t_2$ implică $t_2 =_{\alpha} t_1$

[Tranzitivitate] $t_1 =_{\alpha} t_2$ și $t_2 =_{\alpha} t_3$ implică $t_1 =_{\alpha} t_3$

[Redenumire] $\lambda x.t =_{\alpha} \lambda y.[y/x]t$ dacă $y \notin FV(t)$

[Compatibilitate] $t_1 =_{\alpha} t_2$ implică

$tt_1 =_{\alpha} tt_2$, $t_1 t =_{\alpha} t_2 t$ și $\lambda x.t_1 =_{\alpha} \lambda x.t_2$

$t_1 =_{\alpha} t_2$ și $u_1 =_{\alpha} u_2$ implică $[u_1/x]t_1 =_{\alpha} [u_2/x]t_2$

Exemplu:

$[xy/x](\lambda y.yx) =_{\alpha} [xy/x](\lambda z.zx) =_{\alpha} \lambda z.z(xy)$

α -conversie (α -echivalență)

α -conversia $=_{\alpha}$

[Reflexivitate] $t =_{\alpha} t$

[Simetrie] $t_1 =_{\alpha} t_2$ implică $t_2 =_{\alpha} t_1$

[Tranzitivitate] $t_1 =_{\alpha} t_2$ și $t_2 =_{\alpha} t_3$ implică $t_1 =_{\alpha} t_3$

[Redenumire] $\lambda x.t =_{\alpha} \lambda y.[y/x]t$ dacă $y \notin FV(t)$

[Compatibilitate] $t_1 =_{\alpha} t_2$ implică

$tt_1 =_{\alpha} tt_2$, $t_1 t =_{\alpha} t_2 t$ și $\lambda x.t_1 =_{\alpha} \lambda x.t_2$

$t_1 =_{\alpha} t_2$ și $u_1 =_{\alpha} u_2$ implică $[u_1/x]t_1 =_{\alpha} [u_2/x]t_2$

Exemplu:

$[xy/x](\lambda y.yx) =_{\alpha} [xy/x](\lambda z.zx) =_{\alpha} \lambda z.z(xy)$

Vom lucra modulo α -conversie, doi termeni α -echivalenți vor fi considerați "egali".

Substituție

Exemplu:

$$\square [xy/x]\lambda x.yx =_{\alpha} [xy/x]\lambda z.yz = \lambda z.[xy/x](yz) = \lambda z.yz$$

Observăm că $\lambda z.yz =_{\alpha} \lambda x.yx$

Substituție

Exemplu:

$$\square [xy/x]\lambda x.yx =_{\alpha} [xy/x]\lambda z.yz = \lambda z.[xy/x](yz) = \lambda z.yz$$

Observăm că $\lambda z.yz =_{\alpha} \lambda x.yx$

Exemplu:

$$\square [xy/x]\lambda x.yx =_{\alpha} [xy/x]\lambda z.yz = \lambda z.[xy/x](yz) = \lambda z.yz$$

Observăm că $\lambda z.yz =_{\alpha} \lambda x.yx$

$$\square [y/z]\lambda xy.zzx = \lambda x.[y/z]\lambda y.zzx =_{\alpha} \lambda x.[y/z]\lambda v.zzx = \lambda x.\lambda v.[y/z](zzx) = \lambda xv.yyx$$

β -reducție

β -reducția este o relație pe mulțimea α -termenilor.

β -reducția $\rightarrow_\beta, \rightarrow_\beta^*$

□ un singur pas $\rightarrow_\beta \subseteq \Lambda T \times \Lambda T$

[Aplicarea] $(\lambda x.t)u \rightarrow_\beta [u/x]t$

[Compatibilitatea] $t_1 \rightarrow_\beta t_2$ implică

$tt_1 \rightarrow_\beta tt_2, t_1 t \rightarrow_\beta t_2 t$ și $\lambda x.t_1 \rightarrow_\beta \lambda x.t_2$

□ zero sau mai mulți pași $\rightarrow_\beta^* \subseteq \Lambda T \times \Lambda T$

$t_1 \xrightarrow{*}_\beta t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_\alpha u_0 \rightarrow_\beta u_1 \rightarrow_\beta \dots \rightarrow_\beta u_n =_\alpha t_2$

β -reducție

Să considerăm termenul $(\lambda x.(\lambda y.yx)z)v$

$$\square (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda y.yv)z \rightarrow_{\beta} zv$$

β -reducție

Să considerăm termenul $(\lambda x.(\lambda y.yx)z)v$

$$\square (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda y.yv)z \rightarrow_{\beta} zv$$

$$\square (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda x.zx)v \rightarrow_{\beta} zv$$

Să considerăm termenul $(\lambda x.(\lambda y.yx)z)v$

$$\square (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda y.yv)z \rightarrow_{\beta} zv$$

$$\square (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda x.zx)v \rightarrow_{\beta} zv$$

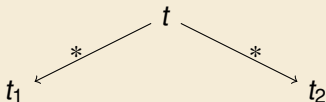
Observăm că un termen poate fi β -redus în mai multe moduri.

Proprietatea de confluență ne asigură că vom ajunge întotdeauna la același rezultat.

Confluența β -reducției

Teorema Church-Rosser

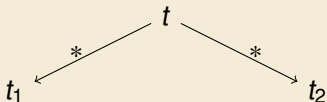
Dacă $t \xrightarrow{*}_{\beta} t_1$ și $t \xrightarrow{*}_{\beta} t_2$



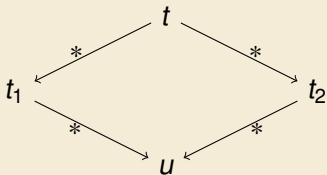
Confluența β -reducției

Teorema Church-Rosser

Dacă $t \xrightarrow{*}_{\beta} t_1$ și $t \xrightarrow{*}_{\beta} t_2$



atunci există u astfel încât $t_1 \xrightarrow{*}_{\beta} u$ și $t_2 \xrightarrow{*}_{\beta} u$.



β -forma normală

Intuitiv, o formă normală este un termen care nu mai poate fi redus (sau punctul final al unui calcul).

Formă normală

- un λ -termen căruia nu i se mai poate aplica reducerea într-un pas \rightarrow_β se numește β -formă normală
- dacă $t \xrightarrow{*}_\beta u_1$, $t \xrightarrow{*}_\beta u_2$ și u_1, u_2 sunt η -forme normale atunci, datorită confluenței, $u_1 =_\alpha u_2$
- un λ -termen poate avea cel mult o β -formă normală (modulo α -echivalență)

β -forma normală

Formă normală

- un λ -termen căruia nu i se mai poate aplica reducerea într-un pas \rightarrow_β se numește β -formă normală
- dacă $t \xrightarrow{*}_\beta u_1$, $t \xrightarrow{*}_\beta u_2$ și u_1, u_2 sunt η -forme normale atunci, datorită confluenței, $u_1 =_\alpha u_2$
- un λ -termen poate avea cel mult o β -formă normală (modulo α -echivalență)

Exemplu:

- zv este β -formă normală pentru $(\lambda x.(\lambda y.yx)z)v$
 $(\lambda x.(\lambda y.yx)z)v \rightarrow_\beta (\lambda y.yv)z \rightarrow_\beta zv$
- există termeni care **nu** pot fi reduși la o β -formă normală, de exemplu $(\lambda x.xx)(\lambda x.xx)$

β -conversia

Intuitiv, β -conversia extinde β -reducția în ambele direcții.

$$\square (\lambda y. yv)z \rightarrow_{\beta} zv \leftarrow_{\beta} (\lambda x. zx)v$$

β -conversia

Intuitiv, β -conversia extinde β -reducția în ambele direcții.

- $(\lambda y. yv)z \rightarrow_{\beta} zv \leftarrow_{\beta} (\lambda x. zx)v$
- $(\lambda y. yv)z \leftarrow_{\beta} (\lambda x. (\lambda y. yx)z)v \rightarrow_{\beta} (\lambda x. zx)v$

β -conversia

Intuitiv, β -conversia extinde β -reducția în ambele direcții.

- $(\lambda y.yv)z \rightarrow_{\beta} zv \leftarrow_{\beta} (\lambda x.zx)v$
- $(\lambda y.yv)z \leftarrow_{\beta} (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda x.zx)v$

β -conversia $=_{\beta}$

- $=_{\beta} \subseteq \Lambda T \times \Lambda T$
 $t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât
 $t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

β -conversia

Intuitiv, β -conversia extinde β -reducția în ambele direcții.

- $(\lambda y.yv)z \rightarrow_{\beta} zv \leftarrow_{\beta} (\lambda x.zx)v$
- $(\lambda y.yv)z \leftarrow_{\beta} (\lambda x.(\lambda y.yx)z)v \rightarrow_{\beta} (\lambda x.zx)v$

β -conversia $=_{\beta}$

- $=_{\beta} \subseteq \Lambda T \times \Lambda T$
 $t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât
 $t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

Exemplu: $(\lambda y.yv)z =_{\beta} (\lambda x.zx)v$

β -conversia

β -conversia $=_{\beta}$

□ $=_{\beta} \subseteq \Lambda T \times \Lambda T$

$t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

β -conversia

β -conversia $=_{\beta}$

$$\square =_{\beta} \subseteq \Lambda T \times \Lambda T$$

$t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

Observații

$\square =_{\beta}$ este o relație de echivalență

β -conversia

β -conversia $=_{\beta}$

□ $=_{\beta} \subseteq \Lambda T \times \Lambda T$

$t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

Observații

□ $=_{\beta}$ este o relație de echivalență

□ pentru t_1, t_2 λ -termeni și u_1, u_2 β -forme normale

dacă $t_1 \xrightarrow{*}_{\beta} u_1$, $t_2 \xrightarrow{*}_{\beta} u_2$ și $u_1 =_{\alpha} u_2$ atunci $t_1 =_{\beta} t_2$

β -conversia

β -conversia $=_{\beta}$

□ $=_{\beta} \subseteq \Lambda T \times \Lambda T$

$t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

□ $=_{\beta}$ este o relație de echivalență

β -conversia

β -conversia $=_{\beta}$

□ $=_{\beta} \subseteq \Lambda T \times \Lambda T$

$t_1 =_{\beta} t_2$ dacă există $n \geq 0$ și u_0, \dots, u_n astfel încât

$t_1 =_{\alpha} u_0$, $u_n =_{\alpha} t_2$ și, pentru orice i , $u_i \rightarrow_{\beta} u_{i+1}$ sau $u_{i+1} \rightarrow_{\beta} u_i$

□ $=_{\beta}$ este o relație de echivalență

□ pentru t_1, t_2 λ -termeni și u_1, u_2 β -forme normale

dacă $t_1 \xrightarrow{*}_{\beta} u_1$, $t_2 \xrightarrow{*}_{\beta} u_2$ și $u_1 =_{\alpha} u_2$ atunci $t_1 =_{\beta} t_2$

β -conversia reprezintă "egalitatea prin calcul", iar β -reducția (modulo α -conversie) oferă o procedură de decizie pentru aceasta.



Pe săptămâna viitoare!