INLEIDING TOT DE λ-CALCULUS DEEL 2

VRIJE UNIVERSITEIT BRUSSEL

BINDEN VAN VARIABELEN

Voor een λ-expressie van de vorm $\lambda x.M$ zegt men:

λx bindt x in de λ-expressie M

Het bereik van de binding is M, d.w.z. dat alle nog niet gebonden voorkomens van x in $\lambda x.M$ gebonden worden

Voorbeeld:

$$((\underline{\lambda x.\lambda z.((x)z)(\lambda x.x)z})\ \lambda y.y)\ \lambda x.x$$

VRIJE EN GEBONDEN VARIABELEN

In een λ-expressie heten alle voorkomen van variabelen die niet gebonden zijn **vrij**

Definitie

Laat M een λ-expressie zijn. Dan wordt de verzameling $VV(M)$ van vrije variabelen van M als volgt gedefinieerd:

- Voor alle variabelen $x \in V$: $VV(x) = \{x\}$
- $VV(\lambda x.M) = VV(M) \setminus \{x\}$
- $VV((M)N) = VV(M) \cup VV(\mathrm{N})$

Definitie

Een λ-expressie zonder vrije variabelen heet gesloten of een combinator. De verzameling van combinatoren wordt genoteerd als Λ_0

SUBSTITUTIE - INTUÏTIEF

formele parameter

actuele parameter

Uitwerking van $(\lambda x.M)P$

- Intuïtief: door elk voorkomen van de parameter x in M te vervangen door P

Om dit formeel te definiëren moeten we eerst de substitutie definiëren.

SUBSTITUTIE - DEFINITIE

Definitie

Zij P en M twee λ-expressies, en x een variabele $x \in V$.

De substitutie $\{P/x\}M$ van P voor x in M, *is als volgt gedefinieerd:*

$\{P/x\}x = P$ (S1)

$\{P/x\}y = y$ *als* $y \in V \backslash \{x\}$ (S2)

$\{P/x\}(F)Q = (\{P/x\}F)\{P/x\}Q$ (S3)

$\{P/x\}\ \lambda x.M = \lambda x.M$ (S4)

$\{P/x\}\ \lambda y.M = \lambda y.\{P/x\}\ M$ *als* $y \neq x$ *en* $y \notin VV(P)$ (S5)

$\{P/x\}\ \lambda y.M = \lambda z.\{P/x\}\{z/y\}M$

als $y \neq x$ *en* $z \notin VV(P)$ *en* $z \notin VV(M)$ (S6)

Toelichting regel (S6):

$\{P/x\}\ \lambda y.M = \lambda z.\{P/x\}\{z/y\}M$ waarbij y vrije variabele in P

Voorbeeld: $\{\underbrace{\lambda u.(y)u}_{P}/x\}\ \lambda y.\underbrace{(x)y}_{M}$

Vóór substitutie: y is vrij in P: $\lambda u.(y)u$

Bij een directe (verkeerde) substitutie zou y gebonden worden:
$\{P/x\}\ \lambda y.M \;=\; \lambda y.(\lambda u.(y)u)y$

Daarom eerst y vervangen door (bijvoorbeeld) z in M: $\lambda z.(x)z$

Dan x vervangen door P. Het correct antwoord:

$\lambda z.(\lambda u.(y)u)z$

VOORBEELD SUBSTITUTIE

$\{\lambda u.(y)u/x\}\ \lambda y.(x)y$ $\{P/x\}M$

$P \equiv \lambda u.(y)u$, $M \equiv \lambda y.(x)y$ en $y \in VV(P)$ dus (S6): $\{P/x\}\ \lambda y.M = \lambda z.\{P/x\}\{z/y\}M$

$\lambda z.\{\lambda u.(y)u/x\}\{z/y\}(x)y$ dan (S3): $\{P/x\}(F)Q = (\{P/x\}F)\{P/x\}Q$

$\lambda z.\{\lambda u.(y)u/x\}(\{z/y\}x)\{z/y\}y$ dan (S2): $\{P/x\}y = y$ als $y \in V\backslash\{x\}$

$\lambda z.\{\lambda u.(y)u/x\}(x)\{z/y\}y$ dan (S1): $\{P/x\}x = P$

$\lambda z.\{\lambda u.(y)u/x\}(x)z$ dan (S3): $\{P/x\}(F)Q = (\{P/x\}F)\{P/x\}Q$

$\lambda z.(\{\lambda u.(y)u/x\}x)\{\lambda u.(y)u/x\}z$ dan (S2): $\{P/x\}y = y$ als $y \in V\backslash\{x\}$

$\lambda z.(\{\lambda u.(y)u/x\}x)z$ dan (S1): $\{P/x\}x = P$

$\lambda z.(\lambda u.(y)u)z$

β-GELIJKHEID - DEFINITIE

Definitie

De β-gelijkheid relatie $=_\beta \subseteq \Lambda \times \Lambda$ is gedefinieerd als volgt:

1. $(\lambda x.M)P =_\beta \{P/x\}M$ (β)
2. $\lambda x.M =_\beta \lambda z.\{z/x\}M$ als $z \notin VV(M)$ (α)
3. $M =_\beta M$ (reflexief)
4. $M =_\beta N$ dan ook $N =_\beta M$ (symmetrisch)
5. $M =_\beta N$ en $N =_\beta L$ dan ook $M =_\beta L$ (transitief)
6. $M =_\beta M'$ en $P =_\beta P'$ dan $(M)P =_\beta (M')P'$ (congruent)
7. $M =_\beta M'$ dan $\lambda x.M =_\beta \lambda x.M'$ (congruent)

VRIJE UNIVERSITEIT BRUSSEL

REKENEN IN λ-CALCULUS

λ-calculus kent geen constanten

- Toch kan men de natuurlijke getallen voorstellen d.m.v. de λ-calculus
- We kunnen zelf een volledige programmeertaal maken met:
  - Constanten, rekenkundige functies
  - Booleans, if-then constructies
  - Recursie
  - ....

REKENEN IN λ-CALCULUS

Definitie

Beschouw λ-expressies F en M. De expressie $(F)^n\ M$ wordt inductief gedefinieerd als volgt:

$$(F)^0\ M \equiv M$$

$$(F)^{1+n}\ M \equiv (F)(F)^n\ M$$

Lemma ("+")

Beschouw λ-expressies F en M.
Voor alle n, $m \in$ IN geldt dat

$$(F)^{n+\mathrm{m}}M \equiv (F)^n(F)^m M$$

Bewijs: per inductie op n

Geval n = 0 $\quad (F)^{0+\mathrm{m}}M \quad \equiv (F)^m M$

definieer $M' \equiv (F)^m M$

$\equiv M'$

$\equiv (F)^0 M' \quad$ *(def)*

$\equiv (F)^0\,(F)^m M$

REKENEN IN λ-CALCULUS

Stel lemma geldt voor het **geval n = k** (IH);

Nu bewijzen voor **het geval n = k+1 of 1+k**:

$(F)^{1+k+m}M = (F)^{1+(k+m)}M$

$\equiv (F)(F)^{(k+m)}\, M$ (def)

$\equiv (F)(F)^{k}\, (F)^{m}\, M$ *(IH: inductiehypothese)*

$\equiv (F)(F)^{k}M'$ $(M' \equiv (F)^{m}\, M\ )$

$\equiv (F)^{1+k}M'$ (def)

$\equiv (F)^{1+k}\, (F)^{m}\, M$ $(M' \equiv (F)^{m}\, M\ ).$

REKENEN IN λ-CALCULUS CHURCH GETALLEN

Voorstelling natuurlijke getallen:

0 als $\lambda f.\lambda x.x$ → Een functie en een parameter

1 als $\lambda f.\lambda x.(f)x$ → Functie 1 keer toegepast op parameter

2 als $\lambda f.\lambda x.(f)(f)x$ → Functie 2 keer toegepast op parameter

enz.

Definitie

De zogenaamde **Church getallen** c_n ($n \in$ **IN**) worden gedefinieerd als volgt:

$$c_n \equiv \lambda f.\lambda x.(f)^n x$$

Functievoorschrift met 2 argumenten; het 1ste argument (een functie) wordt een aantal keer (n) toegepast op het 2de argument (de parameter voor de functie).

REKENEN IN λ-CALCULUS
λ-DEFINIEERBAAR

Definitie

Een numerieke functie $f : \mathbb{N}^p \rightarrow \mathbb{N}$ met p $\in$ **IN** parameters is λ-definieerbaar als er een combinator *F* bestaat zodat

$$((((F)c_{n1})\ c_{n2})\ \ldots)\ c_{np} =_\beta c_{f(n1,\ n2,\ \ldots,\ np)}$$

m.a.w. er bestaat een combinator waarvan het effect op de Church getallen hetzelfde is als de operatie toegepast op de 'echte' getallen

Voorbeeld: de optelling is λ-definieerbaar als er een lambda expressie (combinator) *plus* bestaat zodat:

$$((\text{plus})\ c_{n1})\ c_{n2}) =_\beta c_{+(n1,\ n2)}$$

bv: $((\text{plus})\ c_2)\ c_3) =_\beta c_5$

REKENEN IN λ-CALCULUS "SUCCESSOR"

$\textbf{\textit{succ}}(n) = n+1$ is λ-definieerbaar

Bewijs:

Zoek $\textbf{\textit{succ}}$ zodat: $(\textbf{\textit{succ}})c_n =_\beta c_{n+1}$

We hebben dus nodig:

$(\textbf{\textit{succ}})c_0 \equiv (\textbf{\textit{succ}})\ \lambda f.\lambda x.x =_\beta \lambda f.\ \lambda x.(f)x \equiv c_1$

$(\textbf{\textit{succ}})c_1 \equiv (\textbf{\textit{succ}})\ \lambda f.\ \lambda x.(f)x =_\beta \lambda f.\lambda x.(f)\ (f)\ x \equiv c_2$

In het algemeen moet (**succ**) dus een extra "aanroep" van f toevoegen:

$$(\textbf{\textit{succ}})c_n =_\beta \lambda f.\ \lambda x.(f)\ voorschrift_n$$

Hoe vinden we nu $voorschrift_n$?

REKENEN IN λ-CALCULUS "SUCCESSOR"

Voorschrift$_n$ vinden:

$$(\boldsymbol{succ})c_n =_\beta \lambda f.\ \lambda x.(f)\ voorschrift_n$$

Merk op dat *voorschrift*$_n$ staat voor $(f)^n x$

en $c_n \equiv \lambda f.\lambda x.(f)^n x$

Dus als we de $\lambda f.\lambda x.$ kunnen wegwerken hebben we het voorschrift:

$$\begin{aligned} ((c_n)\ f)\ x &\equiv ((\lambda f.\lambda x.(f)^n x\ )\ f)\ x \\ &=_\beta (\lambda x.(f)^n x\ )\ x \\ &=_\beta (f)^n x \\ &\equiv voorschrift_n \end{aligned}$$

$$(\lambda x.M)P =_\beta \{P/x\}M$$

Dus $(\boldsymbol{succ})c_n =_\beta \lambda f.\lambda x.(f)\ ((c_n)\ f)\ x$ of $\boldsymbol{succ} \equiv \lambda n.\lambda f.\lambda x.(f)((n)f)x$

VRIJE UNIVERSITEIT BRUSSEL

REKENEN IN λ-CALCULUS "SUCCESSOR"

Dus **succ**(n) = $n+1$ is λ-definieerbaar en

succ $\equiv \lambda n.\lambda f.\lambda x.(f)((n)f)x$

Check (bijvoorbeeld voor c_0):

$$(\mathbf{succ})c_0 \equiv (\lambda n.\ \lambda f.\ \lambda x.(f)((n)f)x)\ c_0$$

$$\equiv (\lambda n.\ \lambda f.\ \lambda x.(f)((n)f)x)\ \lambda f.\ \lambda x.x \qquad (def\ \ c_0)$$

$$= (\lambda n.\ \lambda f.\ \lambda x.(f)((n)f)x)\ \lambda f.\ \lambda x.x$$

$$=_\beta \lambda f.\ \lambda x.(f)((\lambda f.\lambda x.x)f)x \qquad (\beta)$$

$$= \lambda f.\ \lambda x.(f)((\lambda f.\lambda x.x)f)x$$

$$=_\beta \lambda f.\ \lambda x.(f)(\lambda x.x)x \qquad (\beta)$$

$$= \lambda f.\ \lambda x.(f)(\lambda x.x)x$$

$$=_\beta \lambda f.\ \lambda x.(f)\ x \qquad (\beta)$$

$$\equiv c_1$$

REKENEN IN λ-CALCULUS
OPTELLING

Is de optelling λ-definieerbaar?

Ja en de combinator is:

$$\textbf{plus} \equiv \lambda n.\lambda m.\lambda f.\lambda x.((n)f)((m)f)x$$

Intuitief (gebaseerd op definitie van church getallen):

n keer toepassen van f op

m keer toepassen van *f* op *x*

=> in totaal *n*+*m* keer toepassen van *f* op *x*

REKENEN IN λ-CALCULUS OPTELLING

$$\textbf{plus} \equiv \lambda n.\lambda m.\lambda f.\lambda x.((n)f)((m)f)x$$

Bewijs

$$
\begin{aligned}
((\text{plus})\ c_n)\ c_m \equiv\ & ((\lambda n.\ \lambda m.\ \lambda f.\ \lambda x.((n)f)((m)f)x)\ \mathbf{c}_n)\ \mathbf{c}_m && \textit{(def plus)}\\
=_\beta\ & (\lambda m.\ \lambda f.\ \lambda x.((\mathbf{c}_n)f)((m)f)x)\ \mathbf{c}_m && (\beta)\\
\equiv\ & (\lambda m.\ \lambda f.\ \lambda x.((\lambda f.\ \lambda x.(f)^n x)f)((m)f)x)\ \mathbf{c}_m && \textit{(def } c_n\textit{)}\\
=\ & (\lambda m.\ \lambda f.\ \lambda x.((\lambda f.\ \lambda x.(f)^n x)f)((m)f)x)\ \mathbf{c}_m && \\
=_\beta\ & (\lambda m.\ \lambda f.\ \lambda x.(\lambda x.(f)^n x)((m)f)x)\ \mathbf{c}_m && (\beta)\\
=\ & (\lambda m.\ \lambda f.\ \lambda x.(\lambda x.(f)^n x)((m)f)x)\ \mathbf{c}_m && \\
=_\beta\ & (\lambda m.\ \lambda f.\ \lambda x.(f)^n((m)f)x)\ \mathbf{c}_m && (\beta)\\
=\ & (\lambda m.\ \lambda f.\ \lambda x.(f)^n((m)f)x)\ \mathbf{c}_m && \\
=_\beta\ & \lambda f.\ \lambda x.(f)^n((\mathbf{c}_m)f)x && (\beta)\\
\equiv\ & \lambda f.\ \lambda x.(f)^n((\lambda f.\ \lambda x.(f)^m x)f)x && \textit{(def } c_m\textit{)}\\
=\ & \lambda f.\ \lambda x.(f)^n((\lambda f.\ \lambda x.(f)^m x)f)x && \\
=_\beta\ & \lambda f.\ \lambda x.(f)^n(\lambda x.(f)^m x)x && (\beta)\\
=\ & \lambda f.\ \lambda x.(f)^n(\lambda x.(f)^m x)x && \\
=_\beta\ & \lambda f.\ \lambda x.(f)^n(f)^m x && (\beta)\\
\equiv\ & \lambda f.\ \lambda x.(f)^{n+m} x && \text{(Lemma “+”)}\\
\equiv\ & c_{n+m} &&
\end{aligned}
$$

REKENEN IN λ-CALCULUS

Volgend doel: vermenigvuldiging

Eerst lemma

Lemma ("x")

Voor alle n, m $\in$ IN geldt dat

$$\underbrace{((\boldsymbol{c}_n)f)^m}_{\text{m keer na elkaar}} \; y =_\beta (f)^{n \times m} \; y$$

Soort copierfunctie.

REKENEN IN λ-CALCULUS

TB: $((\boldsymbol{c}_n)f)^m\ y =_\beta (f)^{n\mathrm{x}m}\ y$

Bewijs per inductie over m

1. Basis geval m = 0

$$
\begin{aligned}
((\boldsymbol{c}_n)f)^0\ y &\equiv y && \text{(def)}\\
&\equiv (f)^0\ y && \text{(def)}\\
&= (f)^{n\mathrm{x}0}\ y
\end{aligned}
$$

REKENEN IN λ-CALCULUS

Vervolg bewijs $((\boldsymbol{c}_n)f)^m\ y =_\beta (f)^{nxm}\ y$

2. Stel bewezen voor geval m=k (IH),

dan nu bewijs voor het geval m= k+1

$((\boldsymbol{c}_n)f)^{k+1}\ y \equiv ((\boldsymbol{c}_n)f)\ ((\boldsymbol{c}_n)f)^k\ y$

$\equiv ((\boldsymbol{c}_n)f)\ (f)^{n \times k}\ y$ inductiehypothese

$\equiv ((\lambda f.\ \lambda x.(f)^n x)f)\ (f)^{n \times k}\ y$ (def $\boldsymbol{c}_n$)

$=\beta\ (\lambda x.(f)^n x)\ (f)^{n \times k}\ y$ (β)

$=\beta\ (f)^n (f)^{n \times k}\ y$ (β)

$\equiv (f)^{n+n \times k}\ y$ (lemma "+")

$= (f)^{n \times (1+k)}\ y$

REKENEN IN λ-CALCULUS VERMENIGVULDING

Is de vermenigvuldiging λ -definieerbaar?

Ja, en de combinator is ***times*** $\equiv \lambda n.\lambda m.\lambda f.(n)(m)f$

Bewijs

$((\text{times})\ c_n\ )\ c_m$	$\equiv ((\lambda n.\lambda m.\lambda f.(n)(m)f)\ \mathbf{c}_n)\mathbf{c}_m$	*(def times)*
	$=_\beta (\lambda m.\lambda f.(\mathbf{c}_n)(m)f)\mathbf{c}_m$	(β)
	$\equiv (\lambda m.\lambda f.(\lambda f.\lambda x.(f)^n x)(m)f)\mathbf{c}_m$	*(def c_n)*
	$= (\lambda m.\lambda f.(\lambda f.\lambda x.(f)^n x)(m)f)\mathbf{c}_m$	
	$=_\beta (\lambda m.\lambda f.\lambda x.((m)f)^n x)\mathbf{c}_m$	(β)
	$= (\lambda m.\lambda f.\lambda x.((m)f)^n x)\mathbf{c}_m$	
	$=_\beta \lambda f.\lambda x.((\mathbf{c}_m)f)^n x$	(β)
	$= \lambda f.\lambda x.((\mathbf{c}_m)f)^n x$	
	$=_\beta \lambda f.\lambda x.(f)^{m \times n} x$	(Lemma "x")
	$\equiv c_{n \times m}$	

BOOLEAANSE EXPRESSIES IN λ-CALCULUS

$\boldsymbol{true} \equiv \lambda t.\lambda f.t$

$\boldsymbol{false} \equiv \lambda t.\lambda f.f$

true geeft 1ste argument terug;
false geeft 2de argument terug

$((\boldsymbol{true})A)B =_\beta A$

$((\boldsymbol{false})A)B =_\beta B$

Bewijs:

$((\lambda t.\lambda f.t)\ A)\ B =_\beta (\lambda f.A)\ B =_\beta A$

$((\lambda t.\lambda f.f)\ A)\ B =_\beta (\lambda f.f)\ B =_\beta B$

“IF” STATEMENT IN λ-CALCULUS

$\boldsymbol{if} \equiv \lambda c.\lambda d.\lambda e.((c)d)e$

Staat voor: if c then d else e

$\lambda t.\lambda f.t$

c (conditie) moet true of false geven

true => ((c)d)e geeft 1ste argument terug, zijnde d

$\lambda t.\lambda f.f$

false => ((c)d)e geeft 2de argument, zijnde e

$(((\boldsymbol{if})\boldsymbol{true})A)B =_\beta A$

Bewijs:

$\lambda t.\lambda f.t$

$$(((\lambda c.\lambda d.\lambda e.((c)d)e)\ true)A)B \quad =_\beta ((\lambda d.\lambda e.((true)d)e)A)B$$

$$=_\beta (\lambda e.((true)A)e)B =_\beta ((true)A)B =_\beta A$$