

Lösung Übungsblatt Nr. 5 / ALP 1 zur Abgabe 19.12.2003

Aufgabe 1:

K:= "kleiner als" \Leftrightarrow NOT"größer/gleich als"

K:= (NOT)G = (NOT)($\lambda xy.Z(xPy)$)

"größer/gleich als" wurde als Funktion G in der Vorlesung durchgenommen.

1. Es sollen zwei Zahlen a und b übergeben werden.
2. $(b-a) \Leftrightarrow$ es wird a mal der Vorgänger von b gebildet mit Hilfe der Vorgängerfunktion $P = (\lambda n.n\text{Phi}(\lambda z.z0)F)$
2.1 $\text{Phi} = (\lambda pz.z(S(pT))(pT))$ mit Nachfolgerfunktion $S = (\lambda wyx.y(wyx))$
[Anm.: $\text{Phi}(\lambda z.z54) \rightarrow (\lambda z.z65)$]
3. Wenn $(b-a)$ gleich Null ist, waren a und b gleich
- a mal der Vorgänger von $a=b$ ist Null.
Wenn $(b-a)$ kleiner Null ist, war a größer als b
- a mal der Vorgänger von $a=b$ ist Null.
[Anm.: Der Vorgänger von Null ist Null]
Anderen Falls liefert aPb ungleich Null.
4. Nun wird das Ergebnis aus Punkt 3 auf Null getestet. Ist dies der Fall, wird ein True ausgegeben. Ansonsten ein False.
5. Also liefert die Funktion ein False, wenn aus Punkt 4 ein True geliefert wird. Ansonsten False.

V:= "größer als"

Wie die Funktion K, nur dass die zu übergebenen Argumente vertauscht werden.

V:= (NOT)G = (NOT)($\lambda xy.Z(yPx)$), da $(a < b) \Leftrightarrow (b > a)$.

Aufgabe 2:

a)

Def. eines Paares: $(x,y) := (\lambda z.zxy)$

Eine (negative) Zahl $x < 0$ wird als $(0,x)$, bzw. $(\lambda z.z0x)$ definiert und eine (positive) Zahl $x \geq 0$ wird als $(x,0)$, bzw. $(\lambda z.zx0)$ definiert

b)

Die Addition für positive und negative Zahlen wurde in der Vorlesung definiert:

Es werden zwei Paare (x,y) , (a,b) übergeben.

$u := x+a$

$v := y+b$

Addition:

Summe von (x,y) und (a,b) gleich $(u,v,0)$, wenn $u \geq v$,
sonst $(0,v-u)$, wenn $v \geq u$

Subtraktion:

Prinzip: $(x,y) - (a,b) \Leftrightarrow (x,y) + -(a,b) \Leftrightarrow (x,y) + (b,a)$

Um das Paar (a,b) in (b,a) zu invertieren, wird folgende Hilfsfunktion genommen: $D := (\lambda x.(\lambda z.z(xF)(xT)))$.

Erläuterung:

Der Funktion D wird ein Zahlenpaar $(a,b) = (\lambda y.yab)$ übergeben:

$D(\lambda y.yab) \rightarrow (\lambda x.(\lambda z.z(xF)(xT)))(\lambda y.yab)$

$\rightarrow (\lambda z.z((\lambda y.yab)F)((\lambda y.yab)T)) \rightarrow (\lambda z.z(Fab)(Tab))$

$\rightarrow (\lambda z.zba)$, das entspricht (b,a) .

Lösung Übungsblatt Nr. 5 / ALP 1 zur Abgabe 19.12.2003

Aufgabe 3

a)

In Haskell: $[1..3] = (1:(2:(3:[])))$
Im λ -Kalkül: $L :=$ leere Liste; (a,b) entspricht $(\lambda y.yab)$.
Idee: $(1,(2,(3,L)))$
Das entspricht: $(\lambda z.z1(\lambda z.z2(\lambda z.z3L)))$
Also $(\lambda z.z1(\lambda z.z2(\lambda z.z3(\lambda z.z4(\dots(\lambda z.znL)\dots))))$
für eine Liste $[1..n]$.

b)

Eine Liste $(\lambda z.z1(\lambda z.z2(\dots(\lambda z.znL)\dots)))$ wird der Funktion A übergeben,
welche das erste Element extrahiert.
Funktion A := $(\lambda a.aT)$.

Beispiel für die Liste $[1,2,3]$:

$(\lambda a.aT)(\lambda z.z1(\lambda z.z2(\lambda z.z3L))) \rightarrow ((\lambda z.z1(\lambda z.z2(\lambda z.z3L)))T)$
 $\rightarrow (T1(\lambda z.z2(\lambda z.z3L))) \rightarrow 1$

c)

Eine Liste $(\lambda z.z1(\lambda z.z2(\dots(\lambda z.znL)\dots)))$ wird der Funktion B übergeben,
welche das zweite Element extrahiert.

$(\lambda b.bF)$ gibt die Liste ohne erstes Element aus. Danach wird A auf das
Resultat angewandt, so erhält man das erste Element des Restes, also das
zweite Element der gesamten Liste.

Also B := $(\lambda b.A(bF))$

Beispiel für die Liste $[1,2,3]$:

$B(\lambda z.z1(\lambda z.z2(\lambda z.z3L))) \rightarrow (\lambda b.A(bF))(\lambda z.z1(\lambda z.z2(\lambda z.z3L)))$
 $\rightarrow (A((\lambda z.z1(\lambda z.z2(\lambda z.z3L)))F)) \rightarrow (\lambda a.aT)((\lambda z.z1(\lambda z.z2(\lambda z.z3L)))F)$
 $\rightarrow (((\lambda z.z1(\lambda z.z2(\lambda z.z3L)))F)T) \rightarrow ((F1(\lambda z.z2(\lambda z.z3L)))T)$
 $\rightarrow (((\lambda z.z2(\lambda z.z3L)))T) \rightarrow (T2(\lambda z.z3L)) \rightarrow (T2(\lambda z.z3L)) \rightarrow 2$

Zahlen

$0 = (\lambda sz.z)$

$1 = (\lambda sz.sz)$

$2 = (\lambda sz.s(sz))$

....

Wahrheitswerte

$T = (\lambda ab.a)$

$F = (\lambda ab.b)$

$NOT = (\lambda x.xFT)$

$AND = (\lambda xy.xyF)$

Addition [n-mal den Nachfolger von m, also n+m]

$nSm \rightarrow n+m$

Nachfolgerfunktion [S4 \rightarrow 5]

$S = (\lambda wyx.y(wyx))$

Conditionaltest/Nulltest [Z0 \rightarrow T; Zn \rightarrow F mit n != 0]

$Z = (\lambda x.xFNOTF)$

Erzeugung des nächsten Paares [Start mit (0,0); Phi(3,2) \rightarrow (4,3)]

$\Phi = (\lambda pz.z(S(pT))(pT))$

Vorgängerfunktion [P0 = 0; P4 \rightarrow 3]

$P = (\lambda n.n\Phi(\lambda z.z00)F)$

Größenvergleich [Gxy \rightarrow Test: x >= y]

$G = (\lambda xy.Z(xPy))$

Gleichheit

$E = (\lambda xy.AND(Gxy)(Gyx))$