Programmierung

Aufgabe 1 (AGS 12.4.32)

(a) Berechnen Sie die Normalform des untenstehenden λ-Terms, indem Sie ihn *schrittweise* reduzieren. Geben Sie dabei vor jedem Schritt für die relevanten Teilausdrücke die Mengen der gebunden bzw. frei vorkommenden Variablen an.

$$(\lambda fx.ffx)(\lambda y.x)z$$

(b) Gegeben sei der λ-Term

$$\langle F\rangle = \Big(\lambda fxyz.\langle\text{ite}\rangle\;(\langle\text{iszero}\rangle\;(\langle\text{sub}\rangle\;x\,y))\;(\langle\text{add}\rangle\;y\,z)$$
$$\Big(\langle\text{succ}\rangle\;(f\,(\langle\text{pred}\rangle\;x)\,(\langle\text{succ}\rangle\;y)\,(\langle\text{mult}\rangle\;\langle 2\rangle\;z))\Big)\Big)\;.$$

Berechnen Sie schrittweise die Normalform des Terms $\langle Y\rangle\langle F\rangle\langle 6\rangle\langle 5\rangle\langle 3\rangle$. Schreiben Sie für jeden Aufruf von $\langle F\rangle$ jeweils zwei Zeilen: eine in der Sie die Werte der Parameter des Aufrufs protokollieren, und eine in der Sie ihre Auswertung skizzieren. Führen Sie im Rechenprozess zweckmäßige Abkürzungen der λ-Terme ein.

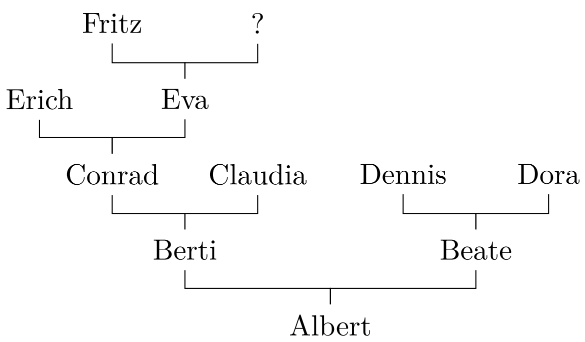
(c) Gegeben sei die folgende Haskell-Funktion:

```
g :: Int -> Int -> Int
g 0 y = 2 * (y + 1)
g x 0 = 2 * (x + 1)
g x y = 4 + g (x - 1) (y - 1)
```

Geben Sie einen λ-Term $\langle G\rangle$ an, so dass `g` $= \langle Y\rangle\langle G\rangle$ gilt. Sie dürfen dabei die in der Vorlesung vorgestellten Terme nutzen.

Aufgabe 2 (AGS 13.1)

Folgender Stammbaum sei gegeben:



(a) Bilden Sie diesen Stammbaum in Prolog ab.

Implementieren Sie folgende Prädikate:

(b) „Vater von“

(c) „Vorfahre von“

(d) „weibliche Vorfahrin von“

https://www.orchid.inf.tu-dresden.de *tobias.denkinger@tu-dresden.de*

Aufgabe 3 (AGS 13.2)

Gegeben ist folgender Prolog-Code.

```
sum(0,X,X).
sum(s(X),Y,s(Z)) :- sum(X,Y,Z).

prod(0,_,0).
prod(s(X),Y,Z) :- prod(X,Y,W), sum(Y,W,Z).
```

Geben Sie eine SLD-Refutation für `?- prod(s(s(0)), s(0), X).` an.

Zusatzaufgabe 1 (AGS 12.4.5)

(a) Berechnen Sie schrittweise die Normalform des λ-Terms $(\lambda zx.z\ x\ (\lambda y.y\ x))\ (\lambda y.z\ x)\ (\lambda z.z)$.

(b) Eine Funktion $f : \mathbb{N}^+ \times \mathbb{N}^+ \to \mathbb{N}^+$ sei wie folgt definiert:

$$\begin{aligned} f(x,y) &= 2 \cdot y && \text{für} \quad x = 1 \\ f(x,y) &= y + (x+1) \cdot f(x-1, x+y) && \text{für alle } x \geq 2 \end{aligned}$$

Geben Sie einen λ-Term $\langle F \rangle$ an, sodass $f = \langle Y \rangle \langle F \rangle$ gilt.

(c) Gegeben sei:

$$\langle G \rangle = \Big(\lambda gxy.\ \langle \text{ite} \rangle\ (\langle \text{iszero} \rangle\ y)\ (\langle \text{succ} \rangle\ x)\ (g\, x\, (\langle \text{pred} \rangle y))\Big)$$

Berechnen Sie $\langle Y \rangle \langle G \rangle \langle 3 \rangle \langle 0 \rangle$. Dokumentieren Sie die Berechnung wie üblich.

Zusatzaufgabe 2 (AGS 12.3.5 ⋆)

Zeigen Sie d.urch Induktion über Listen, dass `map g (ys ++ zs)` = `map g ys ++ map g zs` für alle `g :: a -> b` und `ys, zs :: [a]` gilt. Nutzen Sie dazu die folgenden Definitionen:

```
map :: (a -> b) -> [a] -> [b]
map f [] = []
map f (x:xs) = f x : map f xs

(++) :: [a] -> [a] -> [a]
[] ++ ys = ys
(x:xs) ++ ys = x : (xs ++ ys)
```