Aufgabe 3.3

Die *parametrisierte Primitive Rekursion* h= $\boldsymbol{Pr_t[f,g]}$ ist definiert durch

$$h(x,0) = f(x) \text{ und } h(x,y{+}1) = g(x,y,\ h(t(x),y))$$

Zu zeigen ist, daß h primitiv-rekursiv ist, wenn f, g und t dies sind.

Beweis

Um eine Idee für den Beweis zu finden, analysieren wir zunächst die Funktionswerte $h(x,0), h(x,1), h(x,2), ..h(x,n)$. Es ist

– $h(x,0) = f(x)$
– $h(x,1) = g(x,0,\ h(t(x),0)) = g(x,0,\ f(t(x)))$
– $h(x,2) = g(x,1,\ h(t(x),1)) = g(x,0,\ g(t(x),0,\ f(t(t(x)))))$

$\vdots$

– $h(x,n) = g(x,n,\ h(t(x),n{-}1)) = g(x,n{-}1,\ g(t(x),n{-}2,\ \ldots g(t^{n-1}(x),0,\ f(t^n(x)))\ldots))$

Um also $h(x,n)$ zu berechnen müssen wir die Anwendung von g insgesamt n-mal iterieren und dabei gleichzeitig die Funktion einmal, zweimal, dreimal, und schließlich n-mal auf x anwenden. Dies bedeutet, daß wir im Endeffekt zwei Rekursionen simultan ausführen.
Das Schema der primitiven Rekursion ermöglicht uns aber nur, jeweils eine einzige Rekursion im letzten Argument durchzuführen. Wir müssen also die beiden Rekursionen entkoppeln.
Zu diesem Zweck müssen wir die Funktion h verallgemeinern und eine dreistellige Funktion h' definieren mit der Eigenschaft

– $h'(x,n,y) = g(t^{n-y}(x),y{-}1,\ g(t^{n-(y-1)}(x),y{-}2,\ \ldots g(t^{n-1}(x),0,\ f(t^n(x)))\ldots))$

In dieser Funktion steht der Parameter n für die Anzahl der Iterationen von t und y für die Anzahl der Iterationen von g.

h' ist so aufgeschrieben, daß wir die Rekursion nur im letzten Argument y beschreiben können (also wie eine normale primitive Rekursion) und $t^n(x)$ als Argument hereingeben können. Andererseits läßt sich die gewünschte Funktion h beschreiben durch $h(x,n) = h'(x,n,n)$, wie man am Vergleich der beiden Funktionsgleichungen erkennt.

Im folgenden werden wir zeigen, wie h' tatsächlich als primitiv-rekursive Funktion beschrieben werden kann können und daß wir aus dieser Beschreibung die Gleichung $h(x,n) = h'(x,n,n)$ tatsächlich ableiten können.

Als "Musterlösung" würde der Beweis auf der folgenden Seite vollkommen ausreichen.

Ohne die Erklärungen auf dieser Seite ist jedoch kaum zu verstehen, wie man auf die Definition der Funktion h' kommt.

- Wir "programmieren" zunächst die Funktion $t_1 : \mathbb{N}^2 \to \mathbb{N}$ mit $t_1(x,n) = t^n(x)$.

  Es ist $t_1(x,0) = x = pr_1^1(x)$
  und $t_1(x,n{+}1) = t^{n+1}(x) = t(t^n(x)) = t(t_1(x,n)) = t \circ pr_3^3(x,n,t_1(x,n))$.

  Damit ist das Schema der primitiven Rekursion erfüllt und wir haben $t_1 = Pr[pr_1^1, t \circ pr_3^3]$.

- Mit t_1 können wir nun die Rekursionsgleichung für h' genau ausdrücken. Es ist

  $h'(x,n,0) = f(t^n(x)) = f \circ t_1(x,n)$

  und

  $$\begin{aligned} h'(x,n,y{+}1) &= g(t^{n-(y+1)}(x), y, h'(x,n,y)) = g(t_1(x, n{-}(y{+}1)), y, h'(x,n,y)) \\ &= g \circ (t_1 \circ (pr_1^4, sub \circ (pr_2^4, s \circ pr_3^4)),\ pr_3^4,\ pr_4^4)\ (x,n,y,h'(x,n,y)) \end{aligned}$$

  Damit ist $h' = Pr[f \circ t_1, g \circ (t_1 \circ (pr_1^4, sub \circ (pr_2^4, s \circ pr_3^4)),\ pr_3^4,\ pr_4^4)]$ primitiv-rekursiv.

- Wir zeigen nun, daß $h(x,n) = h'(x,n,n) = h' \circ (pr_1^2, pr_2^2, pr_2^2)(x,n)$ für alle x, n gilt und verwenden hierzu Induktion über n. Es ist

  – $h(x,0) = f(x) = f(t^0(x)) = h'(x,0,0)$

  – $h(x,n{+}1) = g(x,n,\ h(t(x),n)) = g(x,n,\ h'(t(x),n,n))$ nach Induktionsannahme

  – $h'(x,n{+}1,n{+}1) = g(t^{(n+1)-(n+1)}(x), n,\ h'(x,n{+}1,n)) = g(x,n,\ h'(x,n{+}1,n))$

  Wir mussen nun noch beweisen, daß $h'(x,n{+}1,y) = h'(t(x),n,y)$ für alle x, n, y gilt. Wir führen hierzu einen Induktionsbeweis über y.

  Es ist $h'(x,n{+}1,0) = f(t^{n+1}(x)) = f(t^n(t(x))) = h'(t(x),n,0)$

  und $$\begin{aligned} h'(x,n{+}1,y{+}1) &= g(t^{n+1-(y+1)}(x), y,\ h'(x,n{+}1,y)) \\ &= g(t^{n-(y+1)}(t(x)), y,\ h'(t(x),n,y)) = h'(t(x),n,y{+}1) \end{aligned}$$

  Es folgt $h(x,n{+}1) = g(x,n,\ h'(t(x),n,n)) = g(x,n,\ h'(x,n{+}1,n)) = h'(x,n{+}1,n{+}1)$

  und damit ist die Behauptung bewiesen.

- Hieraus folgt schließlich, daß $h = h' \circ (pr_1^2, pr_2^2, pr_2^2)$ ebenfalls primitiv rekursiv ist.