Berechenbarkeit und Komplexität Sommersemester 03

Aufgabenserie 3 – bis 2. 6. 03

ACK) (schriftlich) Die Ackermann-Funktion $A : \mathbb{N}^2 \to \mathbb{N}$ ist definiert durch

$$\begin{aligned} A(0, y) &= y + 1 \\ A(x + 1, 0) &= A(x, 1) \\ A(x + 1, y + 1) &= A(x, A(x + 1, y)) \end{aligned}$$

Diese Funktion ist nicht LOOP-berechenbar (weil sie „zu stark“ wächst).

Zeigen Sie folgende Monotonie-Eigenschaften der Funktion A: (3 Punkte)

- für alle $x, y \geq 0$ gilt: $y < A(x, y) < A(x, y + 1) \leq A(x + 1, y)$
- für alle $x \leq x', y \leq y'$ gilt $A(x, y) \leq A(x', y')$.

(autotool, je ein Punkt) Schreiben Sie primitiv rekursive Programme für die folgenden Funktionen:

- (FUN-SQRT): $x \mapsto \lfloor \sqrt{x} \rfloor$, d. h. Quadratwurzel, abgerundet
- (FUN-FIB): $x \mapsto F(x)$, wobei F die Fibonacci-Funktion $F(0) = 0, F(1) = 1, F(i + 2) = F(i + 1) + F(i)$ ist.
- (FUN-QUIZ) eine zweistellige Funktion mit gegebener Wertetabelle (diese sehen Sie in der autotool-Antwort)

Zur Syntax für FUN-Programme siehe hier:
`http://theo1.informatik.uni-leipzig.de/~autotool/doc/fun.html`.

Sie dürfen (je nach Teilaufgabe) gewisse „eingebaute“ Funktionen (Plus, Minus, Times, Div, Mod) benutzen. Einzelheiten erfahren Sie in der Rückmeldung vom autotool.

DPR) (schriftlich) Einige Funktionen sind induktiv durch Bezug auf *zwei* frühere Werte definiert.

Wir benutzen dazu den Operator DPR mit $f = \mathsf{DPR}(g_0, g_1, h)$ gdw.

$$\begin{aligned} f(x_1, \ldots, x_n, \quad 0 \quad) &= g_0(x_1, \ldots, x_n) \\ f(x_1, \ldots, x_n, \quad 1 \quad) &= g_1(x_1, \ldots, x_n) \\ f(x_1, \ldots, x_n, \quad y + 2 \quad) &= h(x_1, \ldots, x_n, f(x_1, \ldots, x_n, y), f(x_1, \ldots, x_n, y + 1)) \end{aligned}$$

Wie kann man die Fibonacci-Funktion mit DPR (und Grundfunktionen) darstellen? (1 Punkt)

Beweisen Sie, daß man den Operator DPR durch primitiv-rekursive Operatoren und Grundfunktionen darstellen kann. (3 Punkte)

Hinweis: benutzen Sie ähnliche Methoden wie bei simultaner Rekursion (siehe Skript/Buch).
