Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl.-Ing. Christoph Baumann

Theoretische Informatik
7. Übungsblatt
(Abgabe: 19.12.2012)

Aufgabe 1: (Primitiv Rekursive Funktionen) (3+3+3+5+5 Punkte)

Zeige, dass folgende Funktionen primitiv rekursiv (PR) sind! Führe dazu die Funktionen mit Hilfe des Kompositions- und des primitiven Rekursionsschemas aus der Definition der PR Funktionen auf bereits bekannte primitiv rekursive Funktionen zurück!

a) Das beschränkte Produkt $f : \mathbb{N}_0^{r+1} \to \mathbb{N}_0$ für eine primitiv rekursive Funktion $g : \mathbb{N}_0^{r+1} \to \mathbb{N}_0$:

$$f(x_1, \ldots, x_r, m) = \prod_{i=0}^{m} g(x_1, \ldots, x_r, i)$$

b) Die Fakultätsfunktion $F(x) = x!$.

c) Die beschränkte universelle Quantifizierung $Q : \mathbb{N}_0^{r+1} \to \mathbb{B}$ eines primitiv rekursiven Prädikates $q : \mathbb{N}_0^{r+1} \to \mathbb{B}$:

$$Q(x_1, \ldots, x_r, m) = \forall i \leq m.\ q(x_1, \ldots, x_r, i)$$

d) Die Funktion $p : \mathbb{N}_0 \to \mathbb{N}_0$, welche die n-te Primzahl berechnet, also $p(0) = 2, p(1) = 3, \ldots$.
Hinweis: Für jede Primzahl p gilt, dass die nächsthöhere Primzahl nicht größer als $p! + 1$ ist.

e) Das Prädikat $prim : \mathbb{N}_0 \to \mathbb{B}$, welches 0 zurückgibt, wenn n eine Primzahl ist, also:

$$prim(n) = \begin{cases} 0 & : \text{ n ist eine Primzahl} \\ 1 & : \text{ n ist keine Primzahl} \end{cases}$$

Aufgabe 2: (Turingmaschinen) (6 Punkte)

Entwickle eine 1-Band-Turingmaschine, die für alle Wörter $w \in \{0, 1\}^*$ auf Eingabe $w\#$ mit der Ausgabe $w\#w$ terminiert. Wie sich die Turing-Maschine bei anderen Eingaben verhält, ist egal.

Es gibt eine 1-Band-Turingmaschine mit 5 Zuständen, die obige Spezifikation erfüllt. Du solltest nicht wesentlich mehr Zustände benutzen.

Aufgabe 3: (Satz von Church) (5+5+5 Punkte)

In der Vorlesung wurde der Satz von Church behandelt, welcher besagt, das alle Turing-berechenbaren Funktionen auch μ-rekursiv sind. Dabei wurde eine Funktion $\Psi : (A \cup Z)^+ \to \mathbb{N}_0$ eingeführt, die Konfigurationen einer Turingmaschine mit Bandalphabet A, Zustandsmenge Z und Endzuständen $E \subseteq Z$ als natürliche Zahl kodiert. Sei $p = \#(A \cup Z)$ die Mächtigkeit des Bandalphabets und der Zustandsmenge.

$$\forall a \in A \cup Z.\ \Psi(a) \neq 0, \text{ injektiv}$$
$$\Psi(b_1, \ldots, b_{|b|}) \equiv \sum_{j=1}^{|b|} \Psi(b_j) \cdot (p+1)^{|b|-j}$$

Darauf aufbauend wurden dann μ-rekursive Funktionen definiert mit dem Ziel, Berechnungen von M durch die Auswertung einer Funktion zu simulieren.

Löse dazu die folgende Aufgaben:

a) Sei $x = \Psi(b)$ die Kodierung einer Konfiguration $b \in (A \cup Z)^+$. Die Funktionen zur Berechung der Kodierungen der Präfixe und Suffixe von b mit der Länge i, bzw. $|b| - i + 1$, wurden wie folgt definiert:

$$prefix(x, i) \equiv \lfloor x/(p+1)^{L(x)-i} \rfloor \qquad suffix(x, i) \equiv Rest(x, (p+1)^{L(x)-i+1})$$

Dabei ist $L(x) = |b|$ die Länge des kodierten Konfigurationsstrings. Zeige:

$$prefix(x, i) = \Psi(b_1, \ldots, b_i) \qquad suffix(x, i) = \Psi(b_i, \ldots, b_{|b|})$$

b) Für eine kodierte Konfiguration C von M ist die Laufzeit bis zu Halten, falls M hält, durch die Funktion $T(C)$ gegeben. In der Vorlesung wurde diese mit Hilfe der Übergangsfunktion Δ^n und Zustandsfunktion z wie folgt definiert.

$$T(C) \equiv \min\{j \mid z(\Delta^j(C)) \in \Psi(E)\} \quad \text{falls existiert}$$

Zeige, dass $T(C)$ μ-rekursiv ist!