Technische Universität Chemnitz
Professur Theoretische Informatik
Prof. Dr. A. Goerdt, Dipl.-Inf. F. Schädlich, Dipl.-Inf. A. Lanka

Sommersemester 2002

10. Übung Theoretische Informatik II

Aufgabe 1

Beschäftigen Sie sich mit der Äquivalenz von Turingmaschinen und Random-Access-Maschinen (RAM, vgl. Theoretische Informatik I). Zeigen Sie hierzu zunächst folgenden Satz:

Es existiert ein Polynom q, so daß jede im logarithmischen Kostenmaß $t(n)$-zeitbeschränkte Random-Access-Maschine (RAM) durch eine $O\Big(q\big(n+t(n)\big)\Big)$-zeitbeschränkte Turingmaschine simuliert werden kann.

Aufgabe 2

Was berechnet die Rekurrenz

$$f(n) = \begin{cases} f(f(n+11)) & \text{falls } n \leq 100 \\ n-10 & \text{falls } n > 100 \end{cases} \quad ?$$

Aufgabe 3

Entwickeln Sie aus den LOOP-Programmen der Aufgaben 9.2.g) (x-te Fibonacci-Zahl) und 9.2.h) ($x!$) simultan primitiv rekursive Funktionsdefinitionen.

Aufgabe 4

Zeigen Sie, daß die Funktion

$$\text{prim}(p) = \begin{cases} 1 & \text{falls } p \text{ eine Primzahl ist} \\ 0 & \text{sonst} \end{cases}$$

primitiv rekursiv ist.

Aufgabe 5

Betrachten Sie die Simulation einer Turingmaschine M mittels eines GOTO-Programms P. Beschreiben Sie die Umwandlung der nachfolgenden δ-Anweisungen in entsprechende Programmstücke:

a) $\delta(z_i, a_k) = (z_l, a_m, R)$

b) $\delta(z_i, a_k) = (z_l, a_m, N)$