Technische Universität Berlin **SS 2013**
Fakultät II, Institut für Mathematik
Sekretariat MA 5–2, Frau Kiefer

Prof. Dr. Martin Skutella
Torsten Gellert
Martin Groß
Dr. Max Klimm

Aylin Acikel, Katharina Bütow, Christian Döblin, Alexander Hopp, Daniel Kuske, Olivia Röhrig, Robert Rudow, Daniel Schmand, Hendrik Schrezenmaier, Mona Setje-Eilers, Judith Simon, Sebastian Spies, Fabian Wegscheider, Jan Zur

10. Übungsblatt Computerorientierte Mathematik II

Abgabe: 18.06.2013 (bis 14:15 Uhr in MA 004)

1. Aufgabe (3 + 4 Punkte)

Betrachtet die folgende Turingmaschine $M := (Q, \Sigma, \Gamma, q_0, \delta, F)$ mit der Menge der Zustände $Q := \{q_0, q_1, q_2\}$, dem Alphabet $\Sigma := \{0, 1\}$, dem Bandalphabet $\Gamma := \Sigma \cup \{B\}$, dem Leerzeichen B, den akzeptierenden Endzuständen $F := \{q_2\}$ und der in der folgenden Tabelle angegebenen Übergangsfunktion $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, N, R\}$:

δ	0	1	B
q_0	$(q_1, 1, N)$	$(q_1, 0, N)$	(q_2, B, N)
q_1	$(q_0, 0, R)$	$(q_0, 1, R)$	$-$
q_2	$-$	$-$	$-$

Waagerechte Striche bedeuten dabei, dass die Turingmaschine stoppt.

(a) Gebt für die Eingabe “00110” die Konfigurationsfolge der Turingmaschine M an.

(b) Beschreibt informal, was von M berechnet wird und was die Turingmaschine in den einzelnen Zuständen tut.

2. Aufgabe (3 + 6 Punkte)

Sei das Alphabet $\Sigma := \{0, 1, \#\}$ und die Sprache

$$L := \{bin(n)\#w \in \Sigma^* \mid n \in \mathbb{N}, w \in \{0,1\}^*, |w| \leq n\}$$

gegeben. Dabei bezeichnet $|\cdot|$ die Länge eines Wortes und $bin(n)$ die eindeutige Binärdarstellung einer natürlichen Zahl n.

(a) Gebt eine Turingmaschine an, die eine als Binärzahl aufgefasste Eingabe aus $\{0, 1\}^n$ um 1 verringert. Führende Nullen sollen durch Leerzeichen ersetzt werden. Gebt außer der formalen Definition der Turingmaschine eine kurze Beschreibung der Zustände an, so dass die Funktionsweise deutlich wird. Ihr könnt davon ausgehen, dass die Eingabe ungleich 0 ist und keine führenden Nullen hat.

(b) Beschreibt eine 2-Band-Turingmaschine mit einer Laufzeit von $\mathcal{O}(n)$, die genau die Wörter aus L akzeptiert. Das heißt, die Maschine endet genau für die Eingaben in einem akzeptierenden Zustand, wenn die Eingabe in L enthalten ist.

Erklärt nur die grobe Funktionsweise der Maschine, ihr braucht nicht die Zustände und Übergangsfunktion zu spezifizieren. Benutzt die Turingmaschine aus (a) als Unterprogramm.

Zeigt, dass die von euch entworfene Maschine in $\mathcal{O}(n)$ läuft.

Hinweis: Bei der Analyse könnt ihr nutzen, dass die Reihe $\sum_{i=1}^{\infty} i2^{-i}$ gegen 2 konvergiert.

3. Aufgabe (1 + 1 + 1 + 1 Punkte)

Es seien A und B zwei Probleme, wobei sich A durch folgenden Algorithmus lösen lässt:

Input: Eingabe x für Problem A der Größe n
Output: Lösung $A(x)$
Rufe einen Algorithmus auf, der $y := B(x)$ berechnet
Berechne aus y und x eine Lösung für A in Laufzeit $\mathcal{O}(n^3)$

Die Komplexität eines Problems ist definiert als die Laufzeit eines schnellsten Algorithmus für das Problem. (Die Komplexität des Sortierproblems basierend auf paarweisen Vergleichen ist z. B. $\Theta(n \log n)$). Beantwortet folgende Fragen und begründet eure Antwort:

(a) Was lässt sich über die Komplexität von A sagen, wenn wir einen Algorithmus für B mit Laufzeit $\mathcal{O}(n^2)$ kennen?

(b) Was lässt sich für die Komplexität von A schließen, wenn wir wissen, dass jeder Algorithmus für B exponentielle Laufzeit hat?

(c) Was lässt sich für die Komplexität von B schließen, wenn wir einen Algorithmus für A mit Laufzeit $\mathcal{O}(n^2)$ kennen?

(d) Was lässt sich über die Komplexität von B sagen, wenn wir wissen, dass jeder Algorithmus für A mindestens exponentielle Laufzeit hat?