Otto-von-Guericke Universität Magdeburg
Institut für Mathematische Optimierung
Wintersemester 2012/2013
Prof. Volker Kaibel, Jun.-Prof. Gennadiy Averkov

1. Übung zur Vorlesung
Komplexitätstheorie

1. Aufgabe

Beschreiben Sie eine Turingmaschine M, die für eine Eingabe $x \in \{0,1\}^*$ die Anzahl der Nullen und Einsen in x vergleicht. Die Ausgabe soll 1 sein, falls x mehr Einsen als Nullen enthält. Die Ausgabe ist 0, falls x mehr Nullen als Einsen enthält. Die Ausgabe ist eine leere Zeichenkette, falls x genau so viele Nullen wie Einsen enthält.

Geben Sie eine obere Schranke an die Anzahl der Schritte an, die Ihre Maschine M auf einer Eingabe der Länge $n \in \mathbb{N}$ macht.

2. Aufgabe

Beschreiben Sie eine Turingmaschine M, die für eine gegebene Zeichenkette $x \in \{0,1\}^*$ die Anzahl der zusammenhängenden Null-Bereiche in x berechnet (z.B.: 01110011000 enthält 3 Null-Bereiche der Längen 1, 2 und 3; 00100011010000111 enthält 4 Null-Bereiche der Längen $2, 3, 1$ und 4; 1111 enthält keinen Null-Bereich). Die Ausgabe soll im Unärsystem kodiert werden. D.h., wenn x genau k Null-Bereiche enthält, wird die Zeichenkette 1^k ausgegeben.

Geben Sie eine obere Schranke an die Anzahl der Schritte Ihrer Maschine bei einer Eingabe der Länge $n \in \mathbb{N}$ an.

3. Aufgabe

(a) Beschreiben Sie eine Turingmaschine, die eine Binärzahl in das Unärsystem konvertiert. Für eine eingegebene 0/1-Folge $b_0, \dots, b_m$ ($m \geq 0$) soll die Ausgabe 1^k sein, wobei $k = b_0 2^0 + \cdots + b_m 2^m$.

(b) Beschreiben Sie umgekehrt eine Turingmaschine, die eine Unärzahl in das Binärsystem konvertiert. Für eine gegebene endliche Einsenfolge 1^k mit $k \geq 0$ ist die Ausgabe $b_0, \dots b_m$, mit $m \geq 0$, $b_i \in \{0,1\}$ für alle $i \in \{0, \dots, m\}$ und $k = b_0 2^0 + \cdots + b_m 2^m$.

4. Aufgabe

Bestimmen Sie für jedes der unten angegebenen Funktionenpaare f, g, ob $f = o(g)$, $g = o(f)$, oder $f = \Theta(g)$ gilt:

(a) $f(n) = n^2$, $g(n) = 2n^2 + 100\sqrt{n}$.

(b) $f(n) = n^{100}$, $g(n) = 2^{n/100}$.

(c) $f(n) = n^{100}$, $g(n) = 2^{n^{1/100}}$.

(d) $f(n) = \sqrt{n}$, $g(n) = 2^{\sqrt{\log n}}$.

(e) $f(n) = n^{100}$, $g(n) = 2^{(\log n)^2}$.

(f) $f(n) = 1000n$, $g(n) = n \log n$.