Aufgaben zu Kapitel 3 der Vorlesung „Randomisierte Algorithmen“

Aufgabe 3.1
Gegeben sei eine probabilistische TM M (nicht notwendigerweise in Normalform), die für jede Eingabe hält.

Konstruieren Sie eine probabilistische TM M', die für keine Eingabe mehr Schritte macht als M schlimmstenfalls es tut und in jeder Situation genau zwei verschiedene Möglichkeiten der Weiterarbeit hat oder keine.

Lösung 3.1
Als neue Zustandsmenge wählt man $S' = S \cup \bar{S}$, wobei $\bar{S} = \{\bar{s} \mid s \in S\}$ „neue Kopien“ der ursprünglichen Zustände sind.

Falls für ein $(s, b) \in S \times B$ durch δ schon zwei verschiedene Aktionen festgelegt sind, ändert man nichts: $\delta'(s, b) = \delta(s, b)$.

Falls dagegen nur eine Aktion festgelegt ist, also $\delta(s, b) = \{(t, c, d)\}$ ergänzt Man eine weitere, indem man definiert:

$$\delta'(s, b) = \{(t, c, d), (\bar{t}, c, d)\}$$

Damit sind für jedes $(s, b) \in S \times B$ zwei Aktionen festgelegt.

Für jedes $\bar{s} \in \bar{S}$ legt man nun noch fest, dass sich der Zustand „wie das Original s verhält“. Für jedes $(\bar{s}, b) \in \bar{S} \times B$ definiert man:

$$\delta'(\bar{s}, b) = \delta'(s, b)$$

Aufgabe 3.2
Unter welchen Umständen und wie könnte man die PTM M' aus der vorangegangenen Aufgabe so ergänzen, dass eine PTM M'' in Normalform entsteht?

Lösung 3.2
Das verbleibende Problem: alle Berechnungspfade müssen gleich lang sein.

Falls M' so ist, dass man in Abhängigkeit von der Eingabegröße n die maximale Länge $t(n)$ endlicher Berechnungspfade in Zeit $t(n)$ berechnen kann, dann kann M'' für eine Eingabe w zunächst $x = t(|w|)$ berechnen, anschließend M' Schritt für Schritt simulieren und dabei jeweils x um 1 herunterzählen. Wenn M' fertig ist, aber noch $x > 0$, dann werden noch künstlich “Leerschritte“ von M' simuliert, bis $x = 0$ erreicht ist und erst mit der Ausgabe von M' angehalten.

Die resultierende TM hat schlimmstenfalls Zeitbedarf $t(n)\log t(n)$. Insbesondere arbeitet mit M' auch M'' in Polynomialzeit.

Aufgabe 3.3
Beweisen Sie:

$$\sum_{i=0}^{k} i \cdot 2^{-i} = 2 - (k+2)2^{-k}$$

Lösung 3.3
Vollständige Induktion liefert sofort das Gewünschte.

*** Aufgabe 3.4**
Versuchen Sie zu zeigen, dass das folgende Problem unentscheidbar ist:

Probleminstanz: eine probabilistische Turingmaschine M

Frage: Wird jedes Eingabewort von M entweder gar nicht oder mit einer Wahrscheinlichkeit von mindestens $1/2$ akzeptiert?