Lehrstuhl für Informatik 1
Prof. Dr. Gerhard Woeginger
Jan Böker, Tim Hartmann

WS 2018/19

Übung zur Vorlesung
Berechenbarkeit und Komplexität

Lösung Blatt 1

Hausaufgabe 1.1 **(2 + 2 Punkte)**

Geben Sie je eine formale Darstellung für die Sprachen der folgenden Entscheidungsprobleme an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe und zum Eingabealphabet.

(a) **Eine unabhängige Menge (engl.: independent set) in einem Graphen $G = (V, E)$ ist eine Menge $U \subseteq V$ von Knoten, so dass keine zwei verschiedenen Knoten $u, v \in U$ benachbart sind. Die Sprache des Independent-Set-Problems $L_{\text{Indep-Set}}$ enthalte die Kodierungen aller Paare (G, b) mit $b \in \mathbb{N}$, so dass G eine unabhängige Menge der Größe mindestens b besitzt.**

Sei $A = (a_{i,j}) \in \{0,1\}^{n\times n}$ die Adjazenzmatrix zu G, d.h. $a_{i,j} = 1$ genau dann, wenn $\{i, j\} \in E$ ist, für $0 \leq i, j < n$. Wir kodieren G als Aneinanderreihung der Zeilen von A. Dann ist die Sprache des Independent-Set-Problems wie folgt gegeben.

$$L_{\text{Indep-Set}} = \{a_{1,1} \dots a_{n,n} \# \operatorname{bin}(b) \mid n, b \in \mathbb{N}, \exists K \subseteq \{1, \dots, n\} : \\ (|K| \geq b) \land (\forall i, j \in K, i \neq j : a_{i,j} = 0)\}.$$

(b) **Das Bin-Packing-Problem besteht darin, zu entscheiden, ob n Objekte mit Größen $w_1, \dots, w_n \in \{1, \dots, B\}$ in γ Kisten der Größe B gepackt werden können, wobei alle Werte natürliche Zahlen sind. Die Sprache des Bin-Packing-Problems $L_{\text{Bin-Packing}}$ enthalte die Kodierungen der Tupel $(w_1, \dots, w_n, B, \gamma)$ mit $w_1, \dots, w_n \in \{1, \dots, B\}$, für die eine Verteilung der n Objekte auf die γ Kisten möglich ist.**

$$L_{\text{Bin-Packing}} = \{\operatorname{bin}(w_1)\# \dots \# \operatorname{bin}(w_n)\# \operatorname{bin}(B)\# \operatorname{bin}(\gamma) \mid w_1, \dots, w_n, B, \gamma \in \mathbb{N}, \\ \exists \text{ Partition } P \text{ von } \{1, \dots, n\} : |P| \leq \gamma \land \forall S \in P : \sum_{i \in S} w_i \leq B\}$$

Hausaufgabe 1.2 **(2 + 2 Punkte)**

In der Vorlesung „Turing-Maschinen I“ wurde auf den Folien 30–33 eine Turing-Maschine M mit den acht Zuständen $q_0, \dots, q_6, \bar{q}$ für die Sprache

$L = \{0^n1^n \mid n \geq 1\}$ diskutiert. Wir konstruieren eine neue Turing-Maschine M', deren Definition mit der von M in allen Details übereinstimmt, mit der einzigen Ausnahme, dass wir in der Überführungsfunktion

$$\boldsymbol{\delta(q_5, 1) = (q_6, 1, R)}$$

durch

$$\boldsymbol{\delta(q_5, 1) = \textbf{reject}}$$

ersetzen.

(a) **Geben Sie an, welche Sprache L' von der Turing-Maschine M' akzeptiert wird.**

$$L' = \{0^n1^n \mid n \geq 1\} = L$$

(b) **Begründen Sie Ihre Behauptung.**

Die Turing-Maschine wechselt in den Zustand q_5, sobald die linkeste 0 und die rechteste 1 gelöscht wurden. Wenn der Schreib-/Lesekopf in diesem Zustand nun auf einer 1 steht, dann enthält die Eingabe weniger 0en als 1en, d.h., die Eingabe ist nicht in L, und wir können bereits an dieser Stelle verwerfen. An dieser Stelle würde M noch bis zum Ende der Eingabe und wieder zurück zum Anfang laufen und erst dann verwerfen.

Hausaufgabe 1.3 (4 Punkte)

Geben Sie eine textuelle Beschreibung des Verhaltens der folgenden Turingmaschine M an. Geben Sie zusätzlich die von M berechnete Funktion an.

$$\boldsymbol{M = (\{q_0, q_1, q_2, q_3, q_4, \bar{q}\}, \{0, 1\}, \{0, 1, B\}, B, q_0, \bar{q}, \delta)}$$

δ	0	1	B
q_0	$(q_1, 0, R)$	$(q_2, 1, R)$	**reject**
q_1	$(q_1, 0, R)$	$(q_1, 1, R)$	(q_3, B, L)
q_2	$(q_2, 0, R)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	**reject**	**accept**	**reject**
q_4	**accept**	**reject**	**reject**

Zu Beginn merkt sich die Turingmaschine das erste Zeichen und geht entsprechend in Zustand q_1 oder q_2. Dann läuft M nach rechts bis zum Ende des Wortes und überprüft in Zustand q_3 bzw. q_4 das letzte Zeichen.

Die Turingmaschine M akzeptiert, wenn das erste und das letzte Zeichen verschieden sind. Andernfalls verwirft sie die Eingabe.

Das heißt, die Turingmaschine M berechnet

$$f : \{0,1\}^* \to \{0,1\}$$

$$f(w) = \begin{cases} 1, & \text{falls } w = w_1 \dots w_n \text{ mit } w_i \in \{0,1\} \text{ und } w_1 \neq w_n \\ 0, & \text{sonst} \end{cases}$$