Karlsruher Institut für Technologie
Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

Übungsblatt 3

Vorlesung Theoretische Grundlagen der Informatik im WS 19/20

Ausgabe: 19. November 2019
Abgabe: 3. Dezember 2019, 11:00 Uhr (im Kasten im UG von Gebäude 50.34)

Aufgabe 1

(4 + 2 = 6 Punkte)

Gegeben sei das Alphabet $\Sigma = \{\mathtt{0}, \mathtt{1}\}$.

(a) Geben Sie das Bandalphabet sowie den Zustandsgraphen einer deterministischen Turing-Maschine an, die für eine Eingabe $w \in \Sigma^*$ das Spiegelwort w^R berechnet. Die Eingabe `11010` wird also z.B. zu `01011` transformiert.

(b) Geben Sie die Konfigurationsfolge an, die bei Eingabe des Wortes `10` durchlaufen wird.

Aufgabe 2

(1 + 1 + 1 = 3 Punkte)

Betrachten Sie die folgenden Entscheidungsprobleme:

(a) Gegeben eine natürliche Zahl n, ist n *keine* Primzahl?

(b) Gegeben eine Menge $M = \{m_1, \ldots, m_k\}$ von natürlichen Zahlen, gibt es eine Teilmenge $M' \subseteq M$, sodass die Summe von M' und die Summe von $M \setminus M'$ gleich sind?

(c) Gegeben ein ungerichteter Graph $G = (V, E)$, ist G *nicht* zusammenhängend?

Geben Sie für jedes dieser Probleme eine Ja-Instanz und eine Nein-Instanz an. Beschreiben Sie außerdem, wie eine NTM arbeitet, die das Problem in polynomieller Zeit löst. Geben Sie dazu insbesondere an, wie die NTM den Lösungsvorschlag des Orakelmoduls interpretiert und was der deterministische Teil der NTM tun muss, um den Lösungsvorschlag zu überprüfen.

Aufgabe 3

(3 + 2 + 3 = 8 Punkte)

Eine normale Turing-Maschine darf ihren Lese-/Schreibkopf bei jedem Übergang höchstens um ein Feld nach links oder rechts bewegen. In dieser Aufgabe betrachten wir *Sprung-Turing-Maschinen* (STM), die bei der Kopfbewegung Felder überspringen dürfen. Zusätzlich zur endlichen Kontrolle hat eine STM einen Zähler für die *Sprungweite* $v \in \mathbb{N}$. Diese gibt an, wie viele Felder der Kopf bei einer Bewegung überspringt. Bei einem Übergang mit Kopfbewegung nach links (bzw. rechts) geht der Kopf also v Felder nach links (bzw. rechts) statt nur eins.

Anfangs gilt $v = 1$, d.h. die STM verhält sich wie eine normale TM. Bei jedem Übergang hat die STM jedoch die Möglichkeit, die Sprungweite um 1 zu erhöhen oder zu verringern. Formal hat die Überführungsfunktion einer STM dann folgende Form:

$$\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{R, L, N\} \times \{-, 0, +\}$$

Dabei steht „$-$“ für „Verringere Sprungweite um 1“, „+“ für „Erhöhe Sprungweite um 1“ und „0“ für „keine Änderung“. Wir fordern außerdem, dass die Sprungweite nie kleiner als 1 wird.

Wir betrachten nun das Alphabet $\Sigma = \{\mathtt{0}, \mathtt{1}\}$ und die Sprache $L = \{w \in \Sigma^* \mid \exists k \in \mathbb{N} : w_0 = w_k = w_{2k} = \mathtt{1}\}$. Zum Beispiel gilt: $\mathtt{1011001} \in L$ mit $k = 3$, aber $\mathtt{100101} \notin L$.

(a) Beschreiben Sie, wie man L mit einer normalen TM entscheiden kann. Sie dürfen mehrere Spuren verwenden, aber nicht mehrere Bänder. Geben Sie die Laufzeit Ihrer TM im $\mathcal{O}$-Kalkül an.

(b) Zeigen Sie, dass eine STM L in Linearzeit erkennen kann.

(c) Zeigen Sie, dass STMs und normale TMs gleich mächtig sind.

Aufgabe 4

(1 + 1 + 1 + 1 = 4 Punkte)

Zeigen oder widerlegen Sie:

(a) Das Komplement des Halteproblems ist semi-entscheidbar.

(b) Das Komplement der Diagonalsprache ist semi-entscheidbar.

(c) Seien L_1 und L_2 semi-entscheidbare Sprachen. Dann ist auch $L_1 \cup L_2$ semi-entscheidbar.

(d) Seien L_1 und L_2 semi-entscheidbare Sprachen. Dann ist auch $L_1 \setminus L_2$ semi-entscheidbar.

Aufgabe 5

(2 + 3 = 5 Punkte)

Eine Turing-Maschine M zählt eine unendliche Sprache L auf, wenn M niemals stoppt und eine Liste $w_1, w_2, \ldots$ genau der Wörter aus L ausgibt. Dabei ignoriert M die Eingabe und die Wörter der ausgegebenen Liste sind eindeutig voneinander getrennt. Für die Reihenfolge der Wörter in der Liste muss gelten, dass jedes Wort aus L nach endlich vielen Schritten ausgegeben wird. Eine unendliche Sprache L ist aufzählbar, falls eine Turing-Maschine existiert, die L aufzählt.

(a) Zeigen Sie, dass L genau dann entscheidbar ist, wenn L in kanonischer Reihenfolge[1] aufzählbar ist.

[1] Siehe Vorlesung 6, Folie 19. Die kanonische Reihenfolge sortiert Wörter nach aufsteigender Länge und Wörter der gleichen Länge lexikographisch.

(b) Zeigen Sie, dass L genau dann semi-entscheidbar ist, wenn L aufzählbar ist. Erklären Sie auch, warum sich hier im Gegensatz zu Aufgabenteil (a) nicht fordern lässt, dass die Reihenfolge der Aufzählung kanonisch ist.

Aufgabe 6 (2 Punkte)

Zeigen Sie, dass die Sprache $L_{\text{write}} = \{wva \mid T_w \text{ schreibt bei Eingabe } v \text{ das Symbol } a \text{ auf das Band}\}$ nicht entscheidbar ist. Benutzen Sie dafür nicht den Satz von Rice!

Aufgabe 7 (1 + 2 + 1 = 4 Punkte)

In der Übung wurde das Äquivalenzproblem für Turing-Maschinen vorgestellt:

$$L_{\text{äq}} = \{w\#v \in \{0, 1, \#\}^* \mid L(T_w) = L(T_v)\}$$

In dieser Aufgabe sollen Sie zeigen, dass weder $L_{\text{äq}}$ noch $L_{\text{äq}}^c$ semi-entscheidbar ist. Gehen Sie dazu wie folgt vor:

(a) Zeigen Sie, dass das Komplement der universellen Sprache L_u^c nicht semi-entscheidbar ist.

(b) Zeigen Sie, dass $L_{\text{äq}}$ nicht semi-entscheidbar ist. Nehmen Sie dazu an, es gäbe eine Turing-Maschine $M_{\text{äq}}$, die $L_{\text{äq}}$ akzeptiert. Konstruieren Sie daraus eine Turing-Maschine M_u^c, die L_u^c akzeptiert.

(c) Zeigen Sie mit der gleichen Herangehensweise wie in Aufgabenteil (b), dass $L_{\text{äq}}^c$ nicht semi-entscheidbar ist.