Prof. Dr. Sebastian Siebertz Sommersemester 2024

Theoretische Informatik 2

Blatt 01

Abgabe bis zum 22.04.2024

---

1. 30 Punkte Sei $\Sigma = \{a, b\}$, $\Gamma = \{a, b, \sqsubset, \sqcup\}$ und M die Turingmaschine $(\{q_s, q_1, q_a, q_r\}, \Sigma, \Gamma, \sqsubset, \sqcup, \delta, q_s, q_a, q_r)$ mit

$$\begin{aligned} \delta(q_s, a) &= (q_s, a, +1), & \delta(q_s, b) &= (q_1, a, +1), \\ \delta(q_s, \sqcup) &= (q_a, \sqcup, 0), & \delta(q_1, a) &= (q_1, a, 0), \\ \delta(q_1, b) &= (q_1, b, +1), & \delta(q_1, \sqcup) &= (q_a, \sqcup, 0) \end{aligned}$$

sowie für alle $q \in \{q_s, q_1\}$ ist $\delta(q, \sqsubset) = (q, \sqsubset, +1)$ und für alle $x \in \Gamma$ ist $\delta(q_a, x) = (q_a, x, 0)$,$\delta(q_r, x) = (q_r, x, 0)$.

- Gebt eine terminierende Berechnung von M auf der Eingabe *abb* an.
- Welche Sprache erkennt M? Begründet eure Antwort.

Vermerk zu den zwei folgenden Aufgaben: Benutzt für die Lösung das online Tool `https://turingmachine.io/` und gebt auch die yml Datei mit ab. Kommentiert den Code und verwendet sinnvolle Zustandsnamen. Siehe `tm-same-amount-of-As-and-Bs.yml` im StudIP für ein Beispiel.

2. 35 Punkte Für ein Wort $w = a_1 \cdots a_n$ über $\Sigma = \{0, 1\}$ bezeichne $w^R = a_n \cdots a_1$ das *Spiegelwort* von w. Insbesondere definieren wir $\varepsilon^R = \varepsilon$.

Gebt eine Turingmaschine M mit $L(M) = \{w \in \Sigma^* \mid w = w^R\}$ an.

Erklärt die Vorgehensweise eurer Turingmaschine.

3. 35 Punkte Für eine Zahl $n \in \mathbb{N}_0$ schreiben wir $\mathrm{bin}(n)$ für die Binärdarstellung von n über dem Alphabet $\{0, 1, \#\}$. Gebt eine Turingmaschine in grafischer Notation an, die zu der Eingabe $\mathrm{bin}(n)\#\#\mathrm{bin}(m)$, wobei $n \geq m$, die Ausgabe $\mathrm{bin}(n + m)$ berechnet.

Erklärt die Vorgehensweise eurer Turingmaschine.

Hinweis: Ihr dürft eine Binärkodierung annehmen, bei der das signifikanteste Bit rechts steht, d.h. $\mathrm{bin}(2) = 01$ (wir lesen also die Binärzahlen von rechts nach links). Das erste # Zeichen kann benutzt werden um Platz für einen möglichen Übertrag zu schaffen.

Die folgenden Aufgaben werden im Tutorium besprochen und müssen nicht zur Korrektur abgegeben werden.

4. Gebt eine Turingmaschine in grafischer Notation an, die folgende Sprache akzeptiert:

$$L = \left\{ w \in \{a, b\}^* \mid |w|_a = |w|_b \right\}$$

Hierbei bezeichnet $|w|_a$ die Anzahl der a in w.

5. Gebt eine nicht terminierende deterministische TM M in grafischer Notation an, die auf leerer Eingabe alle Wörter über $\Sigma = \{a, b, c\}$ aufzählt, d.h. es gibt einen Zustand $q_f \in Q$, sodass

(a) für alle $w \in \Sigma^*$ M irgendwann die Konfiguration $(q_f, [\, w \sqcup \sqcup \cdots, |w|)$ erreicht,

(b) wann immer M eine Konfiguration (q_f, v, k) erreicht ist $v = [\, w \sqcup \sqcup \cdots$ und $k = |w|$ für ein $w \in \Sigma^*$.

6. Wiederholt die Beweise zu den folgenden Aussagen:

(a) Die Typ 3 Sprachen sind unter Komplement ($^-$), Vereinigung ($\cup$) und Schnitt ($\cap$) abgeschlossen.

(b) Die Typ 2 Sprachen sind unter Vereinigung ($\cup$), Konkatenation ($\cdot$) und Kleene-Stern (*) abgeschlossen.

(c) Die Typ 2 Sprachen sind *nicht* unter Schnitt ($\cap$) und *nicht* unter Komplement ($^-$) abgeschlossen.

7. Wiederholt die Algorithmen für das Wortproblem und das Leerheitsproblem für Typ 3 Sprachen (präsentiert als endlicher Automat) und Typ 2 Sprachen (präsentiert als kontextfreie Grammatik).