Theoretische Informatik 2

Sommersemester 2018

Elizaveta Kovalevskaya
Georg Schnitger
Philipp Tertel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN
Institut für Informatik
AG Theoretische Informatik

Übungsblatt 1

Ausgabe: 17.04.18
Abgabe: 24.04.18

- Solange die Aufgabenformulierung nichts anderes besagt, wird **stets** eine sorgfältige, mathematisch fundierte Begründung der Antwort erwartet.
- Alle Übungsblätter besitzen 24 Punkte als Maximalpunktzahl.

Aufgabe 1.1 *Die Tiefe von DFAs und regulären Sprachen* (2+2+2+2 Punkte)

Im Folgenden betrachten wir ausschließlich Automaten ohne nicht-erreichbare Zustände. Sei $L = L(A)$ für einen DFA $A = (\Sigma, Q, \delta, q_0, F)$. Definiere die *Tiefe von A* als die kleinste Zahl i, so dass $\equiv_A^i$ und $\equiv_A^{i+1}$ übereinstimmen.

a) Zeigen Sie, dass die Tiefe eines DFAs mit der Tiefe seines Äquivalenzklassenautomaten übereinstimmt.

   Kommentar: Die Tiefe eines DFAs A hängt also nur von seiner Sprache $L(A)$ ab. Demgemäß wird die *Tiefe einer regulären Sprache L* durch die Tiefe irgendeines DFA definiert, der L akzeptiert.

b) Zeigen Sie: Eine Sprache L besitzt genau dann die Tiefe 0, wenn L einen DFA mit höchstens zwei Zuständen besitzt.

c) Zeigen Sie, dass es keinen DFA A mit n Zuständen und Tiefe größer als $n - 2$ gibt.

d) Die Sprache L sei regulär. Zeigen Sie, dass die Tiefe von L höchstens $\text{Index}(L) - 2$ beträgt.

Aufgabe 1.2 *Die Algorithmen von Moore und Hopcroft* (6+10 Punkte)

a) In der $(i + 1)$-ten Iteration des Algorithmus von Moore sind zuerst die Zerlegungen $Z_i(a)$ der Zustandsmenge Q für jeden Buchstaben $a \in \Sigma$ zu bestimmen und danach die Zerlegung Z_{i+1}.

   i) Zeigen Sie, wie $Z_i(a)$ in Zeit $\mathcal{O}(|Q|)$ bestimmt werden kann, wenn Z_i bekannt ist.

   ii) Zeigen Sie, wie Z_{i+1} in Zeit $\mathcal{O}(|\Sigma| \cdot |Q|)$ bestimmt werden kann, wenn die Zerlegungen $Z_i(a)$ für alle Buchstaben $a \in \Sigma$ bekannt sind.

   Um eine Zerlegung von Q in k Klassen darzustellen, verwenden Sie die Zahlen $0, \ldots, k-1$ als Klassenkennungen und weisen Sie jedem Zustand die Kennung seiner Klasse zu.

   Hinweis: Radixsort sortiert n natürliche Zahlen $0 \leq a_1, \ldots, a_n \leq n^r - 1$ in Zeit $O(r \cdot n)$.

Bitte wenden!

b) Sei $\Sigma = \{a, b\}$ und sei $w = w_1 \cdots w_n \in \Sigma^n$. Wir definieren

$$L_w = \{u \in \{a,b\}^* \,:\, w \text{ ist ein Teilwort von } u\}.$$

Sei $A_w = (\Sigma, Q, \delta, q_0, F)$ ein DFA mit $Q = \{0, \ldots, n\}$, $q_0 = 0$ und $F = \{n\}$.

i) Definieren Sie die Zustandsüberführungsfunktion δ so, dass $L(A_w) = L_w$ gilt.

ii) Führen Sie den Algorithmus von Moore auf dem DFA A_w aus.

- Bestimmen Sie die Tiefe T_w von A_w.
- Beschreiben Sie die Äquivalenzklassen $\equiv^i_{A_w}$ für jedes $i \leq T_w$.

iii) Führen Sie den Algorithmus von Hopcroft auf dem DFA A_w aus.

- Wie viele Iterationen werden im Algorithmus von Hopcroft ausgeführt?
- Für jede Iteration: Welche Äquivalenzrelation wurde erreicht? Welche Äquivalenzklassen befinden sich in der Menge *Check* und aus welchen Zuständen bestehen sie?

iv) Die *Vielfachheit* $v(q, c)$ eines Paars $(q, c) \in Q \times \Sigma$ ist die Anzahl der verschiedenen Checkmengen Y, zu denen der Übergang $(q, c) \mapsto \delta(q, c)$ gehört[1]. Wie groß kann $v(q, c)$ höchstens werden?

v) Bestimmen Sie die Laufzeiten der Algorithmen von Moore und Hopcroft für den DFA A_w. Sie können annehmen, dass jede Iteration des Algorithmus von Moore lineare Zeit benötigt und dass die Laufzeit des Algorithmus von Hopcroft asymptotisch nach oben beschränkt ist durch

$$\sum_{(q,c) \in Q \times \Sigma} v(q, c).$$

[1]Ein Übergang *gehört* zu einer Checkmenge Y genau dann, wenn $\delta(q, c) \in Y$.