Karlsruher Institut für Technologie

Institut für Angewandte Informatik und Formale Beschreibungsverfahren

Lösung zur Bonusklausur über den Stoff der Vorlesung „Grundlagen der Informatik II“ (45 Minuten)

Name: ____________________ Vorname: ____________________

Matr.-Nr.: ____________________ Semester: __________ (WS 2018/19)

Ich bestätige, dass ich die folgenden Angaben gelesen und mich von der Vollständigkeit dieses Klausurexemplars überzeugt habe (Seiten 1-6).

____

Unterschrift des o. g. Klausurteilnehmers bzw. der o. g. Klausurteilnehmerin

Anmerkungen:

1. Legen Sie bitte Ihren Studierendenausweis bereit.
2. Bitte tragen Sie **Name**, **Vorname** und **Matr.-Nr.** deutlich lesbar ein.
3. Die folgenden **3 Aufgaben** sind vollständig zu bearbeiten.
4. Folgende Hilfsmittel sind zugelassen: **keine.**
5. Täuschungsversuche führen zum Ausschluss von der Klausur.
6. Unleserliche oder mit Bleistift geschriebene Lösungen können von der Klausur bzw. Wertung ausgeschlossen werden.
7. Die Bearbeitungszeit beträgt 45 Minuten.

Nur für den Prüfer :

1	2	3	-	-	-	-	-	-	-	-	-	-	-	-	-	gesamt

Aufgabenübersicht

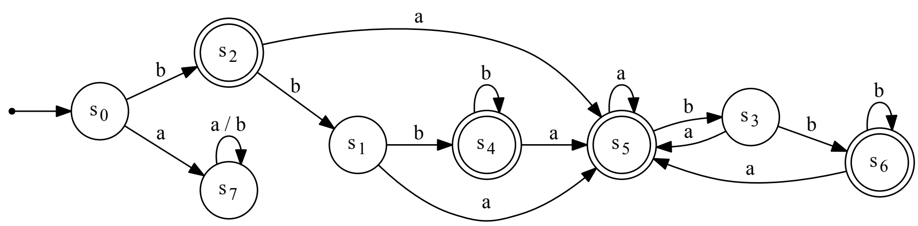
Aufgabe 1

Minimierung endlicher Automaten

Gegeben sei der folgende endliche Automat A mit

$$A = (\{a, b\}, \{s_0, \ldots, s_7\}, \delta, s_0, \{s_2, s_4, s_5, s_6\})$$

δ:



und die zugehörige Zustandsübergangstabelle.

δ	a	b
s_0	s_7	s_2
s_1	s_5	s_4
s_2	s_5	s_1
s_3	s_5	s_6
s_4	s_5	s_4
s_5	s_5	s_3
s_6	s_5	s_6
s_7	s_7	s_7

Minimieren Sie A mit dem Algorithmus aus der Vorlesung. Geben Sie die Minimierungstabelle und den minimierten Automaten A' vollständig an.

Lösung:

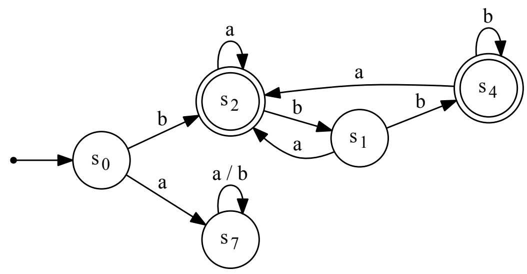
Es ergibt sich folgende Minimierungstabelle, aus der hervorgeht, dass s_1, s_3 und s_2, s_5 und s_4, s_6 zueinander jeweils äquivalent sind.

s_1	$\times_1$						
s_2	$\times_0$	$\times_0$					
s_3	$\times_1$	$-$	$\times_0$				
s_4	$\times_0$	$\times_0$	$\times_1$	$\times_0$			
s_5	$\times_0$	$\times_0$	$-$	$\times_0$	$\times_1$		
s_6	$\times_0$	$\times_0$	$\times_1$	$\times_0$	$-$	$\times_1$	
s_7	$\times_1$	$\times_1$	$\times_0$	$\times_1$	$\times_0$	$\times_0$	$\times_0$
	s_0	s_1	s_2	s_3	s_4	s_5	s_6

Fasst man die äquivalenten Zustände zusammen, ergibt sich folgender minimierter Automat:

$$A' = (\{a, b\}, \{s_0, s_1, s_2, s_4, s_7\}, \delta', s_0, \{s_2, s_4\})$$

δ':



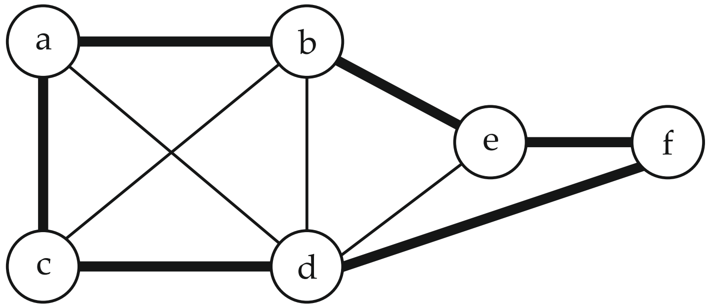
mit dieser Zustandsübergangstabelle:

δ'	a	b
s_0	s_7	s_2
s_1	s_2	s_4
s_2	s_2	s_1
s_4	s_2	s_4
s_7	s_7	s_7

Aufgabe 2

Komplexitätstheorie

Das *ungerichtete Hamiltonkreisproblem (UHP)* ist die Frage, ob in einem ungerichteten Graphen $G = (V, E)$ ein geschlossener Pfad existiert, der jeden Knoten genau einmal enthält. Im abgebildeten Graphen kennzeichnet der dick gezeichnete Kantenzug einen Hamiltonkreis.



(a) Wie könnte eine nichtdeterministische Turingmaschine vorgehen, um das UHP für einen gegebenen Graphen **effizient** zu lösen?

Hinweis: Argumentieren Sie umgangssprachlich; Sie müssen die Turingmaschine **nicht** angeben.

Lösung: Die Turingmaschine könnte eine Teilmenge der Kanten $E' \subseteq E$ des Graphen raten, und in Polynomialzeit überprüfen, ob E' einen zusammenhängenden und geschlossenen Pfad darstellt, der jeden Knoten $v \in V$ genau einmal enthält. Alternativ könnte die TM auch eine Permutation der Knoten raten und überprüfen, ob sie sich alle auf einem geschlossenen Kantenzug befinden.

(b) Wie könnte eine deterministische Turingmaschine vorgehen, um das Vorgehen der nichtdeterministischen Turingmaschine bei der Lösung des UHP zu simulieren?

Hinweis: Sie müssen auch hier die Turingmaschine **nicht** angeben.

Lösung: Die det. Turingmaschine könnte nacheinander die verschiedenen Teilmengen $E' \subseteq E$ der Menge der Kanten des Graphen auf ihrem Band auflisten und jeweils den (deterministischen) Verifikationsprozess der nichtdeterministischen Turingmaschine durchführen.

(c) Wie groß ist der Zeitaufwand Ihrer nichtdeterministischen Turingmaschine, bzw. Ihrer deterministischen Turingmaschine in Abhängigkeit von der Knotenanzahl des Graphen $n = |V|$ im schlimmsten Fall?

Lösung:

- Nichtdet: Das Raten läuft in konstanter Zeit c ab, das Verifizieren in Polynomialzeit n^k für ein konstantes k, also ergibt sich ein Gesamtaufwand von $O(c + n^k) = O(n^k)$. (Auch wenn der Verifikationsprozess polynomiell in der Anzahl der Kanten (anstatt der Knoten) ist, wäre der Gesamtaufwand $O((n^2)^k) = O(n^2k) = O(n^{k'})$ für eine andere Konstante k'.)

- Det: Der Auflistungsprozess beinhaltet alle Teilmengen der Kantenmenge E, also $p(E) = 2^{|E|}$ Elemente. Die Kantenmenge enthält maximal $\frac{n^2-n}{2}$ Elemente, also ist $p(E) \leq 2^{\frac{n^2-n}{2}}$. Für jedes dieser Elemente muss der Verifikationsprozess durchgeführt werden. Also ergibt sich ein Aufwand von $O(2^{\frac{n^2-n}{2}} \cdot n^k) = O(2^{n^2})$. Zu beachten ist, dass diese Zeitangabe zwar überexponentiell in der Anzahl der Knoten wächst, aber immer noch „nur“ exponentiell in der Gesamtlänge der Eingabe ist, da die Kanten auch Teil der Eingabe sind.

Aufgabe 3

Pumping-Lemma für EA-Sprachen

Gegeben sei die Sprache

$$L = \{a^{2^i} \mid i \in \mathbb{N}_0\}.$$

Es gilt also $a, aa, aaaa, aaaaaaaa, \ldots \in L$ und bspw. $\lambda, aaa, aaaaaa \notin L$.

Zeigen Sie mithilfe des Pumping-Lemmas für EA-Sprachen, dass L nicht vom Chomsky-Typ 3 ist.

Lösung:

- Angenommen, es existiert ein EA mit n Zuständen, der L akzeptiert.
- Sei $w = a^{2^n} \in L$ mit $|w| = 2^n \geq n$ für $n \in \mathbb{N}_0$.
- Laut PPL existieren x, y, z mit $w = xyz$ und
  (a) $|xy| \leq n$,
  (b) $|y| \geq 1$ und
  (c) $\forall i \in \mathbb{N}_0 : xy^iz \in L$.
- Sei also $xy = a^k, y = a^l, x = a^{k-l}, z = a^{2^n-k}$.
- Nach (3) sollte mit $i = 2$ gelten:

  $$w' = \underbrace{a^{k-l}}_{x}\underbrace{a^{2l}}_{2y}\underbrace{a^{2^n-k}}_{z} = a^{2^n+l} \in L \text{ mit } |w'| = 2^n + l.$$

  Das ist aber ein Widerspruch: nach (1) und (2) gilt $1 \leq l \leq n$, also auch:

  $$2^n < 2^n + l < 2^{n+1}.$$

  Damit gilt $|w'| = a^{2^n+l} \notin L$ (denn $2^n + l$ ist keine Zweierpotenz).

  (Begründung analog zu Heim-Übungsblatt 1, Aufgabe 4(a).)
- Da wir zu einem Widerspruch gekommen sind, muss die Annahme, L sei von einem EA erkennbar, falsch sein.