Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2021 – Hausaufgabenblatt 2

Abgabe: 03.05.2021, 12:00 CEST

- Beachten Sie die Abgabemodalitäten auf der Vorlesungswebsite!
- Sei $\Phi := \{\{1\},\{2,3\},\{4\},\{5,6\},\{7\}\}$. Nach dem Abgabedatum werden wir für jede Menge $A \in \Phi$ eine zufällige Aufgabe $a \in A$ wählen und korrigieren.
- Es werden diese Aufgaben korrigiert: H2.1, H2.3, H2.4, H2.6, H2.7
- Wenn Sie einen Beweis aufstellen, von dem Sie wissen, das einzelne Schritte problematisch oder unvollständig sind, merken Sie dies bitte in Ihrer Lösung an, damit wir das bei der Korrektur positiv berücksichtigen können.

Aufgabe H2.1. (*DFA/NFA Konstruktion*) 0.5+0.5+0.5+0.5 Punkte

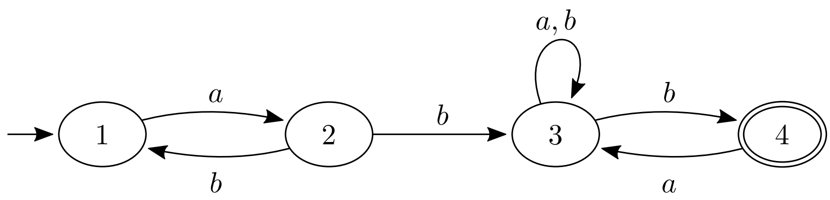
Diese Hausaufgabe wird mit Automata Tutor bearbeitet und abgegeben. Falls Sie es noch nicht bereits gemacht haben, folgen Sie den Schritten in Ü1.2, um ein Konto zu erstellen. Achten Sie darauf, dass Sie sich, wie dort beschrieben, mit Ihrer TUM-Kennung anmelden. Ansonsten können wir Ihnen die Punkte nicht gutschreiben.

Bearbeiten Sie die Hausaufgaben H2.1 (a–d). **Achtung:** Während Sie für die Aufgaben aus dem Übungsblatt beliebig viele Versuche hatten, haben Sie für jede Hausaufgabe nur 5 Versuche. Sie bekommen nur dann einen Punkt, wenn Sie die Aufgabe nach 5 Versuchen vollständig (also mit 10/10 Punkten) gelöst haben.

Aufgabe H2.2. (*Potenzmengenkonstruktion*) 3 Punkte

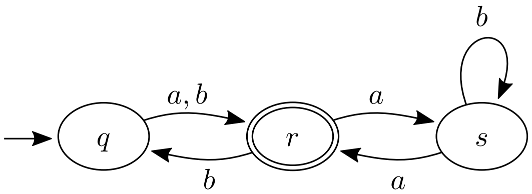
Konvertieren sie den folgenden NFA über dem Alphabet $\Sigma = \{a, b\}$ zu einem DFA, indem Sie die Potenzmengenkonstruktion aus der Vorlesung verwenden.

Hinweis: Es genügt, nur die vom Startzustand erreichbaren Zustände zu konstruieren.

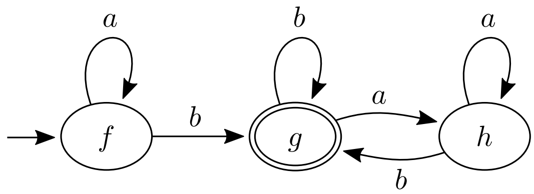


Aufgabe H2.3. (*Produktkonstruktion*) 2+1 Punkte

Zwei DFAs M_1, M_2 sind wie folgt definiert.



M_1



M_2

(a) Konstruieren Sie einen DFA M mit $L(M) = L(M_1) \cap L(M_2)$, indem Sie die Produktkonstruktion verwenden.

(b) Konstruieren Sie einen DFA M' mit $L(M') = L(M_1) \cup L(M_2)$. Sie dürfen ihr Ergebnis aus Teilaufgabe (a) wiederverwenden – in dem Fall genügt es, zu beschreiben, wie Sie es anpassen.

Aufgabe H2.4. (*Fangfrage*) 2+1 Punkte

Sei $\Sigma := \{\mathtt{a}, ..., \mathtt{z}\}$ und $L := \{w \in \Sigma^* : w \text{ enthält } \mathtt{theo}\}$. Die Sprache L besteht also aus genau den Wörtern, die `theo` als Teilwort besitzen, z.B. `apotheose`, `pantheon`, oder `theologie`.

Für einen DFA bezeichnen wir einen Zustand als *Fangzustand*, wenn er keinen Endzustand erreichen kann. (Es kann mehr als einen Fangzustand geben.)

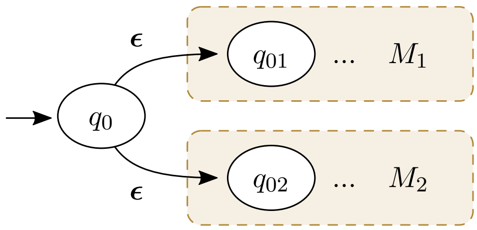
(a) Zeigen Sie, dass jeder DFA, der $\overline{L}$ akzeptiert, einen Fangzustand besitzt.

(b) Zeigen Sie, dass in keinem DFA, der L akzeptiert, ein vom Startzustand aus erreichbarer Fangzustand existiert.

Aufgabe H2.5. (*Limited Power*) 1+3 Punkte

Dora ist traurig. Heute morgen im Kindergarten wurde sie von einem Schmetterling abgelenkt und hat deswegen nicht aufgepasst. Jetzt möchte sie die Vereinigung von zwei deterministischen endlichen Automaten berechnen, weiß aber nicht, wie die Produkt-Konstruktion funktioniert. Sie muss die Vereinigung also mithilfe von ϵ-NFAs erzeugen und hat Angst, dass die dann notwendige Determinisierung über Potenzmengenkonstruktion eine exponentielle Vergrößerung des Zustandsraumes nach sich zieht. Können Sie Dora trösten?

Sei $\Sigma := \{a, b\}$ ein Alphabet und seien zwei beliebige DFAs $M_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ und $M_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$ gegeben. Wir definieren einen ϵ-NFA $M = (Q, \Sigma, \delta, q_0, F)$ formal als $Q := Q_1 \uplus Q_2 \uplus \{q_0\}$, $\delta := \delta_1 \uplus \delta_2 \uplus \{(q_0, \boldsymbol{\epsilon}, q_{01}), (q_0, \boldsymbol{\epsilon}, q_{02})\}$ und $F := F_1 \uplus F_2$.[1] Der NFA M sieht also folgendermaßen aus:



(a) Wenden Sie das Verfahren aus der Vorlesung an, um M zu einem NFA M' zu konvertieren, indem Sie geeignet die ε-Kanten durch neue Kanten ersetzten. Beschreiben Sie das Ergebnis kurz.

(b) Zeigen Sie, dass das Anwenden der Potenzmengenkonstruktion aus der Vorlesung auf M' einen DFA M'' mit höchstens $\mathcal{O}(|Q_1| \cdot |Q_2|)$ Zuständen ergibt. Beachten Sie, dass wir – wie üblich – nur die aus $\{q_0\}$ erreichbaren Zustände konstruieren.

Aufgabe H2.6. (*Sparmaßnahmen*) 1+3 Punkte

[1] Die Notation $A \uplus B$ entspricht $A \cup B$, fordert aber zusätzlich, dass A und B disjunkt sind.

Die Universitätsleitung zeigt sich entsetzt über die große Menge an Tinte, die in die fettgedruckten Symbole $\boldsymbol{\epsilon}$ und $\boldsymbol{\emptyset}$ in regulären Ausdrücken fließt. Deshalb wird angeordnet, dass diese Symbole künftig – wenn möglich – vermieden werden. Jeder reguläre Ausdruck soll in eine der folgenden Formen gebracht werden:

(F1) r (F2) $r \mid \boldsymbol{\epsilon}$ (F3) $\boldsymbol{\epsilon}$ (F4) $\boldsymbol{\emptyset}$

Hierfür ist r ein beliebiger regulärer Ausdruck, der weder $\boldsymbol{\epsilon}$ noch $\boldsymbol{\emptyset}$ enthält.

Beruhigen Sie ihre verzweifelten Kollegen, indem Sie beweisen, dass jeder Ausdruck in diese Form gebracht werden kann.

(a) Beweisen Sie $(r \mid \boldsymbol{\epsilon})^* \equiv r^*$ für jeden regulären Ausdruck r.

(b) Zeigen Sie, dass zu jedem regulären Ausdruck r' ein äquivalenter Ausdruck r existiert, der in einer der obigen Formen ist.

Hinweis: Für (b) können Sie strukturelle Induktion verwenden.

Bonusaufgabe H2.7. (*Passwortsuche*) 1+2 Bonuspunkte

Nach dem jüngsten Hackerangriff hat die IT der Universität beschlossen, neue Sicherheitsvorkehrungen zu treffen. Passwörter müssen nun die folgenden Regeln erfüllen:

- Ein Passwort besteht aus Ziffern $Z := \{0, ..., 9\}$, Kleinbuchstaben $K = \{a, ..., z\}$, Großbuchstaben $G := \{A, ..., Z\}$, und Sonderzeichen $S := \{+, -, \%, \$, /, \#, \sim, !\}$.
- Jedes Passwort hat mindestens 3 und höchstens 5 Zeichen.
- Nach jeder Ziffer kommt ein Sonderzeichen oder das Wortende.
- Es dürfen keine zwei Sonderzeichen aufeinander folgen.
- Das Passwort muss mit einem Buchstaben oder einer Ziffer anfangen und enden.
- Ein Sonderzeichen darf nicht auf einen Buchstaben folgen.
- Nach einem Großbuchstaben darf keine Ziffer stehen.
- Vor einem Großbuchstaben darf kein Kleinbuchstabe stehen.

Die letzten sechs Regeln beziehen sich hierbei immer auf die Zeichen *unmittelbar* danach oder davor. So darf z.B. nur direkt vor einem Großbuchstaben kein Kleinbuchstabe stehen, aber *r0*$\$E$ wäre erlaubt. Sie sollen nun die Stärke dieses Systems einschätzen, indem Sie zählen, wie viele Passwörter erlaubt sind.

(a) Bestimmen Sie zunächst die Anzahl der möglichen Passwörter über dem vereinfachten Alphabet $\Sigma' := \{Z, K, G, S\}$. Hierzu ersetzen wir jeden Kleinbuchstaben durch K, jeden Großbuchstaben durch G, usw. Aus dem Passwort *Te0!1* würde also *GKZSZ* werden.

(b) Erweitern Sie nun ihr Ergebnis aus (a) und zählen sie die Passwörter, die den obigen Regeln genügen.

Sowohl bei (a) als auch bei (b) sind wir an Verfahren interessiert, die sich verallgemeinern ließen, und z.B. auch Passwörter mit Länge 20 zählen könnten. Alle Möglichkeiten durchzugehen ist hierfür nicht hinreichend. Geben Sie sowohl bei (a) als auch bei (b) Ihren vollständigen Rechenweg an. Sie dürfen einen Taschenrechner verwenden, ansonsten lösen Sie die Aufgabe bitte – wie immer – von Hand.

Hinweis: Auch die Bonusaufgaben beziehen sich auf die Vorlesungsinhalte und lassen sich mit diesen lösen. (Es mag auch andere Lösungen geben.)