Prof. Javier Esparza
Kevin Kappelmann, Simon Roßkopf, Lukas Stevens

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2023 – Hausaufgabenblatt 8

- Bitte beachten Sie, dass in dieser Vorlesung generell Antworten mit Begründung gefordert werden, solange die Aufgabe nicht explizit das Gegenteil sagt.
- Zum Bestehen diese Blattes müssen Sie 50% der Punkte erreichen.
- Update: Es wird die Aufgabe 2 korrigiert.

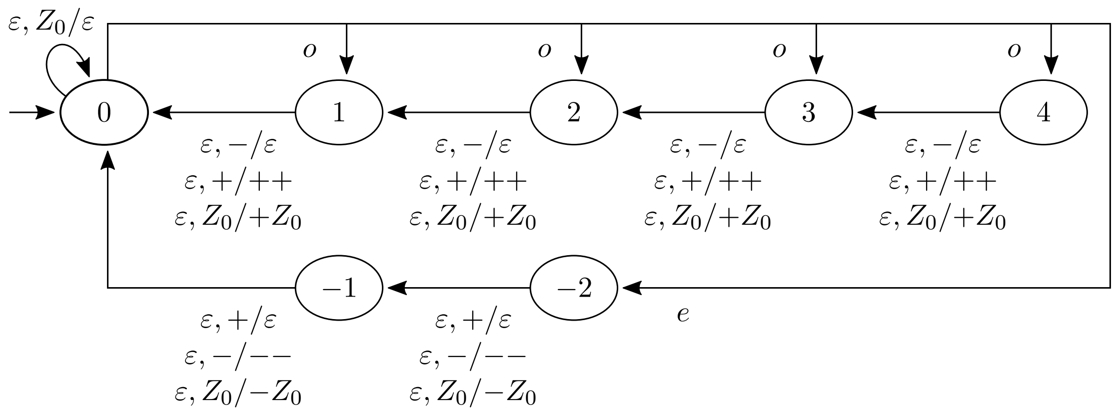
AT-Aufgabe H8.1. (*Based Machine*) 0 Punkte

Bearbeiten Sie diese Aufgabe auf Automata Tutor.

Sei $\Sigma := \{e, o\}$. Wir betrachten $L := \{w \in \Sigma^* : \frac{1}{2}|w|_o \leq |w|_e \leq 2|w|_o\}$. Konstruieren Sie einen PDA für L. Ihr Automat soll über leeren Keller akzeptieren.

Hinweis: Die Musterlösung hat 7 Zustände, ist also relativ groß. Es kann sinnvoll sein, Ihren Automaten zunächst auf Papier oder auf `https://automatonsimulator.com/` auszuarbeiten, und ihn dann nach AT zu übertragen.

Lösungsskizze.



Anmerkung: Es ist auch möglich, diese Aufgabe mit weniger Zuständen zu lösen.

Aufgabe H8.2. (*Hast du den Schleifentrick erst raus...*) 5 Punkte

Beim Tanzen auf dem StuStaCulum wurde Dr. Evilsparza ganz schön schwindelig. Eigentlich hat er sich gefreut, entspannt zu guter Jazzmusik zu swingen. Doch die wilden Studierenden haben kurzum einen Pogo gestartet. Das war schon ziemlich wild: es war fast so, als hätte sich die ganze Welt wie eine Schleife um seinen Kopf gedreht. Das fand Dr. Evilsparza nicht lustig, denn ihm wurde dabei ziemlich übel. Das möchte er den Studierenden jetzt natürlich heimzahlen! Am nächsten Morgen begab sich Dr. Evilsparza also in seine Werkstube, um eine wild rotierende Maschine zu erfinden, die alle Studierenden den Kopf verdrehen soll. Er nennt seine Maschine den *Schleifomat*.

Ein Schleifomat M ist ähnlich zu einem Kellerautomat, aber bevorzugt es seinen Speicher in einer Schleife (bzw. streng genommen einer Queue) anstatt in einem Stapel zu speichern. Formal ist ein Schleifomat ein 6-Tupel $(Q, \Sigma, \Gamma, q_0, Z_0, \delta)$, wobei Q die Zustandsmenge, Σ das

Eingabealphabet, Γ das Schleifenalphabet, $q_0 \in Q$ der Startzustand, $Z_0 \in \Gamma$ der initiale Schleifeninhalt und $\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \to \mathcal{P}(Q \times \Gamma^*)$ die Übergangsfunktion ist. Die Übergänge eines Schleifomats können graphisch genauso dargestellt werden, wie die eines Kellerautomaten.

Eine Konfiguration eines Schleifomatens ist ein Tripel $(q, w, \gamma) \in Q \times \Sigma^* \times \Gamma^*$, wobei q der aktuelle Zustand, w das noch zu lesende Wort und γ der aktuelle Schleifeninhalt ist. Die Übergangsrelation $\to_M$ ist dann wie folgt definiert:

$$(q, aw, Z\gamma) \to_M (q', w, \gamma\gamma') \quad :\overset{\text{Def.}}{\Longleftrightarrow} \quad (q', \gamma') \in \delta(q, a, Z), \quad \text{wobei } a \in \Sigma \cup \{\epsilon\}$$

Gilt also beispielsweise $(q', CD) \in \delta(q, a, A)$, dann gilt $(q, abc, AB) \to_M (q', bc, BCD)$.

Die mit leerer Schleife akzeptierte Sprache eines Schleifomats ist definiert als

$$L_\epsilon(M) := \{w \in \Sigma^* \mid \exists q \in Q.\ (q_0, w, Z_0) \to_M^* (q, \epsilon, \epsilon)\}.$$

Dr. Evilsparza möchte euch nun so richtig überraschen: Er glaubt nämlich, dass seine Schleifomaten mächtiger sind als normale Kellerautomaten. Deswegen fordert er euch heraus, für die folgende *nicht kontextfreie* Sprache (vgl. H6.2 (c)) einen Schleifomaten zu konstruieren:

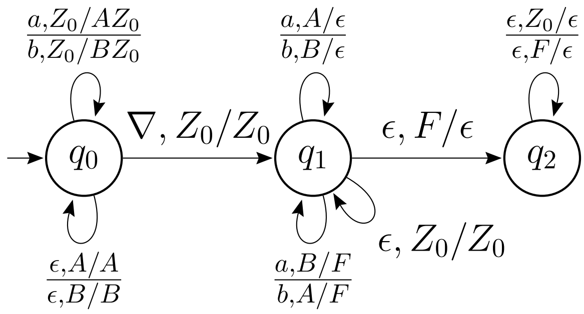
$$L := \{u\nabla v \mid u, v \in \{a, b\}^*, |u| = |v|, u \neq v\}$$

Beschreiben Sie zum erfolgreichen Besiegen des Superschurkens – und zum Erhalten der Hausaufgabenpunkte – ihre Idee auch informell.

Lösungsskizze. Idee: In q_0 wird für jeden Buchstaben links vom Trennzeichen ∇ ein korrespondierendes Zeichen aus dem Schleifenalphabet in die Schleife gelegt. Um die Ordnung beizubehalten, wird vor dem Einlesen eines Buchstabens immer die Schleife zurück zum Startsymbol durchgedreht.

Sobald das Trennezeichen eingelesen wurde, werden alle angehäuften Schleifensymbole in q_1 abgearbeitet. Dabei muss sich das rechte Wort an mindestens einer Stelle unterscheiden. Sobald dies geschieht, merken wir uns den Erfolg durch das Speichern eines Symbols F auf der Schleife. Sobald alle Schleifensymbole aus der ersten Phase abgearbeitet wurden, befindet sich entweder nur noch Z_0 in der Schleife und wir hängen in q_1 fest oder aber es befindet sich mindestens ein F auf der Schleife, mit welchem wir dann in q_2 wechseln können.

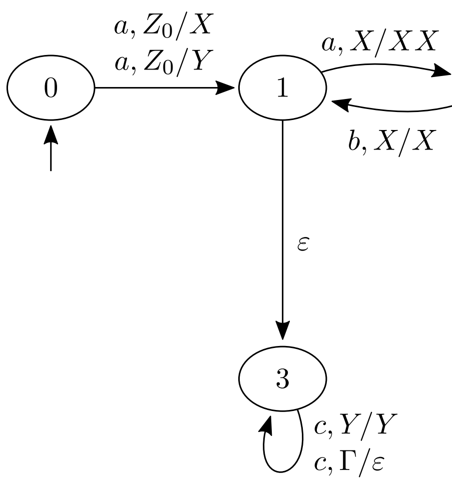
In q_2 werden schlussendlich alle übrigen Symbole von der Schleife entfernt und somit das Wort akzeptiert.



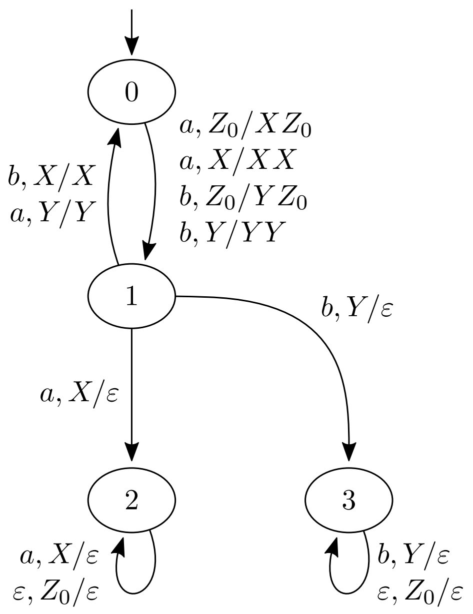
Der initiale Schleifeninhalt sei dabei Z_0.

Aufgabe H8.3. (*Sprachfindung*) 2+2+2+2 Punkte

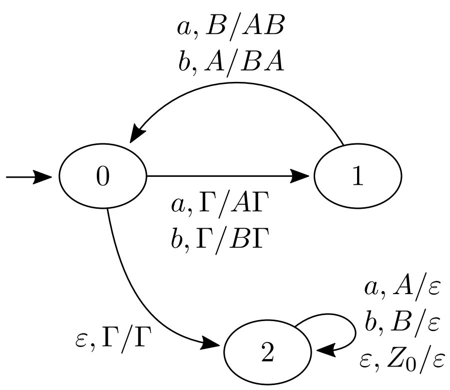
Geben Sie für die folgenden Kellerautomaten jeweils die akzeptierte Sprache in Mengennotation an. (Ihre Lösung soll sich insbesondere nicht auf den PDA beziehen!) Die PDAs akzeptieren über leeren Keller. Bitte beachten Sie die Hinweise zur Notation von PDAs auf Übungsblatt 7 und 8.



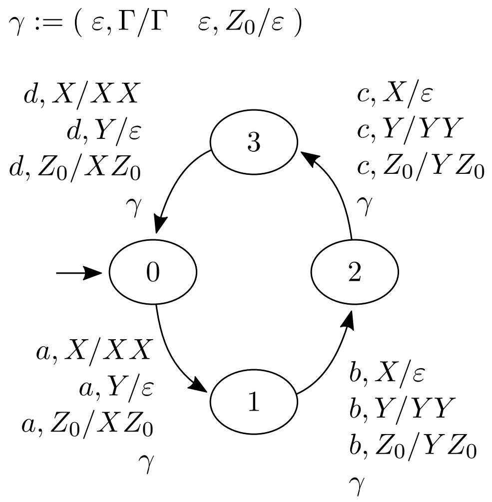
(a)



(b)



(c)



(d)

Lösungsskizze.

(a) $\{a(ab)^n c^{n+1} : n \in \mathbb{N}\} \cup L(ac^+)$

(b) $\{(ab)^n a^{n+2} : n \in \mathbb{N}\} \cup \{(ba)^n b^{n+2} : n \in \mathbb{N}\}$, oder auch $\{a(ba)^n a^{n+1} : n \in \mathbb{N}\} \cup \{b(ab)^n b^{n+1} : n \in \mathbb{N}\}$

(c) $\{ww^R : w \in L((ab|ba)^*)\}$

(d) $\{w \in \{a, b, c, d\}^* : |w|_a + |w|_d = |w|_b + |w|_c\}$

Aufgabe H8.4. (*Ein Komplement für euch*) 0 + 4 Punkte

Sei $\Sigma \coloneqq \{a, b, c\}$. Wir betrachten die Sprache

$$L \coloneqq \{a^n b^n c^n \in \Sigma^* \mid n \in \mathbb{N}\}.$$

Ziel dieser Aufgabe ist es, mit Hilfe von L zu beweisen, dass kontextfreie Sprachen nicht unter Komplement abgeschlossen sind.

(a) (Optionale Aufgabe) Zeigen Sie, dass L nicht kontextfrei ist

Hinweis: der Beweis wurde bereits in der Vorlesung geführt.

(b) Konstruieren Sie einen Kellerautomaten für $\overline{L}$ und beschreiben Sie die Idee hinter Ihrer Konstruktion.

Lösungsskizze.
(a) Siehe Vorlesung.
(b) Falls ein Wort nicht in der Sprache ist, muss die Anzahl der as ungleich der Anzahl der bs sein, die Anzahl der bs ungleich der Anzahl der cs sein, oder das Wort Buchstaben in der falschen Reihenfolge enthalten. Wir können einen der Fälle durch Nichtdeterminismus raten und dann überprüfen: