Übungen zur Vorlesung Grundbegriffe der Theoretischen Informatik

technische universität dortmund

Beate Bollig

Gaetano Geck, Thomas Harweg
David Mezlaf, Christopher Spinrath

fakultät für informatik

SoSe 2017 | Übungsblatt 5 | 23.05.2017

Abgabe bis spätestens am Dienstag, 30.05.2017, 10:00 Uhr

- **in die Briefkästen im Durchgangsflur, der die 1. Etage der OH 12 mit dem Erdgeschoss der OH 14 verbindet.**

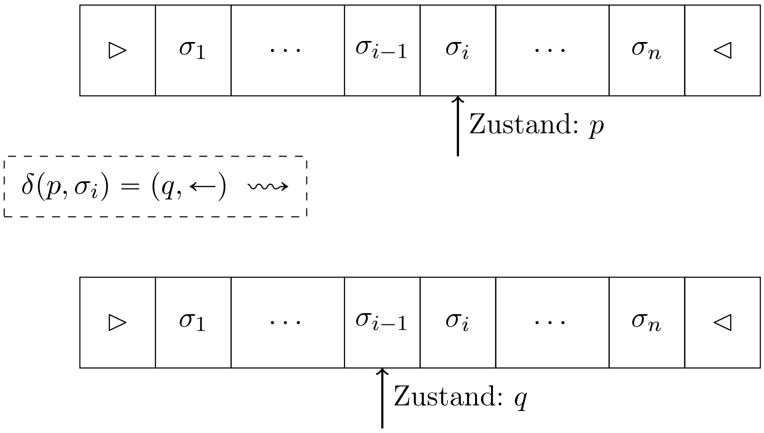
Ansonsten gelten die Hinweise von Blatt 1.

Aufgabe 5.1 [Zwei-Wege-Automaten] **5 Punkte**

Neben den in der Vorlesung vorgestellten Erweiterungen endlicher Automaten gibt es auch die sogenannten *Zwei-Wege-DFAs*. Während die „traditionellen“ endlichen Automaten ein eingegebenes Wort Zeichen für Zeichen einlesen und sich auf der Eingabe lediglich „von links nach rechts bewegen“, können Zwei-Wege-DFAs auch von rechts nach links laufen und so Teile der Eingabe mehrmals und/oder rückwärts lesen. Wie ein herkömmlicher DFA beginnt auch ein Zwei-Wege-DFAs in einem eindeutigen Startzustand und liest das erste Zeichen σ_1 der Eingabe $w = \sigma_1\sigma_2 \dots \sigma_n$.
Formal lässt sich ein *Zwei-Wege-DFA* $\mathcal{A}$ beschreiben als ein Tupel $(Q, \Sigma, \triangleright, \triangleleft, \delta, s, F)$, mit

- Zustandsmenge Q,
- Alphabet Σ,
- linkem Begrenzungssymbol $\triangleright \notin \Sigma$,
- rechtem Begrenzungssymbol $\triangleleft \notin \Sigma$,
- Transitionsfunktion $\delta : Q \times \Sigma \to Q \times \{\leftarrow, \rightarrow\}$,
- Startzustand $s \in Q$,
- Menge akzeptierender Zustände $F \subseteq Q$.

Die Transitionsfunktion gibt für jede Kombination von aktuellem Zustand $q \in Q$ und gelesenem Zeichen $\sigma \in \Sigma$ vor, in welchen Folgezustand der Automat wechselt und welches der zwei benachbarten Zeichen er als nächstes liest.
Ein Zwei-Wege-DFA verhält sich also wie ein Lesekopf, der sich auf einem Eingabeband schrittweise nach links und rechts bewegen kann, um die Eingabe zu lesen. Ein Beispiel für einen Zustandsübergang eines Zwei-Wege-DFAs auf der Eingabe $\sigma_1 \dots \sigma_n$ ist im Folgenden illustriert:



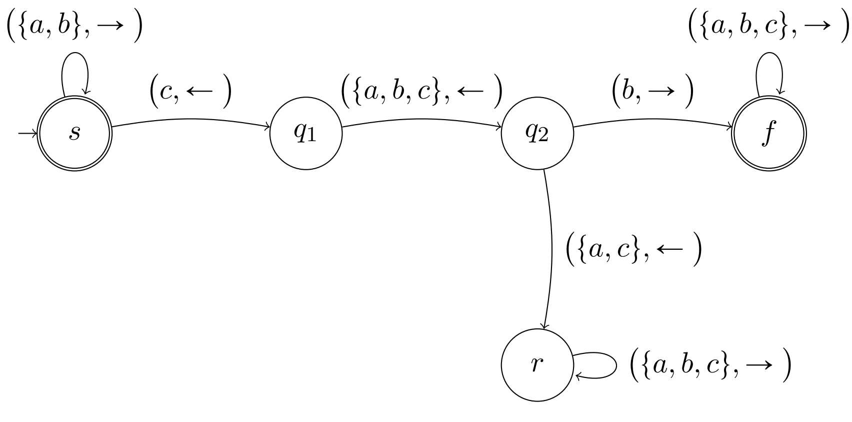
Eine Berechnung eines Zwei-Wege-DFAs endet, sobald der Automat die Eingabe nach links oder rechts verlässt, also eines der Symbole $\triangleright, \triangleleft$ liest. Die Berechnung ist genau dann akzeptierend, wenn der Zwei-Wege-DFA die Eingabe in einem akzeptierenden Zustand nach rechts verlässt. Eine Berechnung, die nicht endet, gilt ebenfalls als verwerfend.

Kurzaufgabe (1 Punkt)

Ein Zwei-Wege-DFA nach dieser Definition *muss* seinen Lesekopf mit jedem Rechenschritt bewegen. Begründen Sie, dass es keinen Nutzen für einen Zwei-Wege-DFA hat, seinen Lesekopf mit einem Rechenschritt *nicht* zu bewegen.

Hauptaufgabe (4 Punkte)

a) Beschreiben Sie die vom folgenden Zwei-Wege-DFA $\mathcal{A}$ entschiedene Sprache über dem Alphabet $\Sigma = \{a, b, c\}$:



Hinweis: Die Kurzschreibweise (S, d) mit $S \subseteq \Sigma$, $d \in \{\leftarrow, \rightarrow\}$ bezeichnet hier die Menge $\{(\sigma, d) \mid \sigma \in S\}$ von Transitionen. Liest dieser Automat also im Startzustand s ein a oder ein b, verbleibt er im Zustand s und bewegt seinen Lesekopf nach rechts. Auch in Zwei-Wege-DFAs werden die akzeptierenden Zustände, hier s und f, durch einen doppelten Rand markiert.

(1 Punkt)

b) Konstruieren Sie einen Zwei-Wege-DFA für die Sprache

$$L_1 = \{w = \sigma_1\sigma_2 \ldots \sigma_n \mid \sigma_1, \ldots, \sigma_n \in \{a, b\},\, n \geqslant 3 \wedge \sigma_{n-2} = a\}\,.$$

(1,5 Punkte)

c) Konstruieren Sie einen Zwei-Wege-DFA für die Sprache

$$L_2 = \{w = \sigma_1\sigma_2 \ldots \sigma_n \mid \sigma_1, \ldots, \sigma_n \in \{a, b\},\, \forall i, j \in \{1, \ldots, n\} : i \equiv_3 j \Rightarrow \sigma_i = \sigma_j\}\,.$$

(1,5 Punkte)

Aufgabe 5.2 [Kontextfreie Grammatiken] 5 Punkte

Kurzaufgabe (1 Punkt)

Was zeichnet eine eindeutige kontextfreie Grammatik aus?

Hauptaufgabe (4 Punkte)

Gegeben sei die folgende kontextfreie Grammatik G über dem Alphabet $\Sigma = \{a, b, c, d\}$ mit den folgenden Regeln:

- $S \to MN$
- $M \to aaMb \mid \varepsilon$
- $N \to ccNd \mid \varepsilon$

a) Geben Sie einen Ableitungsbaum und eine Linksableitung für das Wort *aaaabbccd* an. **(1 Punkt)**

b) Beschreiben Sie die von der Grammatik G erzeugten Sprache zunächst natürlichsprachlich, dann formal (Mengennotation). Gehen Sie bei der natürlichsprachlichen Beschreibung auf die Bedeutung der einzelnen Variablen ein. **(2 Punkte)**

c) Als Alternative schlagen wir für die Sprache $L(G)$ die folgende Grammatik G' vor, die mit einer kleineren Variablenmenge auskommt:

- $S \to aaSd \mid M$
- $M \to bMcc \mid \varepsilon$

Erzeugt diese Grammatik G' wirklich die gleiche Sprache $L(G)$? Begründen Sie Ihre Antwort. **(1 Punkt)**

Aufgabe 5.3 [Konstruktion kontextfreier Grammatiken] **5 Punkte**

Ein klassisches Beispiel für eine kontextfreie Sprache ist die sogenannte Klammersprache:

$$L_{()} = \{w \in \{(,)\}^* \mid \#_((w) = \#_)(w) \wedge \text{für jedes Präfix } p \text{ von } w \text{ gilt: } \#_((p) \geqslant \#_)(p)\}.$$

Wörter dieser Sprache sind „korrekt geklammert“. So liegen zum Beispiel die Wörter (()), ()() und ε in $L_{()}$, nicht jedoch ())(oder)()((().

Kurzaufgabe (1 Punkt)

Als erste kontextfreie Sprache lernten Sie in der Vorlesung die Sprache $L_{ab} = \{a^n b^n \mid n \in \mathbb{N}\}$ kennen. Konstruieren Sie basierend auf der vorgestellten Grammatik zur Sprache L_{ab} eine kontextfreie Grammatik für die Klammersprache $L_{()}$.

Hinweis: Reguläre Ausdrücke und auch arithmetische Ausdrücke sind korrekt geklammert.

Hauptaufgabe (4 Punkte)

Wir interessieren uns nun für verschiedene Einschränkungen der Klammersprache:

$L_1 = \{w \in L_{()} \mid \text{In } w \text{ folgen } \textit{höchstens drei} \text{ öffnende Klammern direkt aufeinander}\}$
$L_2 = \{w \in L_{()} \mid \text{In } w \text{ sind Klammern } \textit{höchstens dreifach} \text{ verschachtelt}\}$

a) Geben Sie für jede der Sprachen L_1, L_2 eine kontextfreie Grammatik an. **(3 Punkte)**

b) Genau eine der beiden gegebenen Sprachen L_1 und L_2 ist regulär. Entscheiden Sie, welche dieser beiden Sprachen regulär ist, und geben Sie für diese gewählte Sprache einen endlichen Automaten oder einen regulären Ausdruck an. **(1 Punkt)**

Testfragen

1. Warum ist die Klammersprache nicht regulär?
2. Lineare Grammatiken als Verallgemeinerung der rechts- und linkslinearen Grammatiken enthalten ausschließlich Regeln der Form $X \to \sigma$, $X \to \sigma Y$ und $X \to Y\sigma$, wobei X, Y Variablen sind und σ ein Terminal. Beschreiben lineare Grammatiken grundsätzlich unendliche Sprachen?
3. Ist die Rechtsableitung eines Wortes bezüglich einer kontextfreien Grammaitk G eindeutig, selbst wenn die Grammatik G nicht eindeutig ist?