6. Aufgabenblatt zur Vorlesung

Grundlagen der theoretischen Informatik

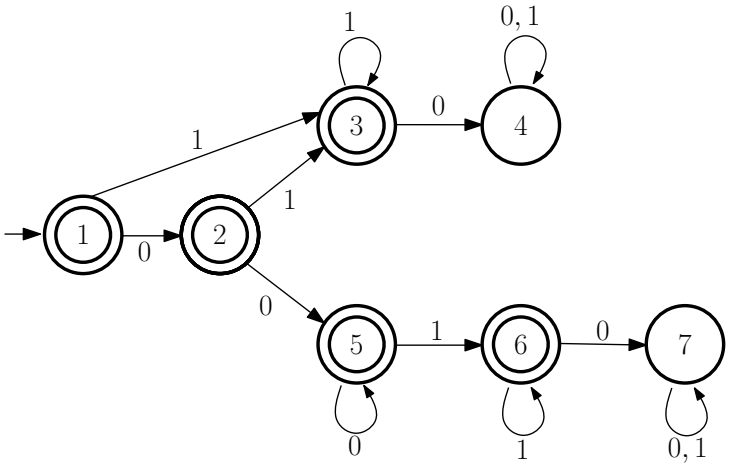
SoSe 2018

Wolfgang Mulzer, Katharina Klost

Abgabe am 1. Juni 2018 bis 10 Uhr in die Tutorenfächer

Aufgabe 1 Minimalautomat *10 Punkte*

Minimieren Sie den folgenden endlichen Automaten mit dem Algorithmus aus der Vorlesung.



Welche Sprache wird akzeptiert?

Aufgabe 2 Turing-Maschinenprogrammierung *10 Punkte*

Geben Sie eine Turing-Maschine an, welche die Sprache

$$L = \{w\#w \mid w \in \{0,1\}^*\}$$

über dem Eingabealphabet $\Sigma = \{0, 1, \#\}$ erkennt. Beschreiben Sie Ihre Maschine in Worten und formal durch Angabe der Überführungsfunktion.

Hinweis: Führen Sie zusätzliche Zeichen im Bandalphabet ein, um Markierungen auf dem Band darzustellen.

Aufgabe 3 Simulation *10 Punkte*

(a) Beschreiben Sie, wie man einen DEA durch eine Turing-Maschine simulieren kann. Beachten Sie dabei insbesondere die unterschiedlichen Akzeptierungsmodi.

(b) Beschreiben Sie, wie man eine Turing-Maschine mit beidseitig unbeschränktem Eingabe-/Arbeitsband durch eine Turing-Maschine mit einseitig unbeschränktem Band simulieren kann.

Aufgabe 4 JFlex *freiwillig, 10 Zusatzpunkte*

In der Vorlesung wurden Scanner-Generatoren erwähnt, die beim Übersetzerbau zum Einsatz kommen. Ein solcher Scanner-Generator heißt *JFlex*.

(a) Auf der Veranstaltungsseite befindet sich das Beispielprogramm `summe.flex`. Laden Sie das Programm herunter und finden Sie mit Hilfe der Dokumentation von JFlex heraus, wie man es ausführt. Führen Sie einige Testläufe durch und dokumentieren Sie diese. Gehen Sie dann den Quelltext von `summe.flex` durch und erklären Sie, was die einzelnen Anweisungen bedeuten.

(b) Bearbeiten Sie eine der beiden folgenden Aufgaben:

(i) Ermitteln Sie, wie der Algorithmus funktioniert, der Scanner-Generatoren zu Grunde liegt (z.B. aus dem Buch von I. Wegener). Erläutern Sie diesen Algorithmus kurz in zwei Absätzen. Erklären Sie auch den Zusammenhang zum Vorlesungsstoff.

(ii) Modifizieren Sie `summe.flex` so, dass bei Eingabe des Zeichens `*` die nächste Zahl nicht addiert, sondern multipliziert wird. Bei Eingabe von `=` soll das bisherige Ergebnis ausgedruckt werden. Außerdem sollen Kommentare im Stil von Haskell unterstützt werden: alles zwischen `--` und dem Ende der Zeile wird ignoriert.