Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Dr. Werner Meixner

Sommersemester 2007
Übungsblatt 9
18. Juni 2007

Einführung in die Theoretische Informatik

Abgabetermin: 25. Juni 2007 **vor** *der Vorlesung*

Hausaufgabe 1 (5 Punkte)

Eine Grammatik $G = (V, \Sigma, P, S)$ sei durch folgende Produktionen gegeben.

$$\begin{array}{lcl} S & \rightarrow & aAB\,, \\ A & \rightarrow & aS \mid bS \mid a\,, \\ B & \rightarrow & bB \mid bS \mid b\,. \end{array}$$

Konstruieren Sie einen NPDA K, der $L(G)$ nicht durch leeren Keller, sondern durch Endzustände akzeptiert.

Hausaufgabe 2 (5 Punkte)

Seien K_1 eine kontextfreie Sprache und K_2, K_3 deterministische kontexfreie Sprachen über Σ. Zeigen Sie:

1. Für $\overline{K_2 \cap K_3}$ ist das Leerheitsproblem entscheidbar.
2. Für $\overline{K_1 \cap K_2}$ ist das Wortproblem entscheidbar.

Hausaufgabe 3 (5 Punkte)

Sei L die zu dem regulären Ausdruck $0 + (1(0+1)^*)$ gehörige Sprache. Für eine Zahl $n \in \mathbb{N}_0$ bezeichne $b(n) \in L$ bzw. $u(n) \in a^*$ deren binäre bzw. unäre Darstellung. Es ist eine deterministische Ein-Band-Turing-Maschine zu konstruieren, die für $n \neq 0$ die Eingabe $b(n)$ in $u(n)$ umwandelt. Die Eingabe und die Ausgabe sollen sich auf dem sonst leeren Band befinden und der Lese-/Schreibkopf der Maschine bei Start und Ende der Berechnung soll links auf dem ersten beschriebenen Feld des Bandes stehen.

Entwerfen Sie eine Programmstruktur für die zu konstruierende Turingmaschine, indem Sie versuchen, die Gesamtlösung auf Teilprobleme zurückzuführen, für deren Lösung dann in der nächstfolgenden Aufgabe jeweils spezielle Turingmaschinen implementiert werden sollen.

Hausaufgabe 4 (5 Punkte)

Implementieren Sie Ihre in der Lösung der vorausgegangenen Aufgabe spezifizierten Turingmaschinen und fügen Sie die Programme zu einem Gesamtprogramm zusammen zur Umwandlung einer binären in eine unäre Darstellung einer natürlichen Zahl.

Hinweis: *Die als* Vorbereitung *bezeichneten Aufgaben werden nicht bewertet und dienen der häuslichen Vorbereitung der Tutoraufgaben, die ebenfalls nicht bewertet werden. Die Abgabe einer Bearbeitung der Vorbereitungsaufgaben zusammen mit der Bearbeitung der Hausaufgaben wird empfohlen. Tutoraufgaben werden in den Übungsgruppen bearbeitet.*

Vorbereitung 1

Eine Menge natürlicher Zahlen läßt sich als Teilmenge von Σ^+ über einem einelementigen Alphabet $\Sigma = \{|\}$ kodieren. Entsprechend werden wir Begriffe für formale Sprachen auf Mengen natürlicher Zahlen anwenden.
Wir betrachten die Menge $G = \{n \in \mathbb{N}\,;\, n \neq 1, (\exists\,\text{Primzahlen}^1\ x, y)[2n = x + y]\,\}$.

1. Geben Sie eine knappe Begründung, warum G entscheidbar ist!
2. Vermutlich werden Sie keine der Aussagen beweisen können, ob G leer ist oder nicht, denn Sie müssten dazu die Goldbachsche Vermutung beweisen oder widerlegen. Warum können Sie trotzdem zeigen, dass für G das Leerheitsproblem entscheidbar ist?

Vorbereitung 2

1. Gegeben seien zwei entscheidbare Prädikate $P(x, y)$ und $Q(x, y)$. Zeigen Sie, dass $R(x, y) = P(x, y) \wedge Q(x, y)$ entscheidbar ist.
2. Ist jede Teilmenge einer rekursiven Sprache rekursiv aufzählbar? Beweis!

Tutoraufgabe 1

1. Gegeben seien zwei rekursiv aufzählbare Teilmengen M_1 und M_2 der Menge Σ^* aller Wörter über einem Alphabet Σ. Sind die folgenden Mengen rekursiv aufzählbar? Beweisen Sie Ihre Antworten!

   a) $M_a = M_1 \setminus M_2$, b) $M_b = (M_1 \cap M_2) \cup (\overline{M_1 \cup M_2})$.

2. Sei $M \subseteq \mathbb{N} \times \mathbb{N}$ eine rekursiv aufzählbare Relation. Zeigen Sie, dass die transitive Hülle von M wieder rekursiv aufzählbar ist!

Tutoraufgabe 2

Die Produktionen einer kontextfreien Grammatik $G = (V, \Sigma, P, S)$ seien gegeben durch

$$\begin{array}{ll} S \to A\,, & A \to E \mid E + A \mid E - (A)\,, \\ E \to P \mid P \times E \mid E/P\,, & P \to (A) \mid a\,. \end{array}$$

1. Zeigen Sie für ein geeignetes k, dass G eine LR(k)-Grammatik ist.
2. Zeigen Sie durch Anwendung von Earley's Algorithmus, dass $a \times a - (a + a) \in L(G)$ gilt. Welchen Vorteil kann man aus der Kenntnis der oben eingeführten Lookaheads bei der Ausführung des Earley-Algorithmus ziehen?

[1] 1 ist keine Primzahl