Theoretische Informatik 2

Sommersemester 2018

Elizaveta Kovalevskaya
Georg Schnitger
Philipp Tertel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Institut für Informatik
AG Theoretische Informatik

Übungsblatt 7

Ausgabe: 29.05.18
Abgabe: 05.06.18

Für den Nachweis der Zugehörigkeit einer Sprache zur Klasse **DL** bzw. zur Klasse **NL** können Sie annehmen, dass konstant viele Arbeitsbänder mit jeweils einem Lese-/Schreibkopf zur Verfügung stehen, und dass konstant viele Leseköpfe für das Eingabeband vorhanden sind. Auf diesem Blatt können Sie bis zu 8 Bonuspunkte erwerben.

Aufgabe 7.1 *DL* ((2+4)+6 = 12 Punkte)

Zeigen Sie, dass die folgenden Probleme mit deterministischen Algorithmen auf logarithmischem Speicher gelöst werden können. Eine informelle Beschreibung Ihrer Vorgehensweise ist ausreichend, solange Sie die Idee Ihrer Konstruktion überzeugend beschreiben. Begründen Sie aber sorgfältig, warum Ihr Algorithmus mit logarithmischen Speicherplatz auskommt.

a) Sei $\Sigma = \{0,1\}$. Für ein Wort $u \in \{0,1\}^*$ sei $\mathrm{Zahl}(u)$ die natürliche Zahl mit Binärdarstellung u: das niedrigstwertige Bit steht also ganz rechts.

   i) Zeigen Sie, dass die Addition zweier Binärzahlen sogar ohne Arbeitsbänder gelingt, wenn konstant viele Leseköpfe zur Verfügung stehen: Für Eingabe $x\#y$ (mit $x, y \in \{0,1\}^*$) ist die Binärdarstellung z der Summe $\mathrm{Zahl}(x) + \mathrm{Zahl}(y)$ auf dem Ausgabeband auszugeben.

   ii) Zeigen Sie, dass die Multiplikation von zwei Binärzahlen auf logarithmischem Platz ausgeführt werden kann: Für Eingabe $x\#y$ (mit $x, y \in \{0,1\}^*$) ist die Binärdarstellung z des Produkts $\mathrm{Zahl}(x) \cdot \mathrm{Zahl}(y)$ auf dem Ausgabeband auszugeben. Für $n = |x| + |y|$ dürfen nur $O(\log_2 n)$ Zellen des Arbeitsbands benutzt werden.

b) Sei L die Sprache aller wohlgeformten Klammerausdrücke mit den Klammertypen $(,)$ und $[,]$. D. h. L wird erzeugt von der Grammatik $G = (\Sigma, \{S\}, S, P)$ mit dem Alphabet $\Sigma = \{\,(,),[,]\,\}$ und den Produktionen

$$S \to (S) \mid [S] \mid SS \mid \varepsilon.$$

Zeigen Sie, dass L zur Klasse **DL** gehört.

Hinweis: Der Fall von zwei Klammertypen ist komplizierter als der eines Klammertyps. Das Beispiel ([ )] zeigt, dass sich die beiden Klammertypen gegenseitig „stören“ können.

Aufgabe 7.2 *Das Wortproblem von DFAs und NFAs* (2+4+4 = 10 Punkte)

Sei $A = (\Sigma, Q, \delta, q_0, F)$ ein DFA oder NFA. Wie schwierig ist die Simulation von DFAs oder NFAs? Um diese Frage zu beantworten, entwerfen wir zuerst eine „maschinenverständliche“ Kodierung $\langle A\rangle$ des jeweiligen Automaten. Dazu nehmen wir an, dass $\Sigma = \{0,1\}$, $Q = \{1,2,\ldots,k\}$ und $q_0 = 1$ gilt; es müssen also nur noch k, F und δ spezifiziert werden. Für $F = \{r_1,\ldots,r_f\}$ setze

$$\langle A\rangle := 1^k\#1^{r_1}\#1^{r_2}\#\cdots\#1^{r_f}\#\#\cdot\prod_{(p,x,q)\in\delta}\Big(1^p\#1^x\#1^q\#\Big)$$

a) Zeigen Sie, dass DFAs mit logarithmischem Speicherplatz simuliert werden können, d.h. zeigen Sie, dass das Wortproblem für DFAs, also die Sprache

$$L_{\text{DFA}} = \{\, \langle A\rangle w \;:\, w \in \{0,1\}^*,\ \text{der DFA } A \text{ akzeptiert } w \,\}$$

in **DL** liegt.

b) Das Wortproblem für NFAs ist schwieriger: Zeigen Sie, dass die Sprache

$$L_{\text{NFA}} = \{\, \langle A\rangle w \;:\, w \in \{0,1\}^*,\ \text{der NFA } A \text{ akzeptiert } w \,\}$$

NL-vollständig ist.

Hinweis: Zeigen Sie zuerst, dass L_{NFA} in **NL** liegt. Dann genügt der Nachweis der LOGSPACE-Reduktion

$$\text{D-Reachability} \leq_{\text{LOG}} L_{\text{NFA}}.$$

c) Zeigen Sie, dass die Sprache

$$E_{\text{DFA}} = \{\, \langle A\rangle \;:\; A \text{ ist ein DFA mit } L(A) \neq \emptyset \,\}$$

NL-vollständig ist.

Hinweis: Zeigen Sie zuerst, dass E_{DFA} in **NL** liegt. Dann genügt der Nachweis der LOGSPACE-Reduktion

$$\text{D-Reachability} \leq_{\text{LOG}} E_{\text{NFA}}.$$

Hier müssen Sie einem gerichteten Graphen G einen DFA A_G über eine Logspace-Reduktion zuweisen. Betrachten Sie zuerst den Fall, dass jeder Knoten in G höchstens zwei direkte Nachfolger hat.

Aufgabe 7.3 *Speicherplatzkomplexität kontextfreier Sprachen* (4 + 2 + 4 Punkte)

Sei L eine beliebige kontextfreie Sprache. Man kann zeigen, dass das Wortproblem für L („gehört die Eingabe w zu L?“) in der Komplexitätsklasse DSPACE($\log_2^2 n$) liegt. Es ist nicht bekannt, ob das Wortproblem für L zu **NL** gehört.

a) Die entsprechende Frage für lineare Sprachen ist anscheinend einfacher. Sei L eine kontextfreie Sprache, die von einer linearen Grammatik $G = (\Sigma, V, S, P)$ erzeugt werde, d.h. es gilt $X \to \alpha Y \beta$ für alle Produktionen in P, wobei α, β Worte über Σ und X, Y Variablen sind.

Zeigen Sie, dass L in **NL** liegt.

b) Sei $\Sigma = \{1, \#\}$. Für eine Kante $e = (i,j) \in \mathbb{N}_{>0} \times \mathbb{N}_{>0}$ ist das Wort $w_e := 1^i\#\#1^j$ die Beschreibung der Kante e. Ein Wort $w \in \Sigma^*$ heißt *legal*, wenn es einen Weg $v_1 \to v_2 \to \cdots \to v_k$ mit $v_1 = 1$ und $v_k = 2$ gibt, sodass

$$w \in \Sigma^* \cdot \# \cdot w_{(v_1,v_2)} \cdot \# \cdot \Sigma^* \cdot \# \cdot w_{(v_2,v_3)} \cdot \# \cdot \Sigma^* \cdots \Sigma^* \cdot \# \cdot w_{(v_{k-1},v_k)} \cdot \# \cdot \Sigma^*$$

Wir definieren die Sprache

$$L = \{\, w \in \Sigma^* \;:\; w \text{ ist legal} \,\}$$

i) Zeigen Sie, dass L kontextfrei ist.

ii) Zeigen Sie, dass L **NL**-vollständig ist.

Kommentar: Es gibt also kontextfreie Sprachen, deren Wortproblem **NL**-vollständig ist. Das Wortproblem für reguläre Sprachen ist viel einfacher und ohne Speicher lösbar.