Komplexitätstheorie

Übungsblatt 5

Abgabe als PDF bis 19. 6. 2018, 23:59 Uhr in Stud.IP, Ordner „Abgabe Übungsblatt 5“
Bitte nur eine PDF-Datei pro Gruppe, Lizenz „Selbst verfasstes, nicht publiziertes Werk“.

---

1. (25 %) Zeige, dass die folgenden Probleme in LogSpace sind. Gib dazu entweder eine präzise Beschreibung einer LogSpace-DTM oder einen Algorithmus in Pseudocode an. Analysiere den Speicherbedarf.

a) Gegeben 3 natürliche Zahlen a, b, c als Zeichenkette $\mathsf{bin}(a)\#\mathsf{bin}(b)\#\mathsf{bin}(c)$, entscheide, ob $a + b = c$ gilt.

b) Gegeben zwei Wörter w, v über einem festen Alphabet Σ als Zeichenkette $w\#v$, entscheide, ob v ein Teilwort von w ist (also ob $w = u_1 v u_2$ mit $u_1, u_2 \in \Sigma^*$).

2. (25 %) Die folgende Reduktion von 3SAT auf CLIQUE wurde in der Vorlesung „Theoretische Informatik 2“ beschrieben:

Zu einer gegebenen 3-Formel $\varphi = (\ell_{11} \vee \ell_{12} \vee \ell_{13}) \wedge \cdots \wedge (\ell_{m1} \vee \ell_{m2} \vee \ell_{m3})$ mit den Literalen $\ell_{ij} \in \{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\}$ konstruiert man den ungerichteten Graphen $G = (V, E)$, dessen Knoten die *Vorkommen* der Literale sind (also gibt es $3m$ Knoten) und dessen Kanten alle Literalvorkommen verbinden, die in verschiedenen Klauseln auftreten und nicht komplementär sind, d. h.:

$$V = \{\langle i, j\rangle \mid 1 \leq i \leq m \text{ und } 1 \leq j \leq 3\}$$
$$E = \{\{\langle i, j\rangle, \langle i', j'\rangle\} \mid i \neq i' \text{ und } \ell_{ij} \neq \overline{\ell}_{i',j'}\}$$

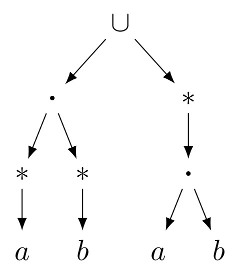
$$\text{wobei } \overline{\ell} = \begin{cases} \neg x & \text{falls } \ell = x \\ x & \text{falls } \ell = \neg x \end{cases}$$

Außerdem setzt man $k = m$. Aus „Theoretische Informatik 2“ ist bekannt: φ ist erfüllbar gdw. G eine k-Clique hat.

Zeige, dass man G und k mit einem LogSpace-Transduktor berechnen kann.

3. (25 %) Zeige, dass das folgende Problem in NLogSpace ist: gegeben einen regulären Ausdruck α über einem Alphabet Σ und ein Wort $w \in \Sigma^*$, entscheide ob $w \in L(\alpha)$.

Der Einfachheit halber sei angenommen, dass der reguläre Ausdruck α als Syntaxbaum gegeben ist, z. B. $(a^* \cdot b^*) \cup (a \cdot b)^*$ wie im Bild.
Diese Bäume werden wie folgt als Eingabe repräsentiert: Die Knoten sind Binärzahlen; der Baum ist gegeben als Liste von beschrifteten Knoten gefolgt von “#” als Trennzeichen gefolgt von Kantenliste. In der Kantenliste steht der linke Nachfolger vor dem rechten. Der abgebildete Baum wird also z. B. durch die folgende Liste repräsentiert (alle Zahlen binär kodiert):



$$(0, \cup)(1, \cdot)(2, *)(3, a)(4, *)(5, b)(6, *)(7, \cdot)(8, a)(9, b)\#$$
$$(0, 1)(1, 2)(2, 3)(1, 4)(4, 5)(0, 6)(6, 7)(7, 8)(8, 9)$$

Bitte wenden.

4. (25 %) Entwirf eine Schaltkreisfamilie für jede der folgenden Sprachen, so dass die Größe der Schaltkreise polynomiell durch die Größe der Eingabe beschränkt ist und die Tiefe logarithmisch:

   a) $L_1 = \{0^n 1^n \mid n \geq 0\}$;

   b) $L_2 = \{0^n 1^m \mid n, m \geq 0\}$

   c) $L_3 = \{(01)^n \mid n \geq 0\} \cup \{(10)^n \mid n \geq 0\}$

5. **Zusatzaufgabe** (20 %) Aus der Definition von NP kennen wir Beweissysteme. Wir nennen ein Beweissystem R *logarithmisch*, wenn gilt:

   - Es gibt ein Polynom p, so dass $|b| \leq p(|w|)$ für alle $(w, b) \in R$ gilt.
   - Es gibt einen LogSpace-*Verifizierer*, der R entscheidet: dies ist eine terminierende LogSpace-DTM M, die w und b auf zwei getrennten Eingabebändern bekommt und akzeptiert gdw. $(w, b) \in R$; dabei darf M auf dem Eingabeband für b den Lesekopf in jedem Berechnungsschritt nur nach rechts bewegen.

   a) Gib ein logarithmisches Beweissystem für GAP an.

   b) Zeige, dass NLogSpace die Klasse aller Probleme ist, für die es ein logarithmisches Beweissystem gibt.