Prof. Dr. Thomas Schneider SoSe 2018

Komplexitätstheorie

Übungsblatt 2

Abgabe als PDF bis **So.** 6. 5. 2018, 23:59 Uhr in Stud.IP, Ordner „Abgabe Übungsblatt 2"
Bitte nur eine PDF-Datei pro Gruppe, Lizenz „Selbst verfasstes, nicht publiziertes Werk".

---

1. (25 %)

a) Das Problem *2-Färbbarkeit (*2F*)* für ungerichtete Graphen ist analog zu 3-Färbbarkeit definiert, nur mit zwei statt mit drei Farben. Zeige, dass $2\mathsf{F} \in \mathsf{P}$.

b) Eine Formel φ in Klauselform heißt *Spezialformel*, wenn jede Variable höchstens zwei Mal in φ vorkommt, wobei sowohl positive als auch negative Vorkommen gezählt werden. Sei SpSAT die Menge aller erfüllbaren Spezialformeln. Zeige, dass $\mathsf{SpSAT} \in \mathsf{P}$.

Hinweis: Eliminiere wiederholt Variablen, die mehrfach in der Eingabeformel auftauchen, so dass sich die (Un)Erfüllbarkeit der Formel erhält.

Man kann auch zeigen, dass 2SAT (die Menge aller erfüllbaren 2-Formeln) in P ist, was wir hier aber nicht tun wollen.

2. (20 %) Ein nichtdeterministischer endlicher Automat (NEA) heißt *azyklisch (ANEA)*, wenn seine Übergangsrelation azyklisch ist, d. h. es gibt keine Folge $a_0, \ldots, a_{n-1}$ von Symbolen und $q_0, \ldots, q_n$ von Zuständen, so dass $q_0 = q_n$ und

$$(q_i, a_i, q_{i+1}) \in \Delta \quad \text{für alle } i < n.$$

Beweise durch Angeben eines polynomiellen Beweissystems, dass das folgende Problem in NP ist: Gegeben ANEAs $\mathcal{A}_1$ und $\mathcal{A}_2$, gilt $L(\mathcal{A}_1) \not\subseteq L(\mathcal{A}_2)$?

3. (25 %) Für eine Sprache L ist $L_0 = \{\varepsilon\}$, $L_{i+1} = L_i \cdot L$ für alle $i \geq 0$ und $L^* = \bigcup_{i \geq 0} L_i$. Beweise:

a) Wenn $L \in \mathsf{NP}$, dann $L^* \in \mathsf{NP}$.

b) Wenn $L \in \mathsf{P}$, dann $L^* \in \mathsf{P}$.

Es genügt, Algorithmen in Pseudocode (und nicht als TM) anzugeben und ihre Korrektheit zu begründen sowie die Laufzeit zu analysieren.

Hinweis: Für b) hilft dynamische Programmierung.

4. (10 %) Beweise mithilfe der Resultate aus der Vorlesung:

Wenn $\mathsf{P} = \mathsf{NP}$, dann ist jedes nichttriviale NP-Problem auch NP-vollständig.

(Ein Problem $L \subseteq \Sigma^*$ heißt nichttrivial, wenn $L \neq \emptyset$ und $L \neq \Sigma^*$.)

5. (20 %) Beweise durch Reduktion von 3SAT, dass IPROG NP-hart ist (beachte die Hinweise auf Folie 44 in Kapitel 3).

Zur Erinnerung: IPROG lässt nur *Gleichungen* bestimmter Form zu (siehe Folie 10). Möchte man z. B. auch „$\leq$" und „$\geq$" verwenden, so muss man dies vorher unter Verwendung von „$=$" als Abkürzung definieren.

Bitte wenden.

6. Zusatzaufgabe (20 %) Sei φ eine AL-Formel in KNF. Eine *$\neq$-Wertzuweisung ($\neq$-WZ)* für die Variablen in φ ist eine Wertzuweisung, so dass jede Klausel in φ mindestens zwei Literale mit ungleichen Wahrheitswerten enthält. Mit anderen Worten: eine $\neq$-WZ erfüllt φ ohne in irgendeiner Klausel alle drei Literale wahr zu machen. Sei $\neq$3SAT die Menge aller 3-Formeln, für die es eine $\neq$-WZ gibt. Beweise, dass $\neq$3SAT NP-vollständig ist.

Hinweis: Für die NP-Härte verwende eine Reduktion von 3SAT, die jede 3-Klausel in zwei 3-Klauseln umwandelt. Um die Korrektheit der Reduktion zu beweisen, hilft es wahrscheinlich, folgende Beobachtung auszunutzen: wenn man die Wahrheitswerte einer $\neq$-WZ für eine AL-Formel φ vertauscht (wahr durch falsch, falsch durch wahr), erhält man wieder eine $\neq$-WZ für φ.