Technische Universität München
Fakultät für Informatik
Prof. Tobias Nipkow, Ph.D.
Dr. Werner Meixner, Alexander Krauss

Sommersemester 2008
Übungsblatt 11
7. Juli 2008

Einführung in die Theoretische Informatik

Abgabetermin: 7. Juli 2008 **vor** *der Vorlesung in die THEO Briefkästen*

Hausaufgabe 1 (4 Punkte)

Beim Post'schen Korrespondenzproblem bezeichnen wir eine Lösung $i_1, \dots, i_n$ als *c-beschränkt*, wenn für alle $j \le n$ gilt:

$$\big|\, |x_{i_1} \dots x_{i_j}| - |y_{i_1} \dots y_{i_j}| \,\big| \le c\ .$$

Zeigen Sie: Für gegebenes c ist es entscheidbar, ob eine PCP-Instanz eine c-beschränkte Lösung hat.
Hinweis: Konstruieren Sie einen endlichen Automaten über $\Sigma = \{1, \dots, k\}$, der ein Wort $i_1 \dots i_n$ genau dann akzeptiert, wenn es eine c-beschränkte Lösung ist.

Hausaufgabe 2 (4 Punkte)

Sei $\Sigma = \{0, 1\}$ und $f : \Sigma^* \to \Sigma^*$ eine beliebige (möglicherweise partielle) Funktion. Der *Graph* von f ist die Relation $G_f = \{(v, w) \in \Sigma^* \times \Sigma^* \mid f(v) = w\}$.
Beweisen oder widerlegen Sie:

1. Wenn G_f entscheidbar ist, dann ist f berechenbar.
2. Wenn f berechenbar ist, dann ist G_f entscheidbar.

Hausaufgabe 3 (4 Punkte)

Beweisen oder widerlegen Sie: Wenn A semi-entscheidbar und B entscheidbar ist, dann ist $A \setminus B$ semi-entscheidbar.

Hausaufgabe 4 (4 Punkte)

Zeigen Sie:

1. $L_1 = \{w \mid \exists v.\ \varphi_w(v) \ne \bot\}$ ist rekursiv aufzählbar.
2. $L_2 = \{w \mid \forall v.\ \varphi_w(v) \ne \bot\}$ ist nicht rekursiv aufzählbar.

Hausaufgabe 5 (4 Punkte)

Wir betrachten eine modifizierte WHILE-Sprache, in der alle Variablen durch eine feste Konstante C beschränkt sind. (Z.B. ist $C = 2^{32}$, wenn die Variablen durch 32-Bit-Register implementiert werden.) Die Syntax der Sprache ist nun dieselbe wie bei WHILE, aber die Semantik der Anweisung $x_i := x_j + n$ ändert sich insofern, dass der neue Wert von x_i dann $(x_j + n) \bmod C$ ist.
Zeigen Sie, dass das Halteproblem für diese Art von WHILE-Programmen entscheidbar ist.

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

Zeigen Sie, dass die polynomielle Reduzierbarkeit $\leq_p$ eine transitive Relation ist.

Tutoraufgabe 1

1. Ist $ntime_M$ für jede deterministische Turingmaschine M berechenbar? Begründung!
2. Sei $f : \mathbb{N} \to \mathbb{N}$. Falls $NTIME(f(n))$ eine nichtentscheidbare Sprache enthält, dann ist f nicht berechenbar. Beweis!
3. Sei $f : \mathbb{N} \to \mathbb{N}$ berechenbar. Dann gilt für alle NTM M:
$$(\forall w \in \Sigma^*.\ ntime_M(w) \leq f(|w|)) \quad\Longrightarrow\quad ntime_M \text{ berechenbar.}$$
4. Wir betrachten die Komplexitätsklasse P. Dann gibt es für jede DTM M mit $L(M) \in \text{P}$ ein Polynom p, so dass $time_M(w) \leq p(|w|)$ für alle $w \in \Sigma^*$ gilt.

   Richtig oder Falsch? Begründung!
5. Ist jede in polynomieller Zeit berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$ auch polynomiell beschränkt ($\exists$ Polynom $p.\ \forall n.\ f(n) \leq p(n)$)? Begründung!
6. Sei p ein Polynom und M eine DTM mit $time_M(w) \leq p(|w|)$ für alle $w \in \Sigma^*$. Konstruieren Sie einen einfachen polynomiell beschränkten Verifikator für $L(M)$.

Tutoraufgabe 2

Das Stundenplan-Problem, das in der Praxis allen bekannt ist, lässt sich vereinfacht wie folgt formal beschreiben:

Gegeben: Endliche Mengen S (Studierende), V (Vorlesungen) und T (Termine) und eine Relation $R \subseteq S \times V$. Dabei bedeutet $(s, v) \in R$, dass s die Vorlesung v besuchen möchte.

Problem: Gibt es eine Abbildung $f : V \to T$, so dass alle Studierenden einen überschneidungsfreien Stundenplan haben, also
$$(s, v_1) \in R \wedge (s, v_2) \in R \wedge v_1 \neq v_2 \implies f(v_1) \neq f(v_2)\ .$$

1. Zeigen Sie, dass Stundenplan in NP liegt.
2. Zeigen Sie, dass Stundenplan NP-hart ist, indem sie eine polynomielle Reduktion von 3col auf Stundenplan angeben.
3. Geben Sie nun eine Reduktion von Stundenplan auf SAT an, die es erlaubt, das Stundenplanproblem mit Hilfe eines SAT-Solvers zu lösen.