



Klausur Komplexitätstheorie WS 99/00

NICHT MIT BLEISTIFT ODER ROTSTIFT SCHREIBEN!

Heften Sie die Blätter bei Abgabe zusammen, und tragen Sie auf jedem Blatt Ihren Namen, Vornamen, Studiennummer und Matrikel ein.

Name, Vorname:

Studiennummer und Matrikel:

abgegeben: ... **Aufgabenblätter**
 ... **eigene Blätter/... Seiten mit Lösungen**

Aufgabe	1	2	3	4	5	6	7	Ges.
erreichbare Punktzahl	30	12	12	6	15	15	10	100
erreichte Punktzahl								

Es sind keine Hilfsmittel, insbesondere Taschenrechner oder Mobiltelefone, zugelassen.

Aufgabe 1

[30 Punkte]

Bitte kreuzen Sie für jede der folgenden 10 Fragen entweder “JA” oder “NEIN” in jeder Zeile an.

Bewertung: Ist C die Anzahl der richtigen Antworten, so errechnet sich die Anzahl P der erzielten Punkte aus $P := \frac{3}{2} \cdot \max\{0, C - 10\}$. **Insbesondere: Kein Kreuz, kein Punkt.**

Hinweis: Bisher nur vermutete oder bisher unbekannte Aussagen sind zu verneinen.

(a) Sei $L \in P$. Welche der folgenden Aussagen stimmt/stimmen?

JA NEIN

Für jede Sprache L' gilt: $L \leq_p L' \implies L' \in P$.

Für jede Sprache L' gilt: $L' \leq_p L \implies L' \in P$.

Für jede Sprache L' gilt: $L' \text{ NP-vollständig} \implies L \leq_p L'$.

(b) Es sei M eine $O(n^2)$ -zeitbeschränkte DTM und $L \in \Sigma^*$ eine Sprache; sei $f : \Sigma^* \rightarrow \Sigma^*$ so, dass $x \in L \iff f(x) \in L_M$. Welche der folgenden Aussagen stimmt/stimmen stets?

JA NEIN

f ist auf einer $O(n^3)$ -zeitbeschränkten DTM berechenbar $\implies L \in \text{DTIME}(n^3)$.

f ist auf einer $O(n^3)$ -zeitbeschränkten DTM berechenbar $\implies L \in \text{DTIME}(n^5)$.

f ist auf einer $O(2^n)$ -zeitbeschränkten DTM berechenbar $\implies L \in \text{DTIME}(4^n)$.

(c) Welche der folgenden Aussagen stimmt/stimmen für alle Funktionen $f : \{0, 1\}^* \rightarrow \mathbb{N}$?

JA NEIN

f ist auf einer RAM mit Kosten $O(n^3)$ im logarithmischen Kostenmaß berechenbar $\implies f \in FP$.

f ist auf einer 3-Band-TM in Zeit $O(n^3)$ berechenbar $\implies f \in FP$.

$f \in FP \implies f$ ist auf einer 1-Band-TM in Zeit $O(p(n))$ berechenbar für ein Polynom

p .

(d) Welche der folgenden Größen bestimmt/bestimmen die „Zeit“ $t_M(x)$ einer NTM M auf einer Eingabe x ?

JA NEIN

Die Anzahl der Schritte in einer kürzesten Berechnung von M auf x .

Die Gesamtzahl aller Schritte in allen möglichen Berechnungen von M auf x .

Die Tiefe des Berechnungsbaumes $CT(M, x)$.

(e) Die Aussage „Es gibt eine Sprache $L \in NP - P$ “ ...

JA NEIN

... ist ein offenes Problem in der Theoretischen Informatik.

... folgt aus dem Satz von Cook.

... impliziert, dass $3\text{-SAT} \in NP - P$ ist.

(f) Welche der folgenden Aussagen stimmt/stimmen?

JA NEIN

$P \neq NP \implies$ Das Problem TRAVELING SALESPERSON besitzt keinen Approximations-Algorithmus mit endlicher Güte.

$P \neq NP \implies$ Das Problem BINPACKING besitzt keinen Approximations-Algorithmus mit endlicher Güte.

$P \neq NP \implies$ Das Problem RUCKSACK besitzt kein polynomielles Approximations-schema.

(g) Welche der folgenden Aussagen stimmt/stimmen bewiesenermaßen oder per Definition?

JA NEIN

$P \subseteq \bigcup_{k \geq 0} DTIME(n^k)$.

$NP \subseteq \bigcup_{k \geq 0} DTIME(2^{n^k})$.

$NP \subseteq DTIME(2^n)$.

(h) Welche der folgenden Aussagen stimmt/stimmen?

JA NEIN

Für jede Sprache L gilt: $\text{SAT} \leq_p L$ und $L \leq_p \text{SAT} \implies L$ NP -vollständig.

Für jede Sprache L gilt: L NP -vollständig $\implies \text{SAT} \leq_p L$ und $L \leq_p \text{SAT}$.

Für jede Sprache L gilt: L NP -schwer $\implies \text{SAT} \leq_p L$.

(i) Sei $L \in P$. Welche der folgenden Aussagen stimmt/stimmen bewiesenermaßen?

JA NEIN

$\{x \in \{0, 1\}^* \mid \forall y \in \{0, 1\}^{|x|} : x \# y \in L\} \in NP$.

$\{x \in \{0, 1\}^* \mid \exists y \in \{0, 1\}^{|x|} : x \# y \in L\} \in NP$.

$\{x \in \{0, 1\}^* \mid \forall y_1 \in \{0, 1\}^{|x|} \exists y_2 \in \{0, 1\}^{|x|} : x \# y_1 \# y_2 \in L\} \in \Sigma_2^p$.

(j) Welche der folgenden Aussagen stimmt/stimmen bewiesenermaßen?

JA NEIN

$NSPACE(\log n) \subseteq P \subseteq NP \subseteq PSPACE$.

$NSPACE(n^2) \subseteq DSPACE(n^3)$.

$NSPACE(n) \subseteq \bigcup_{c > 0} NTIME(2^{cn})$.

Name, Vorname, Stud.-Nr., Matrikel:

**Am Ende des Blattes befinden
WICHTIG sich Definitionen einiger WICHTIG
Konzepte**

Aufgabe 2 (Linearzeitreduktionen)

[12 Punkte]

Für n Geradenstücke (Strecken) g_1, g_2, \dots, g_n , die jeweils durch die beiden Endpunkte $p_i^a \in \mathbb{R}^2$ und $p_i^b \in \mathbb{R}^2$ ($1 \leq i \leq n$) gegeben sind, soll getestet werden, ob sich irgendwelche der n Geradenstücke schneiden. Dies ist das Entscheidungsproblem der folgenden Sprache LSIT (*Line Segment Intersection Test*):

$$\text{LSIT} = \{(g_1, \dots, g_n) \mid n \in \mathbb{N}, \exists 1 \leq i < j \leq n : g_i \text{ und } g_j \text{ haben einen gemeinsamen Punkt}\}$$

Zeigen Sie: ε -CLOSENESS \leq_N LSIT. Geben Sie eine möglichst gute untere Schranke für die Zeit an, die man benötigt, um LSIT zu entscheiden.

Aufgabe 3 (Satz von Cook)

[12 Punkte]

Es sei M eine nichtdeterministische Turingmaschine mit der Zustandsmenge Q . Zu $x \in \Sigma^*$ wird im Beweis des Satzes von Cook eine KNF-Formel φ_x konstruiert, die folgende Teilformel φ_2 enthält:

$$\varphi_2 \equiv \bigwedge_{0 \leq t \leq T} \left(\left(\bigvee_{q \in Q} Y_{t,q} \right) \wedge \bigwedge_{\substack{q, q' \in Q \\ q \neq q'}} (\bar{Y}_{t,q} \vee \bar{Y}_{t,q'}) \right)$$

(Dabei ist $T = p(|x|)$ eine Schranke für die Laufzeit von M auf x .)

Zeigen Sie: Für jede Belegung v der Variablen von φ_2 gilt:

$$v(\varphi_2) = 1 \iff \text{Für jedes } t, 0 \leq t \leq T, \text{ gilt: Es gibt genau ein } q \in Q \text{ mit } v(Y_{t,q}) = 1.$$

Aufgabe 4

[6 Punkte]

Beschreiben Sie eine Sprache $L^0 \in P$ mit:

$$x \in \text{INDEPENDENT SET} \iff \exists y \in \{0, 1\}^{q(|x|)} : x \# y \in L^0.$$

Geben Sie dabei q an.

Aufgabe 5 (NP-Vollständigkeit)

[15 Punkte]

Formulieren Sie 2-PROCESSOR SCHEDULING als Sprache und zeigen Sie, dass 2-PROCESSOR SCHEDULING NP-vollständig ist. Benutzen Sie dabei für die Reduktion das Problem PARTITION.

Aufgabe 6 (Varianten des Vertex-Cover-Problems)

[15 Punkte]

(a) Formulieren Sie folgende Varianten des Vertex-Cover-Problems:

- (i) Suche nach optimaler Substruktur

- (ii) Berechnung des optimalen Parameters
- (iii) Entscheidungsvariante

(b) Es sei ein Algorithmus \mathcal{A} für (ii) gegeben. Formulieren Sie einen Algorithmus \mathcal{B} für (i), der abgesehen von polynomiell vielen Aufrufen von \mathcal{A} nur polynomielle Zeit braucht. **Hinweis:** Benutzen Sie folgende (nicht von Ihnen zu beweisende Aussage): Es sei f eine Funktion, die (ii) berechnet. Dann gilt:

$$f(G) > f(G - v) \iff \begin{array}{l} \text{Es gibt eine Knotenüberdeckung von } G \text{ mit } f(G) \\ \text{Knoten, die } v \text{ enthält.} \end{array}$$

Dabei ist $G - v := (V - \{v\}, E - \{(u, v) \mid u \in V\})$ für einen Graphen $G = (V, E)$ und $v \in V$.

Aufgabe 7 (Binpacking)

[10 Punkte]

Sei $x = (a_1, \dots, a_n, b)$ eine Eingabe für die Variante „Berechnung des optimalen Parameters“ des BINPACKING-Problems. Betrachten Sie nun den Algorithmus BEST FIT, der die Lösung dieses Problems approximiert:

Lege der Reihe nach die Objekte a_1, \dots, a_n nach folgender Regel in einen Behälter: Lege das Objekt a_j in einen bisher benutzten Behälter mit der geringsten Restkapazität, der noch genügend Platz bietet. Existiert kein solcher, lege a_j in einen bisher unbenutzten Behälter.

Sei $\text{opt}(x)$ die kleinste Anzahl von benötigten Behältern und $\text{BEST FIT}(x)$ die Anzahl der Behälter, die BEST FIT benutzt. Zeigen Sie, dass $\text{BEST FIT}(x) \leq 2\text{opt}(x) + 1$.

Anhang: Definitionen

ε -CLOSENESS = $\{(x_1, \dots, x_n) \mid n \in \mathbb{N}, x_1, \dots, x_n \in \mathbb{R}, \exists 1 \leq i < j \leq n : |x_i - x_j| \leq \varepsilon\}$

INDEPENDENT SET = $\{\langle G \rangle \text{bin}(k) \mid k \in \mathbb{N}, G = (V, E), \exists V' \subseteq V \text{ mit } |V'| \geq k : u, v \in V' \Rightarrow (u, v) \notin E\}$

2-PROCESSOR SCHEDULING: Die Sprache 2-PROCESSOR SCHEDULING beschreibt folgendes Entscheidungsproblem: Können mit zwei gleichen Prozessoren n Aufträge mit den Kosten $\ell_1, \dots, \ell_n \in \mathbb{N}$ innerhalb einer Zeitschranke D abgearbeitet werden?

PARTITION = $\{(a_1, \dots, a_n) \mid n \in \mathbb{N}, a_1, \dots, a_n \in \mathbb{N}, \exists I \subseteq \{1, \dots, n\} : \sum_{i \in I} a_i = \sum_{i \notin I} a_i\}$

Vertex-Cover-Problem: Gegeben ein Graph $G = (V, E)$. Aufgabe: Finde eine möglichst kleine Menge $V' \subseteq V$, die alle Kanten überdeckt, das heißt $\forall (u, v) \in E : \{u, v\} \cap V' \neq \emptyset$. V' heißt *Knotenüberdeckung*.

BINPACKING = $\{(a_1, \dots, a_n, b, k) \mid n \in \mathbb{N}, a_1, \dots, a_n, b, k \in \mathbb{N}, \exists r_1, \dots, r_n \in \{1, \dots, k\} : \forall i \in \{1, \dots, k\} : \sum_{j \text{ mit } r_j = i} a_j \leq b\}$

VIEL ERFOLG!