Übungen zur Komplexitätstheorie — Wintersemester 01/02

Johannes Köbler/Olaf Beyersdorff

Übungsblatt 1

Aufgabe 1 (schriftlich, 10 Punkte)

Zur Erinnerung: Für zwei Funktionen $f, g : \mathbb{N} \to \mathbb{N}$ schreiben wir $f(n) \in O(g(n))$, falls es Konstanten c und n_0 gibt, so dass für alle $n \geq n_0$ gilt: $f(n) \leq c \cdot g(n)$. Wir schreiben $f(n) \in \Omega(g(n))$, falls umgekehrt $g(n) \in O(f(n))$. Schließlich ist $f(n) \in \Theta(g(n))$, falls sowohl $f(n) \in O(g(n))$ als auch $f(n) \in \Omega(g(n))$ gilt. Betrachten Sie jeweils zwei der folgenden Funktionen, und bestimmen Sie, ob (i) $f(n) \in O(g(n))$, (ii) $f(n) \in \Omega(g(n))$, oder ob (iii) $f(n) \in \Theta(g(n))$ gilt:

(a) n^2 (b) n^3 (c) $n^2 \log n$
(d) 2^n (e) n^n (f) $n^{\log n}$
(g) 2^{2^n} (h) $2^{2^{n+1}}$ (j) n^2, falls n gerade ist, 2^n sonst

Aufgabe 2

Betrachten Sie eine Turingmaschine, die ein *zweidimensinales* Band zur Verfügung hat. Der Schreib-Lesekopf kann sich also auch nach oben und unten bewegen.

a) Beschreiben Sie die Überführungsfunktion.

b) Zeigen Sie, wie eine solche Turingmaschine mit einer "normalen" Turingmaschine simuliert werden kann.

Aufgabe 3

Eine Turingmaschine heißt *blind* (engl. oblivious: vergesslich, blind) falls die Kopfposition zu jedem Zeitpunkt t der Berechnung nur von der Länge der Eingabe $|x|$ (und natürlich auch t), nicht aber von der Eingabe x selbst abhängt. Zeigen Sie, dass jede Turingmaschine von einer blinden Turingmaschine simuliert werden kann.

Literatur:

J. Balcázar, J. Diaz, J. Gabarró. *Structural Complexity I+II,* Springer, 1995.
C. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1994

Übungen zur Komplexitätstheorie — Wintersemester 01/02

Johannes Köbler/Olaf Beyersdorff

Übungsblatt 2

Aufgabe 4

Betrachten Sie die Menge der Palindrome $L = \{x \in \Sigma^* \mid x = x^R\}$, wobei x^R das Wort ist, bei dem die Symbole von x in umgekehrter Reihenfolge aufgeschrieben sind. Beschreiben Sie sowohl eine 1-TM als auch eine 2-TM, die L entscheidet.

Aufgabe 5 (schriftlich, 10 Punkte)

Zeigen Sie: Jede $s(n)$-platzbeschränkte Offline-k-TM kann von einer $s(n)$-platzbeschränkten Offline-2-TM simuliert werden.

Aufgabe 6

Zeigen Sie: Jede Sprache, die von einer k-NTM in $f(n)$ vielen Schritten entschieden wird, kann auch von einer 2-NTM in $\mathcal{O}(f(n))$ vielen Schritten entschieden werden.

Aufgabe 7

Sei $\mu = M_0, M_1, \ldots$ eine Aufzählung aller Turingmaschinen, die über dem Alphabet $\{0,1\}$ kodiert seien. Für jedes Wort $x \in \{0,1\}^*$ bezeichne

$$K_\mu(x) = \min\{|M_j| \mid M_j(\epsilon) = x\}$$

die *Kolmogorov-Komplexität* von x bezüglich μ. Hierbei bezeichne ϵ das leere Wort, d.h. $K_\mu(x)$ ist die Länge der kleinsten (Kodierung einer) Turingmaschine, die *ohne* Eingabe das Wort x ausgibt. Zeigen Sie:

a) Es gibt eine Aufzählung μ_0, so dass es für alle Aufzählungen μ eine Konstante c gibt, so dass für alle Wörter $x \in \{0,1\}^*$ gilt:

$$K_{\mu_0}(x) \leq K_\mu(x) + c.$$

Hinweis: Benutzen Sie eine universelle TM.

Für die beiden folgenden Teilaufgaben definieren wir $K(x) = K_{\mu_0}(x)$.

b) Es gibt eine Konstante c, so dass für alle Wörter $x \in \{0,1\}^*$ gilt: $K(x) \leq |x| + c$.

c) Für alle $n \geq 1$ gibt es ein Wort x der Länge n mit $K(x) \geq n$.

Übungsblatt 3

Aufgabe 8 (schriftlich, 10 Punkte)

Sei M eine 1-TM, die die Sprache der Palindrome (siehe Aufgabe 4) entscheidet. Wir möchten zeigen, dass M hierzu Zeit $\Omega(n^2)$ benötigt. Erreicht M bei Eingabe x eine Konfiguration (q, u, av) und führt diese in die Konfiguration (q', u', v') über, so heißt das Paar (q, a) eine *i-Überquerung* von $M(x)$, falls

- $|u| = i - 1$ und $|u'| = i$ oder
- $|u| = i$ und $|u'| = i - 1$

gilt. Die Folge $S_i(M, x) = (q_1, a_1), \ldots, (q_m, a_m)$ aller von $M(x)$ der Reihe nach ausgeführten i-Überquerungen heißt *i-Überquerungsfolge* von $M(x)$ (engl. crossing-sequence). Für $y \in \Sigma^n$ sei $t(y)$ der Zeitverbrauch $time_M(y0^ny^R)$ von M bei Eingabe des Palindroms $x = y0^ny^R$. Zeigen Sie:

a) Es gibt eine Zahl i, $n + 1 \leq i \leq 2n$, so dass $S_i(M, x)$ die Länge $m \leq \frac{t(y)}{n}$ hat.

b) Das Wort y ist eindeutig durch Angabe von M, $S_i(M, x)$, n und i beschreibbar.

c) $K(y) \in O(\frac{t(y)}{|y|} + \log |y|)$.

d) M benötigt Zeit $\Omega(n^2)$.

Aufgabe 9

Der *Kleene-Stern* einer Sprache L ist definiert durch

$$L^* = \{x_1 \ldots x_k \mid k \geq 0 \text{ und } x_1, \ldots, x_k \in L\}.$$

Zeigen Sie, dass die Komplexitätsklassen P und NP abgeschlossen sind unter

a) Vereinigung,

b) Schnitt und

c) dem Kleene-Stern.

Übungen zur Komplexitätstheorie Wintersemester 01/02
Johannes Köbler/Olaf Beyersdorff

Übungsblatt 4

Aufgabe 10

Zeigen Sie: Es gibt eine Sprache L, so dass es für jede $s(n)$-platzbeschränkte TM M mit $L = L(M)$ eine $s'(n)$-platzbeschränkte TM M' mit $L = L(M')$ gibt, wobei $s'(n) \leq \log s(n)$ für fast alle n.
Hinweis: Definieren Sie die Funktion T durch $T(1) = 2$ und $T(n+1) = 2^{T(n)}$, und konstruieren Sie (bezüglich einer geeigneten Aufzählung aller TMs $M_1, M_2, \dots$) eine Sprache L mit den beiden folgenden Eigenschaften:

a) Für alle $k \geq 1$ gibt es eine TM M mit $L = L(M)$, die für Eingaben der Länge n höchstens $T(n-k)$ Platz benötigt.

b) Falls $L = L(M_i)$ gilt, dann benötigt M_i wenigstens $T(n-i)$ Platz für Eingaben der Länge n.

Aufgabe 11

Zeigen Sie: Es gibt eine rekursive Funktion S mit DSPACE($S(n)$) = PSPACE.
Hinweis: Definieren Sie $S(n)$ bezüglich einer Aufzählung aller TMs $M_1, M_2, \dots$, so dass die beiden folgenden Bedingungen gelten:

a) Für alle k und für fast alle n gilt $S(n) \geq n^k$.

b) Falls der Platzbedarf $S_i(n)$ von M_i für alle k die Bedingung $S_i(n) > n^k$ für unendlich viele n erfüllt, dann gilt auch $S_i(n) > S(n)$ für unendlich viele n.

Aufgabe 12 (schriftlich, Blum Komplexität, 10 Punkte)

Sei Φ eine Funktion, die (geeignete Kodierungen von) TMs M und Eingaben x in die natürlichen Zahlen abbildet. Dann heißt Φ ein *Komplexitätsmaß*, falls die beiden folgenden Axiome erfüllt sind:
Axiom 1: $\Phi(M, x)$ ist genau dann definiert, wenn $M(x)$ definiert ist.
Axiom 2: Es ist rekursiv entscheidbar, ob $\Phi(M, x) = m$ gilt.
Sind die folgenden Maße Komplexitätsmaße?

a) $\mathsf{time}_M(x)$ und $\mathsf{space}_M(x)$ für TMs und NTMs.

b) $\mathsf{ink}_M(x)$: Anzahl der Ersetzungen eines Symbols durch ein anderes Symbol.

c) $\mathsf{carbon}_M(x)$: Anzahl der Ersetzungen eines Symbols durch das *gleiche* Symbol.

Übungen zur Komplexitätstheorie   Wintersemester 01/02
Johannes Köbler/Olaf Beyersdorff

Übungsblatt 5

Aufgabe 13 (schriftlich, 10 Punkte)

Beweisen Sie den Platzhierarchiesatz: Ist f eine echte Komplexitätsfunktion und gilt

$$\liminf_{n\to\infty} \frac{g(n)}{f(n)} = 0,$$

dann ist

$$\mathrm{DSPACE}(f(n)) \setminus \mathrm{DSPACE}(g(n)) \neq \emptyset.$$

Aufgabe 14

Seien Φ und Φ' zwei Komplexitätsmaße. Dann gibt es eine rekursive Funktion r, so dass für alle Turingmaschinen M und für fast alle x gilt: $\Phi(M,x) \leq r(x, \Phi'(M,x))$.

Aufgabe 15

Zeigen Sie, dass die folgenden Sätze für beliebige Komplexitätsmaße gelten.

a) Gap-Theorem (siehe Vorlesung)

b) Speedup-Theorem (Aufgabe 10)

c) Union-Theorem (Aufgabe 11)

Aufgabe 16

Zeigen Sie, dass die $\leq_{\log}$-Reduzierbarkeit reflexiv und transitiv ist.

Aufgabe 17

Zwei Sprachen A und B heißen *äquivalent* ($A \equiv_{\log} B$), falls $A \leq_{\log} B$ und $B \leq_{\log} A$ gilt. Der *Grad* einer Sprache A ist die Klasse aller Sprachen B, die äquivalent zu A sind. Aus wie vielen verschiedenen Graden besteht die Klasse L (logarithmischer Platz)?

Übungen zur Komplexitätstheorie Wintersemester 01/02

Johannes Köbler/Olaf Beyersdorff

Übungsblatt 6

Aufgabe 18

Zeigen Sie:

a) Die Komplexitätsklassen **L**, **NL**, **P**, **NP**, **co-NP**, **PSPACE**, **EXP** und **EXPSPACE** sind unter $\leq$ abgeschlossen.

b) Falls es eine **NP**-vollständige Menge in **P** gibt, dann ist $\mathbf{P} = \mathbf{NP}$.

c) Falls es eine **NL**-vollständige Menge in **L** gibt, dann ist $\mathbf{L} = \mathbf{NL}$.

Aufgabe 19 (schriftlich, 10 Punkte)

Zeigen Sie, daß die Komplexitätsklassen **NP** und **E** verschieden sind. Betrachten Sie hierzu die Sprache

$$K := \{\langle M, x, 1^t\rangle \mid \text{ M kodiert eine } k\text{-TM, die } x \text{ in } \leq 2^t \text{ Schritten akzeptiert}\}$$

und zeigen Sie:

a) K ist **E**-vollständig.

b) Eine Sprache ist genau dann **E**-hart, wenn sie **EXP**-hart ist.

c) **E** ist nicht unter $\leq$ abgeschlossen.

Aufgabe 20

Zeigen Sie, daß aus $\mathbf{E} \neq \mathbf{NE}$ folgt, daß $\mathbf{P} \neq \mathbf{NP}$ gilt (*downward separation*). Zeigen Sie hierzu die Äquivalenz

$$\mathbf{E} \neq \mathbf{NE} \iff \text{ es gibt eine tally Menge } T \in \mathbf{NP} - \mathbf{P}.$$

Eine Sprache T heißt **tally**, wenn sie über einem einelementigen Alphabet gebildet werden kann (also z.B. $T \subseteq \{0\}^\star$).
Hinweis: Betrachten Sie die "tally Version" einer Sprache A,

$$tally(A) = \{0^{num(1x)} \mid x \in A\},$$

wobei $num(1x)$ die durch die Binärzahl $1x$ repräsentierte natürliche Zahl ist.