

Übung zur Vorlesung BeKo 2013/14
Berechenbarkeit und Komplexität / Theoretische Informatik

Sie können für die 4 Aufgaben auf diesem Übungszettel insgesamt bis zu **20** Punkte erhalten. Genauere Angaben zur Abgabe der Übungszettel finden Sie auf der letzten Seite nach den Aufgaben.

Aufgabe 34 *Eigenschaften von Turingmaschinen* (5 Punkte)

Bestimmen Sie für die folgenden Eigenschaften von deterministischen Turingmaschinen, ob sie entscheidbar oder nicht entscheidbar sind.

Bitte geben Sie auch eine sinnvolle Begründung/einen Beweis für Ihre Antwort. Für Antworten ohne sinnvolle Begründung gibt es *keine* Punkte !

- (a) Die TM terminiert nach spätestens 100 Schritten, wenn sie auf dem leeren Band gestartet wird. (1 Punkt)
- (b) Die TM besucht maximal 100 Zellen des Bandes, wenn sie auf dem leeren Band gestartet wird. (2 Punkte)
- (c) Die TM schreibt irgendwann mal ein „a“ auf das Band. (2 Punkte)

Aufgabe 35 *Eine nicht entscheidbare Sprache* (8 Punkte)

Gegeben sei folgendes Entscheidungsproblem P : Bei Eingabe von zwei Turingmaschinen soll entschieden werden, ob die erste auf dem leeren Wort (ε) anhält, und die zweite bei Eingabe des leeren Wortes *nicht* anhält. Formal:

$$P = \{w_1 \# w_2 \mid M_{w_1} \text{ hält auf } \varepsilon, M_{w_2} \text{ hält nicht auf } \varepsilon\}$$
$$\chi_P(w_1 \# w_2) = 1 \Leftrightarrow M_{w_1} \text{ hält auf } \varepsilon \text{ und } M_{w_2} \text{ hält nicht auf } \varepsilon$$

- (a) Zeigen Sie, dass P nicht semi-entscheidbar ist. (4 Punkte)
- (b) Zeigen Sie, dass das Komplement von P nicht semi-entscheidbar ist. (4 Punkte)

(*Hinweis:* Das Halteproblem auf dem leeren Wort H_0 ist semi-entscheidbar, aber das Komplement $\overline{H_0}$ von H_0 ist nicht semi-entscheidbar.

Zusätzlich dürfen Sie folgenden Satz verwenden: Falls $A \leq B$ gilt und A nicht semi-entscheidbar ist, so ist auch B nicht semi-entscheidbar)

Aufgabe 36 Die Landau-Notation (4 Punkte)

Um die Laufzeiten von Algorithmen zu klassifizieren, wird die Landau-Notation benutzt. Mit dieser Notation stellt man jeweils zwei Funktionen zueinander in Relation. Für die Komplexitätstheorie wird sehr häufig die O -Notation benutzt (*groß-O* Notation), die nur einen Teil der Landau-Notation ausmacht.

Für zwei gegebene Funktionen f und g ($f, g: \mathbb{R} \rightarrow \mathbb{R}$) ist $f \in O(g)$ so definiert:

$$f \in O(g) \Leftrightarrow \exists C > 0 \exists N \forall x > N : |f(x)| \leq C \cdot |g(x)| \quad N, C \in \mathbb{R}$$

Um den Umgang mit der O -Notation zu üben, beweisen Sie bitte folgende Aussagen. Geben Sie bei den Beweisen auch ein mögliches C und N an, mit denen die Definition erfüllt wird.

- (a) $f(x) = \sqrt{|x|} + 10$, $g(x) = x^3$. Zeigen Sie: $f \in O(g)$. (1 Punkt)
- (b) $f \in O(g)$, $h \in O(g)$, $\varphi(x) = f(x) + h(x)$. Zeigen Sie: $\varphi \in O(g)$. (1 Punkt)
- (c) $f(x) = 3^x$, $g(x) = 2^x$. Zeigen Sie: $f \notin O(g)$. (1 Punkt)
- (d) $f(x) = \sum_{i=0}^n (k_i \cdot x^i)$ $k_i \in \mathbb{R}$ (f ist also ein beliebiges Polynom vom Grad n). Zeigen Sie: $f \in O(x^n)$ (1 Punkt)

Aufgabe 37 Unäres vs. logarithmisches Kostenmaß (3 Punkte)

Für die Laufzeit eines Algorithmus' ist es sehr wichtig sich zu überlegen, in Bezug zu welcher Größe die Laufzeit angegeben wird. Wenn ein Programm eine natürliche Zahl als Eingabe bekommt, so könnte man z.B. die Laufzeit in Abhängigkeit von dieser Zahl berechnen (unäres Kostenmaß). Da man meistens eine Zahl in Binär-Codierung erhält, könnte man auch die Anzahl der Bits, die man zur Darstellung dieser Zahl braucht, als Bezugswert nehmen (logarithmisches Kostenmaß). Diese Zahl würde (bei Binär-Codierung) ja auch genau der Länge der Eingabe für die Turingmaschine entsprechen. Wie groß der Unterschied bei diesen beiden Kostenmaßen ist, sollen Sie sich an dieser Aufgabe verdeutlichen:

Eingabe ist eine natürliche Zahl n der Form $n = 2 \cdot k + 3$ mit $k \in \mathbb{N}_0$. Die Zahl n kann durch b Bits codiert werden. Der Algorithmus soll nun eine natürliche Zahl $x > 0$ bestimmen, so dass $2^x \text{ MOD } n = 1$ ist. Dazu geht er wie folgt vor: Er probiert für x alle Werte zwischen 2 und $n - 1$ durch, indem er im ersten Schritt $2^2 \text{ MOD } n$ berechnet und auf Gleichheit mit 1 prüft. In jedem weiteren Schritt nimmt er das Ergebnis der MOD Rechnung des vorhergehenden Schrittes, verdoppelt dieses und prüft es wieder auf Gleichheit mit 1. Für die MOD Rechnung, die Verdopplung und die Gleichheitsprüfung werden zusammen $O(b)$ Schritte benötigt. Der Algorithmus gibt die erste Zahl x aus, bei der die Gleichheitsprüfung *wahr* zurückgibt. Die Bedingung, dass n von der Form $n = 2 \cdot k + 3$ mit $k \in \mathbb{N}_0$ ist, dient dazu, sicherzustellen, dass der Algorithmus eine Lösung findet (warum stellt diese Bedingung das sicher?).

- (a) Geben Sie die (worst-case) Laufzeit des Algorithmus in unärem Kostenmaß, das heißt in Abhängigkeit von n , an. (1 Punkt)

- (b) Geben Sie die (worst-case) Laufzeit des Algorithmus in logarithmischen Kostenmaß, das heißt in Abhängigkeit von b an. (2 Punkte)

Begründen Sie Ihre Antworten! Es reicht nicht, einfach nur die Laufzeiten anzugeben, man muss auch erläutern, wie diese Laufzeit zustande kommt.

Hinweise zur Abgabe

Die Hausaufgaben zu diesem Übungsblatt müssen bis spätestens Montag, den 20. Januar 2014 um 12:00 Uhr abgegeben werden. Eine Bearbeitung der Übungsaufgaben *zu zweit* ist möglich. In diesem Fall geben Sie bitte *nur eine Abgabe (auch bei Moodle!)* ab und schreiben Sie den *Namen* und die *Matrikelnummern beider Personen* auf Ihre Abgabe bzw. in das PDF-Dokument. Ihr Name, Ihre Matrikelnummer, Ihre Gruppennummer und das Fach (BeKo 2013/14) müssen deutlich sichtbar auf die Hausaufgabe geschrieben sein. Sie können Ihre Aufgaben an folgenden Stellen abgeben:

Campus Duisburg:

Der mit *Berechenbarkeit und Komplexität / Theoretische Informatik* beschriftete Briefkasten neben Raum LF259.

Moodle:

Sie können ihre Hausaufgaben auch gerne per MOODLE¹ abgeben. Achten Sie aber darauf, dass Sie ihre Hausaufgaben als eine einzelne pdf-Datei hochladen. Wenn ein Betreuer Ihre Abgabe nicht öffnen kann (Formate wie docx o.ä.), bringt das nur unnötige Verzögerungen mit sich.

¹<http://moodle2.uni-due.de/course/view.php?id=1338>