Prof. Dr. Elmar Schömer
André Müller

Datenstrukturen und effiziente Algorithmen · WS 2014/15

JOHANNES GUTENBERG UNIVERSITÄT MAINZ

Übungsblatt 14

Bearbeitung freiwillig

Dieses Übungsblatt enthält Aufgaben, wie sie (Umfang und Schwierigkeitsgrad betreffend) typischerweise in den Klausuren gestellt werden.

Am Mittwoch, den 18.02. (Aschermittwoch) findet eine Zusatzübung von 14-16 Uhr in Raum 03-428 (großer Vorlesungsraum der Informatik) statt.

14.1 Asymptotische Laufzeit und $\mathcal{O}$ -Notation

Begründen Sie kurz und präzise, ob die folgenden Aussagen richtig oder falsch sind.

a) Es gilt $f(n) = 2^{2n} \in \mathcal{O}(2^n)$, mit $n \in \mathbb{N}$.

b) $f(n) = n, n \in \mathbb{N}$, dann gilt $f(n) \in \mathcal{O}\big((\log n)^{\log n}\big)$

14.2 Rekurrenzen

Geben Sie für die folgenden Rekurrenzen jeweils eine explizite Abschätzung des Wachstumsverhaltens in $\mathcal{O}$-Notation an. Es sei jeweils $n \in \mathbb{N}$.

a) $T(n) = 2\,T\left(\left\lfloor \frac{n}{4} \right\rfloor\right) + 4\sqrt{n},\;\; T(1) = 1$

b) $T(n) = T\big(\lfloor \sqrt{n} \rfloor\big) + \log n\,,\;\; T(1) = 0$

14.3 Suchen & Sortieren

Stimmt folgende Aussage? Begründen Sie!

In einem binären Suchbaum kann man zu jedem Element x das nächst kleinere Element $y \leq x$ im Worst-Case in konstanter Laufzeit finden.

14.4 Hashing

Gegeben ist eine Hashtabelle mit m Buckets, die eine 1-universelle Hashfunktion $h: U \to H$ verwendet und Kollisionen mittels Verkettung behandelt. Die Tabelle wurde mit n Schlüsseln befüllt. Wie sollte m in Abhängigkeit von n gewählt werden, so dass die erwartete Gesamtanzahl an Kollisionen in $\mathcal{O}(1)$ liegt?

14.5 Graphenalgorithmen

a) In einem ungerichteten Baum $T = (V, E)$ mit reellen Kantengewichten $w: E \to \mathbb{R}$ kann man den längsten Weg von einem Knoten s zu einem anderen Knoten t nicht in linearer Zeit $\mathcal{O}(|V| + |E|)$ finden.

b) Der Algorithmus von Kruskal liefert für den selben Eingabegraphen $G = (V, E)$ mit reellen Kantengewichten $w: E \to \mathbb{R}$ immer den gleichen Spannbaum wie der Algorithmus von Prim.

c) Gegeben sei ein gerichteter Graph $G = (V, E)$ mit Kantengewichten $w: E \to \mathbb{R}$ sowie ein Knoten $s \in V$. Sei $G' = G$ mit Kantengewichten $w': E \to \mathbb{R}$. Für alle $e \in E$ gelte nun $w'(e) = w(e)^2$. Dann sind die kürzesten Wege von s zu allen anderen Knoten in G' und in G gleich.

d) In einem gerichteten Graph mit positiven Kantengewichten, der einen Zyklus enthält, kann es sein, dass der Dijkstra-Algorithmus eine Kante mehrfach relaxiert.

14.6 2-Färbbarkeit

Sei $G = (V, E)$ ein ungerichteter Graph. Wir nennen G k-färbbar, wenn man seine Knoten so mit k Farben markieren kann, dass keine zwei benachbarten Knoten die gleiche Farbe haben. Beschreiben Sie einen Algorithmus, der mittels Breitensuche feststellt, ob ein Graph 2-färbbar ist.

14.7 Dicke Pfade

Gegeben ist ein zusammenhängender, gerichteter und gewichteter Graph $G = (V, E, w)$. Wir definieren die *Dicke* eines Pfades P auf G als das Maximum aller Kantengewichte in P. Geben Sie einen <u>effizienten</u> Algorithmus an, der für G und zwei Vertices $u, v \in V$ die minimale Dicke findet, die ein Pfad von u nach v haben kann. Geben Sie die Worst-Case-Laufzeit Ihres Algorithmus an und begründen Sie diese <u>kurz</u>.

14.8 Wackeliges Array

Wir sagen ein Array A mit $2n + 1$ Elementen sei *wackelig*, wenn gilt:

$$A[1] \leq A[2] \geq A[3] \leq A[4] \geq \cdots \leq A[2n] \geq A[2n+1].$$

Gegeben sei nun ein unsortiertes Array B mit $2n + 1$ reellen Zahlen $B[1]$ bis $B[2n + 1]$. Beschreiben Sie einen Algorithmus, der eine Permutation A von B ausgibt, so dass A ein wackeliges Array ist. Dieser sollte eine erwartete (oder, falls möglich, sogar Worst-Case) Laufzeitkomplexität von $\mathcal{O}(n)$ haben. Begründen Sie die Laufzeitkomplexität Ihres Algorithmus kurz, aber nachvollziehbar!