Höhere Algorithmik, WS 2014/15 — 2. Übungsblatt

20 Gummipunkte. Bearbeitung: Dienstag, 28. Oktober 2012
Allgemeiner Hinweis: Denken Sie daran, dass Ihre Lösung für eineX andereX StudeX, diX die Lösung korrigiert, verständlich sein muss.

---

9. Kuchenteilen, 0 Punkte

   Zeigen Sie, dass der in der Vorlesung behandelte randomisierte divide-and-conquer-Algorithmus zum Kuchenteilen im Erwartungswert mit linear vielen Schnitten auskommt.

10. Schmutzige Tricks im Einheitskostenmaß, 10 Punkte

    Für drei ganzzahlige Vektoren $x = (x_1, x_2, \dots, x_n)$ $y = (y_1, y_2, \dots, y_n)$ und $z = (z_1, z_2, \dots, z_n)$ mit $|x_i|, |y_i|, |z_i| \le M$ und einen Wert $u > 4M$ betrachten wir die Zahlen $a = x_1 u^{n-1} + x_2 u^{n-2} + \cdots + x_{n-1} u + x_n$, $b = y_1 u^{n-1} + y_2 u^{n-2} + \cdots + y_{n-1} u + y_n$ und $c = z_1 u^{n-1} + z_2 u^{n-2} + \cdots + z_{n-1} u + z_n$.

    Zeigen Sie: $a + b = c$ genau dann, wenn $x_i + y_i = z_i$ für alle i ist;

    (Man kann also die Addition beliebig langer Vektoren in einem Schritt ausführen. Warum funktioniert das nicht mehr für reelle Vektoren?)

11. Programmieren einer Registermaschine, 10 Punkte

    Schreiben Sie ein Programm für die Registermaschine, das bei Eingabe von n und einem Vektor $x = (x_1, x_2, \dots, x_n)$ die größte unter den n Teilsummen $x_1 + x_2 + \cdots + x_i$ $(1 \le i \le n)$ berechnet. Ihr Programm soll so dokumentiert sein, dass es für einen Menschen verständlich ist. Beschreiben Sie den Algorithmus zuerst in Pseudocode.

    Bestimmen Sie die asymptotische Laufzeit und den Speicherplatz (a) im Einheitskostenmaß und (b) im logarithmischen Kostenmaß.

    Testen Sie Ihren Algorithmus mit dem RAM-Emulator[1], den Sie auf der Vorlesungsseite im Netz finden.

12. Median und Mittelwert, 0 Punkte

    (a) Berechnen Sie den Mittelwert und den Median (den Wert, der in sortierter Reihenfolge in der Mitte steht) der ersten 19 Zweierpotenzen, die größer als 5 sind $\{8, 16, 32, \dots, 2^{21}\}$.

    (b) Wenn 19 reelle Zahlen zwischen 3 und 10 gegeben sind, wie groß kann dann die Differenz zwischen Mittelwert und Median höchstens sein?

13. (0 Punkte) Berechnen Sie die ersten 10 Glieder der durch die Rekursion

    $$a_n = a_{n-1} + a_{n-2} + a_{n-3} + 1, \text{ für } n > 3$$

    und $a_1 = a_2 = a_3 = 1$ gegebene Folge.

14. (0 Punkte) Die Funktion f berechnet das n-te Glied a_n der Folge aus Aufgabe 13:

    ```
    def f(n):
        if n ≤ 3: return 1
        else: return f(n − 1) + f(n − 2) + f(n − 3) + 1
    ```

[1] http://www.farbtrommel.de/ram/

Bestimmen Sie die Anzahl der Aufrufe der Funktion f bei Berechnung von $f(n)$ in Abhängigkeit von n, inklusive aller rekursiven Aufrufe. *Hinweis*: Für kleine n bestimmen, Vermutung formulieren, und mit Induktion beweisen. Eine explizite Formel für das Ergebnis ist nicht gefordert. Können Sie das Ergebnis einfach erklären?

15. Fehlstände und Bubble-Sort, 0 Punkte

    (a) Die Zahlenfolge

    $$54, 56, 34, 89, 38, 57, 67, 87$$

    soll in aufsteigende Folge sortiert werden. Dabei dürfen immer nur zwei *benachbarte* Zahlen vertauscht werden. Wie viele paarweise Vertauschungen sind nötig?

    (b) Zeigen Sie, dass die Anzahl der Vertauschungen nicht davon abhängt, an welcher Stelle man beginnt, wenn man sich nicht blöd anstellt. Wie kann man die Anzahl der nötigen Vertauschungen leicht bestimmen, ohne die Vertauschungen durchzuführen?

    (c) Zusatzfrage: Wie viele paarweise Vertauschungen sind nötig, wenn man beliebige zwei Zahlen vertauschen darf?

16. Rekursionsbaum, 0 Punkte

    Berechnen Sie eine asymptotische Schranke für die durch die Rekursion

    $$T(n) \leq 2T(\lfloor n/2 \rfloor) + O(n^2), \text{ für } n > n_0$$

    und $T(n) \leq C$ für $n \leq n_0$ gegebene Laufzeit eines Algorithmus, indem Sie den Rekursionsbaum aufstellen und ebenenweise aufsummieren.

    Wenn Sie der Einfachheit halber zunächst annehmen, dass n eine Zweierpotenz ist, dann *begründen* Sie auch, dass Ihr Ergebnis auch für andere Werte gilt.

17. 20 Fragen, 0 Punkte

    Jemand denkt sich eine ganze Zahl n zwischen 2 und 166 aus. Wieviele Ja-Nein-Fragen benötigt man im schlimmsten Fall, um den kleinsten Teiler (größer als 1) von n herauszufinden?[2]

[2] http://domino.research.ibm.com/Comm/wwwr_ponder.nsf/challenges/November2009.html