Klausur Algorithmik SS 2011

Iwanowski 05.09.2011

Hinweise:

Bearbeitungszeit: 100 Minuten

Erlaubte Hilfsmittel: keine

Bitte notieren Sie Ihre Antworten ausschließlich auf dem Aufgabenblatt! Bei Bedarf benutzen Sie die Rückseite! Für Skizzen und Entwürfe steht ebenfalls die Rückseite zur Verfügung. Entwürfe, die nicht gewertet werden sollen, sind durchzustreichen.

Insgesamt gibt es 40 Bewertungseinheiten (BE) zu erzielen. Zum Bestehen benötigen Sie mindestens 20 BE.

Viel Erfolg!

Aufgabe 1: Thema: Einführung in die formale Behandlung von Algorithmen (2 BE)

Ordnen Sie folgende Komplexitätsklassen der Größe nach:

a) $O(n^2 (\log_2 n)^3)$

b) $O(n^2 (\log_3 n)^2)$

c) $O(n!)$

d) $O(n^{3,01})$

e) $O(n^3 (\log_2 n)^2)$

f) $O(n^3 (\log_3 n)^2)$

g) $O(n)$

Geben Sie eine lineare Kette an, in der Sie kenntlich machen, welche Klasse die andere echt umfasst und welche gleich ist. Sie können zur Schreibersparnis mit den Buchstaben a bis g arbeiten.

Aufgabe 2: Thema: Such- und Sortieralgorithmen (6 BE)

a) Definieren Sie eine Rekursionsgleichung für eine obere Schranke von Quicksort im schlechtesten Fall und beweisen Sie das daraus resultierende asymptotische Laufzeitverhalten mit vollständiger Induktion. (5 BE)

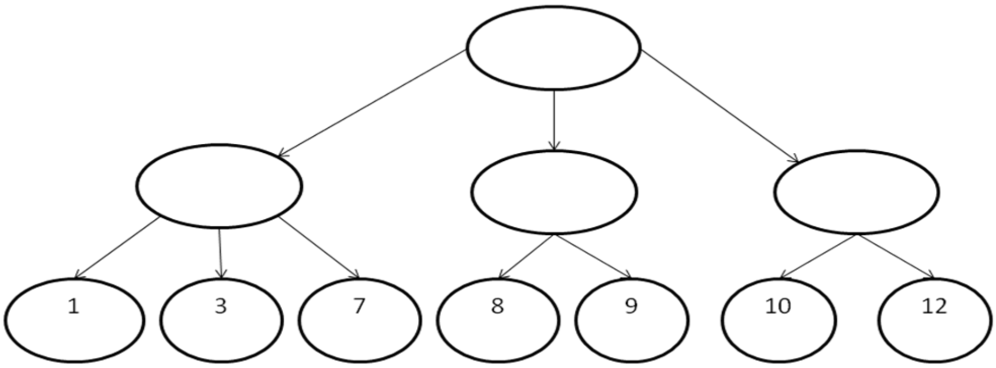
b) Durch welches Implementierungsdetail im Rekursionsschritt kann man es für jede Eingabe sehr unwahrscheinlich machen, dass Quicksort sich tatsächlich so wie in a) verhält? (zu den tatsächlichen Wahrscheinlichkeiten müssen Sie sich nicht äußern)(1 BE)

Aufgabe 3: Thema: Lösungen des Wörterbuchproblems (5 BE)

a) Schreiben Sie in die inneren Knoten des unten angegebenen 2-3-Baums die Schlüssel hinein! (1 BE)

b) Stellen Sie den 2-3-Baum dar, der entsteht, wenn Sie im unten angegebenen Baum das Element 6 einfügen ! Akualisieren Sie auch die Schlüssel der inneren Knoten!(2 BE)

c) Löschen Sie jetzt das Element 10 und stellen Sie den resultierenden 2-3-Baum dar! Aktualisieren Sie auch die Schlüssel der inneren Knoten! (2 BE)



Aufgabe 4: Thema: Graphenalgorithmen (7 BE)

a) Welches mengentheoretische Problem löst ein Union-Find-Algorithmus? (1 BE)

b) Erklären Sie einen effizienten Algorithmus am Beispiel {{1, 2, 3}, {4, 5}, {6, 7, 8, 9}}, indem Sie zunächst eine geeignete Datenstruktur für die Abspeicherung dieser Menge aufzeichnen und dann die Veränderung dokumentieren, wenn Sie zunächst die erste mit der zweiten Menge und schließlich das Resultat mit der dritten Menge vereinigen. (3 BE)

c) Welche Laufzeit hat der von Ihnen skizzierte Algorithmus für die in a) definierten Operationen, wenn insgesamt n Elemente in der Datenstruktur abgespeichert sind? (2 BE)

d) Für welche graphentheoretische Anwendung wird dieser Algorithmus eingesetzt? (1 BE)

Aufgabe 5: Thema: Graphenalgorithmen (5 BE)

Analysieren Sie den folgenden Algorithmus von Floyd-Warshall:

for $i = 1, \ldots, n$ **do**
    for $j = 1, \ldots, n$ **do**
        $d_{ij}^{(0)} = \begin{cases} c(i,j): & \text{falls } (i,j) \in E \\ \infty: & \text{sonst} \end{cases}$
    end for
end for
for $k = 1, \ldots, n$ **do**
    for $i = 1, \ldots, n$ **do**
        for $j = 1, \ldots, n$ **do**
            $d_{ij}^{(k)} = \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
        end for
    end for
end for

a) Welches Problem löst dieser Algorithmus? Wofür stehen die Variablen $d_{ij}^{(k)}$? Geben Sie das in Abhängigkeit von i,j und k an! (3 BE)

b) Geben Sie die asymptotische Gesamtlaufzeit dieses Algorithmus an und begründen Sie das anhand der Teilschritte! (Sie können die Begründung in den oben gegebenen Code hineinschreiben) (2 BE)

Aufgabe 6: Thema: Graphenalgorithmen (4 BE)

a) Geben Sie die drei äquivalenten Bedingungen des Satzes von Ford-Fulkerson für die Flussberechnung an. (3 BE)

b) Beweisen Sie eine der 3 benötigten Implikationen. (1 BE)

Aufgabe 7: Thema: Algorithmische Geometrie (4 BE)

Betrachten Sie das Problem Minimum Spanning Tree in der Ebene:

a) Beschreiben Sie einen naiven Algorithmus zur Lösung dieses Problems und begründen Sie seine Laufzeit! (2 BE)

b) Skizzieren Sie, wie man das Problem mit Hilfe eines bereits erstellten Voronoi-Diagramms effizienter löst und begründen Sie die Laufzeit dieser Lösung.
(Hinweis: Die Laufzeit ist nicht ganz trivial, sondern hängt von einer Eigenschaft des Voronoi-Diagramms ab, welche hier erwähnt werden soll) (2 BE)

Aufgabe 8: Thema: Algorithmische Geometrie (7 BE)

a) Definieren Sie, was ein Voronoi-Diagramm darstellt und nennen Sie die wesentlichen Bestandteile, die für seine Repräsentation im Rechner benötigt werden (Verlinkung zwischen den Bestandteilen muss nicht erwähnt werden). (3 BE)

b) Die Sweep Status Structure in einem Plane-Sweep-Verfahren wird immer rechts durch eine Gerade begrenzt. Charakterisieren Sie die linke Begrenzung für ein Plane-Sweep-Verfahren zur Berechnung eines Voronoi-Diagramm und nennen Sie die einzelnen Bestandteile. (2 BE)

c) In welcher Laufzeit kann man ein Voronoi-Diagramm mit einem Plane Sweep berechnen? Geben Sie konkret an, in welcher Laufzeit man die Sweep Status Structure aktualisieren kann und wie viele Ereignisse berücksichtigt werden müssen (asymptotische Angaben reichen aus). (2 BE)