

# Klausur Algorithmik SS 2010

Iwanowski 19.08.2010

## Hinweise:

**Bearbeitungszeit:** 90 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 (7 BE)

Vergleichen Sie die Algorithmen Quicksort und Mergesort zum Sortieren von  $n$  Objekten im schlechtesten Fall:

a) Stellen Sie für die Laufzeit beider Algorithmen Rekursions(un)gleichungen auf! (2 BE)

b) Geben Sie die asymptotische Gesamtlaufzeit für beide Algorithmen an und beweisen Sie die von Mergesort mit Hilfe von vollständiger Induktion und a)!  
(Hinweis: Es ist hilfreich, dass Sie erst einmal genau definieren, was im Induktionsschluss gezeigt werden muss, exakter als mit O-Notation) (5 BE)

**Aufgabe 2:** Thema: Such- und Sortieralgorithmen

(3 BE)

Bekanntlich ist die Binärsuche bereits ein optimales Verfahren zur Suche eines Eintrags in einem Datenfeld.

- a) Was ist das Ziel eines Verbesserungsverfahrens wie der Interpolationssuche oder quadratischen Binärsuche? Was wird durch diese Verbesserungsverfahren konkret erreicht? (2 BE)

- b) Handelt man sich dadurch auch in gewisser Hinsicht eine Verschlechterung ein? Wenn ja, welche? (1 BE)

**Aufgabe 3:** Thema: Such- und Sortieralgorithmen

(8 BE)

Simulieren Sie den deterministischen Auswahlalgorithmus Select ( $k$ ,  $A$ ) mit linearer Laufzeit an folgender Eingabe:

$$k = 12, A = [13, 2, 19, 15, 1, 28, 29, 16, 11, 37, 4, 31, 17, 33, 3]$$

Beschreiben Sie dafür in Worten die jeweiligen Teilaufgaben und geben Sie das jeweilige Ergebnis der Teilaufgabe an (die Elementarschritte müssen Sie nicht vorführen).

Für jede Teilaufgabe soll die Rekursion abbrechen, wenn das Eingabefeld aus höchstens 6 Elementen besteht.

**Aufgabe 4:** Thema: Graphenalgorithmen

(7 BE)

Analysieren Sie den Algorithmus von Kruskal zur Berechnung eines spannenden Baums:

a) Geben Sie in Worten oder Pseudocode an, wie er funktioniert! (3 BE)

b) Analysieren Sie die in a) geschilderten Teilschritte, indem Sie angeben, mit welcher asymptotischen Laufzeit man diese lösen kann, und geben Sie dann die Gesamtlaufzeit an! (4 BE)

**Aufgabe 5:** Thema: Graphenalgorithmen

(10 BE)

Betrachten Sie das Problem, in einem Graphen mit vorgegebenen Flusskapazitäten den maximalen Fluss zu bestimmen.

a) Definieren Sie genau, was ein Erweiterungsweg ist (auch für Rückwärtskanten) und erklären Sie den Zusammenhang zur Berechnung eines maximalen Flusses! (3 BE)

b) Beweisen Sie eine Richtung der Behauptung von a) in Worten! (2 BE)  
(Hinweis: Überlegen Sie sich, welches der triviale Teil ist und beweisen Sie nur den)

c) Definieren Sie, was ein blockierender Fluss ist, und begründen Sie in Worten, warum ein blockierender Fluss nicht notwendigerweise ein maximaler Fluss sein muss.  
(Anmerkung: Sie dürfen das natürlich auch an einem Beispiel zeigen, aber das ist vermutlich zeitaufwändiger) (2 BE)

d) Welches oben genannte Konzept führt schneller zur Berechnung eines maximalen Flusses? Geben Sie die jeweilige Laufzeit an und begründen Sie, warum die eine besser ist! (3 BE)

**Aufgabe 6:** Algorithmische Geometrie

(5 BE)

Betrachten Sie die Implementierung der Berechnung eines Voronoi-Diagramms mit Hilfe eines Plane-Sweep-Verfahrens:

- a) Geben Sie die Gesamtlaufzeit an, in der das erreicht werden kann! Spezifizieren Sie die Bedeutung aller genannten Variablen in Ihrem Term für die Gesamtlaufzeit (z.B.: Was ist  $n$ ?)! (2 BE)

- b) Normalerweise arbeitet ein Plane Sweep mit zwei parallelen Geraden, die über die Ebene gezogen werden. Für Voronoi-Diagramme muss eine der Geraden durch etwas anderes ersetzt werden: Beschreiben Sie 2 Bestandteile dieser Ersetzung. (3 BE)