

Klausur 30.09.03 DAP2-Teil

Matrikelnummer (Leserlich!): _____

Vorname: _____ Familienname: _____

Ich studiere (Bitte markieren): Informatik / Inform. Lehramt / Inf.technik/ Physik / Mathe
bzw. Sonstiges:

Bitte beachten:

- **Auf jedem Blatt Name und Matrikelnummer gut lesbar eintragen!**
(Die Klausur wird bei der Korrektur „entklammert“.)
- **Nur Papier verwenden, das von uns ausgeteilt wurde!**
(Papiernachschub kann von der Klausuraufsicht erbeten werden.)
- **Nur dokumentenechte Schreibgeräte verwenden!**
(Z.B. Kugelschreiber, nicht Bleistifte.)

Hinweis:

Wir als Klausurveranstalter sind organisatorisch nicht dazu in der Lage, vor bzw. während der Klausur zu überprüfen, ob die Teilnehmer/-innen dazu berechtigt sind, die Klausur mitzuschreiben bzw. ob sie ordnungsgemäß bei der jeweils zuständigen Stelle angemeldet sind. Daher gilt folgendes:

Durch die Teilnahme an der Klausur erkennt der Teilnehmer bzw. die Teilnehmerin an, dass diese unter Vorbehalt stattfindet. Genauer: Die Teilnahmeberechtigung und Anerkennung der Klausur wird erst nach der Klausur von der jeweils zuständigen Stelle überprüft und ist nicht automatisch durch die Teilnahme an der Klausur gegeben.

Pro Aufgabe werden 10 Punkte vergeben. Richtwerte für das Bestehen der Klausur:

- Kombinierte DAP1/DAP2-Klausur:
DAP1-Teil \geq 20 Punkte, DAP2-Teil \geq 20 Punkte, gesamt \geq 45 Punkte.
- Klausur nur über DAP2: \geq 23 Punkte.

Bewertung, nur vom Korrektor/der Korrektorin auszufüllen:

Aufgabe	1	2	3	4	5	6	Gesamt
Punkte							

Name:

Matrikelnummer:

1. Aufgabe: DFS für gerichtete Graphen

- a) Beschreibe den DFS-Algorithmus für gerichtete Graphen in algorithmischer Notation, also „Pseudocode“. Erkläre dabei die Bedeutung der benutzten Variablen bzw. Parameter.
- b) Der Algorithmus teilt die Kanten des Graphen in verschiedene Klassen ein. Beschreibe, wie dies geschieht, indem Du für jede der Klassen die entsprechende(n) Bedingung(en) hinschreibst, die erfüllt sein müssen. Gehe bei Deiner Beschreibung davon aus, dass die Kante (v, w) klassifiziert werden soll.
- c) Nenne die Laufzeit des DFS-Algorithmus, wenn ein Graph mit m Kanten und n Knoten durch ...
 - seine Adjazenzlisten gegeben ist.
 - seine Adjazenzmatrix gegeben ist.

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 1. Aufgabe:

Name:

Matrikelnummer:

2. Aufgabe: π -OBDDs

- a) Zeichne für $\pi = (x_4, x_2, x_3, x_1)$ ein möglichst kleines π -OBDD für die Funktion $f = x_1 \vee (\overline{x_2} \wedge x_3)$.
- b) Definiere, was man unter einem Erfüllbarkeitstest versteht und beschreibe, wie man einen solchen Erfüllbarkeitstest auf π -OBDDs durchführt.
- c) Beschreibe, wie man π -OBDDs minimiert.

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 2. Aufgabe:

Name:

Matrikelnummer:

3. Aufgabe: Balancierte Suchbäume

- a) Ein Baum heißt 2-3-Baum genau dann, wenn bestimmte Eigenschaften erfüllt sind. Zähle diese Eigenschaften auf.
(Zur Erinnerung: Im Skript gab es fünf Eigenschaften).
- b) Sei B ein 2-3-Baum mit n Schlüsseln. Zeige, dass für die Tiefe d von B gilt, dass $d = O(\log n)$ ist und dass $d = \Omega(\log n)$ ist.
- c) Beschreibe, wie die INSERT-Operation bei 2-3-Bäumen abläuft. Du darfst dabei davon ausgehen, dass das Blatt, in das ein neuer Schlüssel zunächst eingefügt wird, schon gefunden wurde.

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 3. Aufgabe:

Name:

Matrikelnummer:

4. Aufgabe: Quickselect

- a) Wie ist das Auswahlproblem definiert?
- b) Beschreibe den Algorithmus `quickselect` zur Lösung des Auswahlproblems.
- c) Beschreibe den Ansatz für die average-case-Analyse für die Anzahl an wesentlichen Vergleichen.
- d) Nenne das Endergebnis der Analyse, also: wieviele wesentliche Vergleiche finden im average-case höchstens statt?

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 4. Aufgabe:

Name:

Matrikelnummer:

5. Aufgabe: Dynamische Programmierung beim Rucksackproblem

Wir haben in der Vorlesung das Rucksackproblem kennen gelernt.

- a) Definiere das Problem formal.
- b) Wie löst man das Rucksackproblem mit der dynamischen Programmierung?
Genauer: Gib die Bellmansche Optimalitätsgleichung an und begründe, warum sie gilt. Erläutere die Parameter.
Hinweis: Es sei nur der optimale Nutzen zu berechnen, nicht die optimale Be-
packung selbst.
- c) Nenne und begründe die Laufzeit dieses Verfahrens.

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 5. Aufgabe:

Name:

Matrikelnummer:

6. Aufgabe: Minimale Spannbäume

- a) Ergänze folgende Aussage:
Ein ungerichteter Graph $T = (V, E)$ heißt Baum genau dann, wenn ...
- b) Ergänze folgende Aussage:
Ein ungerichteter Graph $G' = (V', E')$ heißt Spannbaum von $G = (V, E)$ genau dann, wenn ...
- c) Beschreibe den Algorithmus von Kruskal.
- d) Beweise, dass der Algorithmus von Kruskal einen Spannbaum berechnet.
- e) Beweise, dass der Algorithmus von Kruskal einen *minimalen* Spannbaum berechnet.

Klausur 30.09.03 DAP2-Teil

Name:

Matrikelnummer:

Fortsetzung 6. Aufgabe: