

**Hinweise zur Bearbeitung der Klausur zum Kurs
01664 „Implementierungskonzepte für Datenbanksysteme“**

Bitte lesen Sie sich diese Hinweise vollständig und aufmerksam durch, bevor Sie mit der Bearbeitung der Klausur beginnen.

1. Prüfen Sie bitte die Vollständigkeit Ihrer Unterlagen. Die Klausur umfasst:
 - 2 Deckblätter
 - 1 Formblatt für eine Teilnahmebescheinigung zur Vorlage beim Finanzamt
 - diese Hinweise
 - 5 Aufgaben auf den Seiten 2 - 3
2. Die Klausurdauer beträgt 2 Stunden.
3. Für die Klausur sind **keine Hilfsmittel zugelassen**. Es darf nur unbeschriebenes Konzeptpapier und Schreibzeug verwendet werden. Die Reinschrift der Klausur darf **nicht mit Bleistift** erfolgen.
4. Schreiben Sie Ihre Lösungen auf Ihr **eigenes Papier** (DIN A4) und nicht auf die Seiten mit den Aufgabenstellungen.
5. Bevor Sie mit der **Bearbeitung der Klausuraufgaben** beginnen, füllen Sie bitte die folgenden Teile der Klausur aus:
 - **sämtliche Deckblätter mit Name, Anschrift sowie Matrikelnummer.**
 - Schreiben Sie bitte **auf jedes weitere Blatt oben links Ihren Namen und oben rechts Ihre Matrikelnummer.** Nummerieren Sie Ihre Seiten bitte durch.
 - die Teilnahmebescheinigung, falls Sie diese wünschen.
6. **Vor der Abgabe Ihrer Klausur:**
 - **Heften Sie Ihre Bearbeitung an Ihr vollständiges Klausurexemplar. Die Aufgabenblätter müssen mit abgegeben werden!**
 - **Kreuzen Sie auf beiden Deckblättern die von Ihnen bearbeiteten Aufgaben an.**
7. Durch Lösen der Aufgaben sind maximal 100 Punkte erreichbar. Sie dürfen damit rechnen einen Übungsschein bzw. ein Zertifikat zu erhalten, wenn Sie insgesamt mindestens 50 Punkte erreichen.

Aufgabe 1: Ausrichtung von Feldwerten

20 Punkte

- (a) Gegeben sei die folgende Anordnung von Feldtypen:

12 Punkte

¹ a: Char ⁴ b: Integer c: String[5] ⁸ d: Double ¹ e: Boolean
⁸ f: Double g: Boolean h: Integer i: String[2]

Bestimmen Sie die Größe eines Datensatzes, der die angegebenen Typen in der vorgegebenen Reihenfolge enthält. Wieviel Speicherplatz wird verschwendet, wenn 1.000.000 Datensätze hintereinander gespeichert werden?

- (b) Geben Sie ein Anordnung der einzelnen Felder an, die zu einer minimalen Datensatzgröße führt. Begründen Sie Ihr Ergebnis.

8 Punkte

Verwenden Sie für die Lösung dieser Aufgabe die folgenden Typgrößen:

Boolean 1 Byte
 Integer 4 Byte
 Double 8 Byte
 Char 1 Byte
 String[i] wird als Folge von i Chars angesehen und belegt daher i Byte;
 jedoch kann die Anfangsadresse auf jeder beliebigen Position liegen.

Aufgabe 2: Erweiterbares Hashing

26 Punkte

- (a) Fügen Sie durch Anwendung des erweiterbaren Hashings der Reihe nach die Suchschlüssel

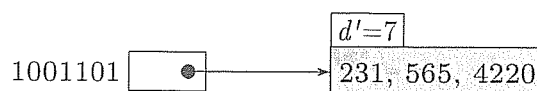
19 Punkte

10, 15, 11, 1, 29, 26, 19, 18, 6, 4

in eine Datei ein.

Zeichnen Sie die erweiterbare Hash-Struktur jeweils nach jedem Einfügevorgang, bei dem ein Behälter überläuft. Dabei lautet die verwendete Hash-Funktion $h(x) = x \bmod 8$, und jeder Behälter kann 3 Datensätze speichern.

Hinweis: Um eine aufwändige graphische Darstellung zu vermeiden, notieren wir die Binärdarstellung z eines Indexes für einen Verzeichniseintrag als (z) und die in einem Behälter mit lokaler Tiefe d' enthaltenen Datensätze anhand ihrer Schlüssel k_1, \dots, k_n mit $[k_1, \dots, k_n]_{d'}$. Die Zuordnung von (z) zu $[k_1, \dots, k_n]_{d'}$ notieren wir mit $(z) \rightarrow [k_1, \dots, k_n]_{d'}$. Einen Verzeichniseintrag der Form



stellen wir also durch die Schreibweise $(1001101) \rightarrow [231, 565, 4220]_7$ dar.

- (b) Entfernen Sie nun die Suchschlüssel 18 und 11 aus der Hash-Struktur aus (a). Auch hier genügt die Angabe der Hash-Struktur nach jedem Löschvorgang, der Änderungen an der Verzeichnisstruktur bewirkt. 7 Punkte

Aufgabe 3: KD-Baum

18 Punkte

Fügen Sie in einen anfangs leeren KD-Baum die folgenden Punkte in der gegebenen Reihenfolge ein. Zeichnen Sie den daraus resultierenden Baum sowie die entsprechende Partitionierung des Datenraums.

(50, 25), (25, 50), (75, 60), (10, 38), (80, 86), (90, 16)

Aufgabe 4: Natürliches Mergesort

15 Punkte

Sortieren Sie die unten angegebene Folge mittels natürlichen Mergesorts unter der Voraussetzung, dass zur Bildung der Läufe 3 Speicherplätze zur Verfügung stehen. Geben Sie alle Schritte zur Berechnung der Läufe sowie die einzelnen Phasen des Sortierens an.

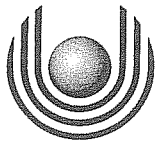
67, 54, 98, 1, 4, 96, 15, 30, 89, 17, 66, 14

Aufgabe 5: Anfrageoptimierung

18 Punkte

Beantworten Sie die folgenden Fragen zum Thema Join:

- (a) Wie unterscheidet sich der Index-Nested-Loop-Join vom Nested-Loop-Join? Wie hoch sind die Kosten für den Nested-Loop-Join? 5 Punkte
- (b) Wie unterscheidet sich der Grace-Hash-Join vom Hash-Join? 2 Punkte
- (c) Welches sind die drei Phasen im Basisalgorithmus des einfachen Hash-Joins? 3 Punkte
- (d) Was ist die Grundidee beim Sort-Merge-Join? 3 Punkte
- (e) Wie lautet die Laufzeitkomplexität des Sort-Merge-Joins (mit Begründung)? 5 Punkte



FernUniversität in Hagen

**Lösungsvorschläge
zur Hauptklausur
„1664 Implementierungskonzepte für DBS“**

15.02.2014

Einfügen von 15, 11, 1

Das Einfügen der 1 führt zum Überlauf des Behälters, so dass d auf 1 erhöht werden muss und nun zwei Behälter zur Verfügung stehen.

$$\begin{aligned}(0) &\rightarrow [1, 10, 11]_1 \\ (1) &\rightarrow [15]_1\end{aligned}$$

Einfügen von 29, 26

Das Einfügen der 26 führt zum Überlauf des Behälters mit Index 0, und wegen $d' = d$ muss d auf 2 erhöht werden.

$$\begin{aligned}(00) &\rightarrow [1]_2 \\ (01) &\rightarrow [10, 11, 26]_2 \\ (10) &\} \\ (11) &\} \rightarrow [15, 29]_1\end{aligned}$$

Einfügen von 19

Das Einfügen dieses Schlüssels führt zum Überlauf des Behälters mit Index 01, und wegen $d' = d$ muss d auf 3 erhöht werden.

$$\begin{aligned}(000) &\} \\ (001) &\} \rightarrow [1]_2 \\ (010) &\rightarrow [10, 26]_3 \\ (011) &\rightarrow [11, 19]_3 \\ (100) &\} \\ (101) &\} \\ (110) &\} \\ (111) &\} \rightarrow [15, 29]_1\end{aligned}$$

Einfügen von 18, 6, 4

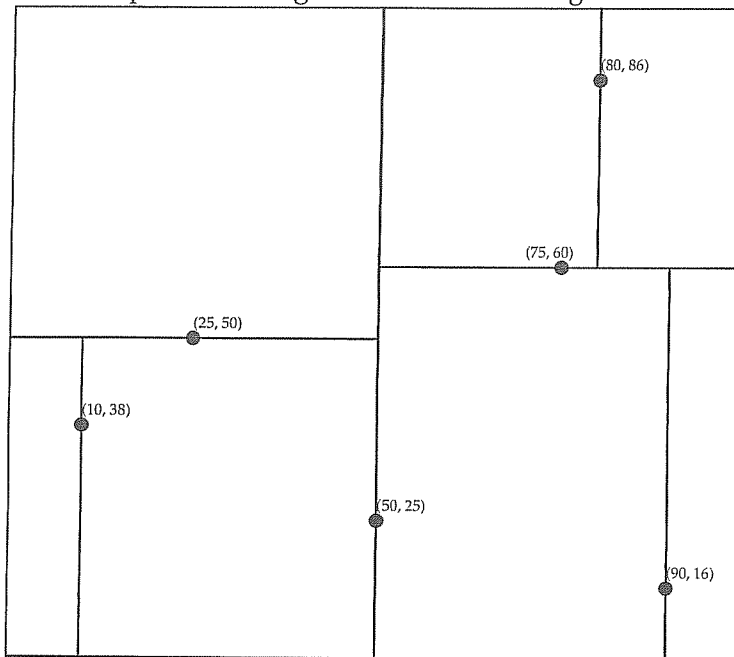
$$\begin{aligned}(000) &\} \\ (001) &\} \rightarrow [1]_2 \\ (010) &\rightarrow [10, 18, 26]_3 \\ (011) &\rightarrow [11, 19]_3 \\ (100) &\} \\ (101) &\} \rightarrow [4, 29]_2 \\ (110) &\} \\ (111) &\} \rightarrow [6, 15]_2\end{aligned}$$

(b)

Löschen von 18 und 11

Das Löschen des Schlüssels 18 verursacht keine Änderungen an der Verzeichnisstruktur. Das Löschen der 11 hingegen führt dazu, dass die Behälter mit den Indizes 010 und 011 verschmolzen werden. Die lokale Tiefe d' wird von 3 auf 2 verringert.

Dies entspricht der folgenden Partitionierung des Raums:



Aufgabe 4 Natürliches Mergesort

Die initialen Läufe berechnen sich wie folgt:

1	2	3	1. Lauf	1	2	3	2. Lauf	1	2	3	3. Lauf
<u>54</u>	67	98	54	<u>1</u>	4	96	1	<u>14</u>	66	17	14
<u>67</u>	98	<u>1</u>	67	<u>4</u>	15	96	4	<u>17</u>	66		17
<u>98</u>	<u>4</u>	<u>1</u>	98	<u>15</u>	30	96	15	<u>66</u>			66
<u>96</u>	<u>4</u>	<u>1</u>		<u>30</u>	89	96	30				
				<u>89</u>	96	<u>17</u>	89				
				<u>96</u>	<u>66</u>	<u>17</u>	96				
				<u>14</u>	<u>66</u>	<u>17</u>					

Die eigentliche Sortierung ist nun:

Initiale Läufe:

f1: 54 67 98 | 14 17 66 |

f2: 1 4 15 30 89 96 |

Phase 1

g1: 1 4 15 30 54 67 89 96 98 |

g2: 14 17 66 |

Phase 2

f1: 1 4 14 15 17 30 54 66 67 89 96 98 |

f2: