

Aufgabe 1

1 Punkt

Betrachten Sie die folgende Festplatte:

- Sektorgröße $B = 2^9\text{B}$,
- Interblock Gap Size $G = 2^2\text{B}$,
- Sektoren/Spur $S = 2^8$,
- Spuren pro Scheibenseite $T = 300$,
- Anzahl der beidseitig beschriebenen Scheiben $D = 2^3$,
- mittlerer Spurwechsel $sp = 2\text{ms}$,
- Drehzahl $dz = 60 \cdot 2^7\text{rpm}$

Berechnen Sie die **Zugriffszeit pro Sektor (in ms)** für einen **Random Access**.

Lösung: _____

Aufgabe 21 Punkt

Gegeben ist die folgende Slotted Page:

- Größe: 2^{10} Bytes
- Header: dense (dicht gepackt)
- Adressierungstyp: **Byte-Adressierung**

Es werden 2^6 Datensätze der Größe 2^3 Bytes auf der Slotted Page gespeichert.

Berechnen Sie die **Größe des gesamten Headers (in Bits)** (ohne Free Space und ohne Datensätze).

Lösung: _____

Aufgabe 3

1 Punkt

Gegeben ist eine Relation $R[A, B, C, D]$ mit folgenden Eigenschaften:

- $|R| = 1.000.000$ Tupel.
- R ist nach B sortiert.
- Werte von A, B : **nicht** eindeutig, gleichverteilt zwischen 1 und 10.000
- Tupel pro Datenblock: 25.

Auf R existieren folgende Indizes:

- $Index_{A1}$: einstufiger dense Index auf Attribut A (**kein** B^+ Baum Index).
- $Index_{A2}$: einstufiger sparse Index auf $Index_{A1}$ (**kein** B^+ -Baum Index).
- $Index_B$: sparse B^+ Baum Index auf B .
- Einträge/Schlüssel pro Indexblock/Knoten: 60.

Geben Sie die **Strategie** sowie die **Anzahl der nötigen Blockzugriffe** für die effizienteste Zugriffsmethode an, um die **Anfrage** $\sigma_{B=5.000}(R)$ zu beantworten.

Lösung: _____

Aufgabe 4

1 Punkt

Gegeben sind die Blätter eines B^+ Baumes ($m = 3$). Konstruieren Sie die **darüberliegenden Ebenen** (d.h. die inneren Knoten) so, dass ein **minimaler und gültiger B^+ Baum** entsteht.



Aufgabe 5

1 Punkt

Auf der folgenden Tabelle soll ein **Hash-Index** konstruiert werden. Als Schlüssel wird das Attribut *persNum* verwendet. Der Hashwert ist die **zweite Ziffer** des Attributwerts. Es können **2 Tupel pro Bucket** gespeichert werden. Bucket Overflows werden durch **Overflow Chaining** aufgelöst. **Illustrieren** Sie den **Hash-Index**.

| <i>Name</i> | <i>persNum</i> |
|-------------|----------------|
| Elizabeth | 2006 |
| Ross | 8628 |
| David | 2263 |
| Norman | 9097 |
| Jayne | 6529 |
| Gary | 9627 |
| Michael | 0198 |
| Daniel | 5009 |
| Keith | 6451 |
| Alan | 4192 |

Aufgabe 61 Punkt

Führen Sie **externes Merge-Sort** auf der folgenden Relation $R[A]$ aus.
Jeder **Block fasst 2 Tupel**. Die Größe des **Puffer** beträgt **3 Blöcke**.

| |
|-----|
| 100 |
| 35 |
| 10 |
| 45 |
| 95 |
| 20 |
| 25 |
| 5 |
| 30 |
| 90 |
| 40 |
| 0 |
| 15 |
| 50 |
| 80 |
| 60 |
| 55 |
| 75 |
| 65 |
| 70 |
| 85 |

Aufgabe 7**1 Punkt**

Gegeben ist die Relation $R[A, B, C, D]$. Auf Attribut C existiert ein **dense B⁺ Baum Index**. Werte von Attribut C sind gleichverteilt. Was ist die **effizienteste Strategie** um **Bereichsanfragen** von folgendem Typ zu beantworten?

$$\sigma_{a < C < b}(R)$$

Annahme:

$b - a \ll \max(R.C) - \min(R.C)$, d.h. der abgefragte Wertebereich ist wesentlich kleiner als der gesamte Wertebereich von Attribut C .

Geben Sie **alle notwendigen Schritte** an.

Aufgabe 81 Punkt

Gegeben ist eine Relation $R[A, B, C, D]$ mit den folgenden Eigenschaften:

- $|R| = 1.000.000$ Tupel.
- Werte von B liegen zwischen 1 und 4.000.
- Jeder Wert von Attribut B kommt 250 mal vor.
- Ist auf 20.000 Blöcken gespeichert.
- $Index_A$: dense B⁺ Baum Index auf Attribut A ($m = 251$).
- $Index_B$: sparse B⁺ Baum Index auf Attribut B ($m = 251$).

Duplikate werden durch Verknüpfung mit dem TID (Tuple Identifier) eindeutig gemacht.

(0.5 Punkte) Markieren Sie die **zwei effizientesten Strategien** (bzgl. der Lockzugriffe) um die folgende **Anfrage** zu beantworten:

$$\sigma_{B < 10}(R)$$

(0.5 Punkte) Begründen Sie, warum Sie die **nicht gewählten Strategien** nicht gewählt haben.

- Da R nicht nach B sortiert ist, muss die gesamte Relation gescannt werden und für jedes Tupel $B < 10$ überprüft werden.
- Suche kleinsten Schlüssel in $Index_B$. Scanne Blattknoten des B⁺ Baumes sequentiell bis $B \geq 10$. Verfolge gelesene Pointer zu den Datenblöcken.
- Suche kleinsten Schlüssel in $Index_A$. Scanne alle Blattknoten des B⁺ Baumes sequentiell und überprüfe, ob $B < 10$. Benutze Bitmap Index Scan um mehrfaches Lesen von Datenblöcken zu verhindern.
- Beginne Scan der Datenblöcke beim ersten Datenblock von R . Scanne bis erstes Tupel mit $B \geq 10$ gefunden wird.

Aufgabe 9

1 Punkt

Gegeben sind **zwei Relationen** mit folgenden Eigenschaften:

| $R[A, B, C, D]$ | $S[D, E, F, G]$ |
|---|--|
| $ R = 20.000$ Tupel, $b_R = 1.000$ Blöcke | $ S = 50.000$ Tupel, $b_S = 2.000$ Blöcke |
| flacher dense Index auf A | sparse B ⁺ Baum Index auf D ($m = 100$) |
| dense B ⁺ Baum Index auf D ($m = 100$) | |

Die Werte in R und S sind gleichverteilt und unabhängig.

Berechnen Sie die **Kosten (Blockzugriffe)** eines **Natural Joins** zwischen R und S mit Hilfe des **Index-Nested-Loop-Join** Algorithmus und geben Sie die **effizientere Join-Reihenfolge** ($R \bowtie S$ oder $S \bowtie R$) an.

Aufgabe 10

1 Punkt

Gegeben sind **zwei Relationen** mit folgenden Eigenschaften:

| $R[A, B, C, D]$ | $S[D, E, F, G]$ |
|---|--|
| $ R = 20.000$ Tupel, $b_R = 1.000$ Blöcke | $ S = 50.000$ Tupel, $b_S = 2.000$ Blöcke |
| flacher dense Index auf A | sparse B ⁺ Baum Index auf D ($m = 100$) |
| dense B ⁺ Baum Index auf D ($m = 100$) | |
| $V(R, A) = 1.000$, $V(R, B) = 250$ | $V(S, D) = 5.000$, $V(S, E) = 500$ |
| $V(R, C) = 400$, $V(R, D) = 100$ | $V(S, F) = 1.000$, $V(S, G) = 200$ |

Die Werte in R und S sind gleichverteilt und unabhängig. $V(R, A)$ gibt die Anzahl der verschiedenen Werte von Attribut A in Relation R an.

Berechnen Sie die **Kosten (Blockzugriffe)** der folgenden Anfrage, wenn der **Sort-Merge-Join** Algorithmus für den Join benutzt wird:

$$\sigma_{B=b}(R) \bowtie S$$

Es steht ein **Puffer der Größe $M = 3$ Blöcke** zur Verfügung. Die **Selektion** soll **vor dem Join** durchgeführt werden und in einer temporären Relation gespeichert werden.

Lösung: _____

Aufgabe 11**1 Punkt**

Gegeben sind **zwei Relationen** $R[A, B, C, D]$ und $S[D, E, F, G]$ und die folgende **Anfrage**:

$$\pi_{A,E,G}(\sigma_{C=c}(R) \bowtie \sigma_{D=d \wedge G=g}(S))$$

(0.5 Punkte) Zeichnen Sie zuerst die **algebraische Normalform** (Projektionen, Selektionen, Kreuzprodukte).

(0.5 Punkte) Bringen Sie die Anfrage anschließend in die **optimierte Form**.

Aufgabe 12

1 Punkt

Gegeben sind **drei Relationen** mit folgenden Eigenschaften:

| $R[A, B, C, D]$ | $S[D, E, F, G]$ | $T[G, H, I, J]$ |
|----------------------|----------------------|-----------------------|
| $ R = 20.000$ Tupel | $ S = 50.000$ Tupel | $ T = 100.000$ Tupel |
| $V(R, A) = 1.000,$ | $V(S, D) = 5.000,$ | $V(T, G) = 100.000,$ |
| $V(R, B) = 250,$ | $V(S, E) = 500,$ | $V(T, H) = 750,$ |
| $V(R, C) = 400,$ | $V(S, F) = 1.000,$ | $V(T, I) = 500,$ |
| $V(R, D) = 20.000$ | $V(S, G) = 200$ | $V(T, J) = 2.400$ |

$V(R, A)$ gibt die Anzahl der verschiedenen Werte von Attribut A in Relation R an. Schätzen Sie die **Kardinalität des Ergebnisses** der folgenden **Anfrage** ab ($\sigma_{A=a}(R) \neq \emptyset, \sigma_{I=i}(T) \neq \emptyset$).

$$(\sigma_{A=a}(R)) \bowtie S \bowtie (\sigma_{I=i}(T))$$

Lösung: _____