
Aufgabe 1

1 Punkt

Betrachten Sie die folgende Festplatte:

- Sektorgröße $B = 512\text{B}$,
- Interblock Gap Size $G = 64\text{B}$,
- Sektoren/Spur $S = 200$,
- Spuren pro Scheibenseite $T = 300$,
- Anzahl der beidseitig beschriebenen Scheiben $D = 10$,
- mittlerer Spurwechsel $sp = 2\text{ms}$,
- Drehzahl $dz = 7200\text{rpm}$.

Wie lange dauert es, 100 zufällig ausgewählte Sektoren zu lesen? (nicht-sequentieller Zugriff)

Lösung: _____

Name:

Matrikelnummer:

Aufgabe 2

1 Punkt

Wieviele Datensätze der Größe 28 Bytes können auf einer Slotted Page der Größe 8192 Bytes mit Byte-Adressierung gespeichert werden? Die Bits werden im Kopfteil dicht gepackt.

Lösung: _____

Aufgabe 3**1 Punkt**

Gegeben ist eine Relation $R[A, B, C, D]$. R ist in einer sequentiellen Datei nach Attribut A sortiert gespeichert und enthält 10000 Tupel. Die Werte von Attribut A sind eindeutig und liegen zwischen 1 und 10000. Jeder Datenblock kann bis zu 10 Tupel enthalten. Es gibt einen einstufigen dense Index $Index_1$ (kein B^+ -Baum Index) auf Attribut A . Weiters gibt es einen einstufigen sparse Index $Index_2$ (kein B^+ -Baum Index) auf $Index_1$. Jeder Indexblock kann bis zu 50 Einträge (Key + Pointer) beinhalten.

Geben Sie die minimale Anzahl der nötige Blockzugriffe an, um einen Datensatz mit $A = 6789$ zu lesen.

Lösung: _____

Name:

Matrikelnummer:

Aufgabe 4

1 Punkt

Konstruieren Sie einen B^+ -Baum mit einer minimalen Anzahl von Knoten und $m = 4$ Pointer pro Knoten. Der B^+ -Baum soll die folgenden Werte enthalten.

1, 5, 25, 18, 7, 24, 28, 2, 13, 14, 3, 29, 30, 20, 16, 19, 10, 15, 21, 12

Lösung: _____

Aufgabe 5**1 Punkt**

Gegeben ist ein B^+ -Baum in dem jeder Knoten maximal $m = 4$ Pointer enthalten kann. Zu Beginn besitzt der B^+ -Baum einen Wurzelknoten (root) mit zwei Kindern. Diese beiden Kindknoten sind gleichzeitig Blattknoten und enthalten die Schlüssel $\{2, 4, 6\}$ b.z.w. $\{8, 10, 12\}$. Der Wurzelknoten enthält den Wert 8.

Wieviele Einfüge-Operationen sind mindestens erforderlich (minimale Anzahl) damit der Baum um eine Ebene wächst? Geben Sie entsprechende Schlüssel an.

Lösung: _____

Name:

Matrikelnummer:

Aufgabe 6

1 Punkt

Betrachten Sie eine erweiterbare Hash-Struktur, in der Buckets bis zu zwei Schlüssel beinhalten können. Weiters sind keine Overflow Blöcke erlaubt. Diese Hash-Struktur ist anfangs leer.

Einige Schlüssel werden eingefügt. Alle eingefügten Schlüssel haben unterschiedliche Hash-Werte. Es werden keine Schlüssel gelöscht.

Geben Sie die minimale Anzahl von eingefügten Schlüsseln an, sodass nach dem Einfügen des letzten Schlüssels drei Buckets alloziert (verwendet) sind.

Lösung: _____

Aufgabe 7

1 Punkt

Betrachten Sie eine Relation $R[A]$ mit 2000 Tupel. Ein Block hat Platz für zwei Tupel. Der Buffer kann 10 Blöcke speichern.

Wieviele Blöcke müssen gelesen/geschrieben werden um einen externen Merge-Sort auf Relation R auszuführen? Zählen Sie dabei den letzten Schreib-Schritt, der das Ergebnis wieder auf die Festplatte schreibt, **nicht** mit.

Lösung: _____

Name:

Matrikelnummer:

Aufgabe 8

1 Punkt

Gegeben ist eine Relation $R[A, B, C, D]$. Diese Relation ist in einer sequentiellen Datei gespeichert. R beinhaltet 10000 Tupel, wobei die Werte von Attribut A zwischen 1 und 10000 liegen und sind eindeutig. Jeder Datenblock kann bis zu 40 Tupel speichern. Es gibt einen sparse B^+ -Baum Index auf Attribut A . Jeder Knoten im B^+ -Baum kann maximal 10 Schlüssel beinhalten.

Geben Sie die minimale Anzahl der nötige Blockzugriffe an, um alle Datensätze mit $7980 < A < 8500$ zu lesen.

Lösung: _____

Aufgabe 9**1 Punkt**

Betrachten Sie zwei Relationen $R[A, B]$ und $S[B, C]$, wobei $|R| = 2000$ Tupel und $|S| = 1000$ Tupel. Die Relationen sind auf 400 bzw. 1000 hintereinander liegenden Blöcken gespeichert. Der Buffer kann 25 Blöcke halten.

Geben Sie die minimalen Kosten für einen Natural Join zwischen R und S unter Verwendung des Block Nested Loop Join Algorithmus (den schnellsten den Sie kennen) an.

Lösung: _____

Name:

Matrikelnummer:

Aufgabe 10

1 Punkt

Welcher Join Algorithmus (Sort-Merge Join, Index Nested Loop Join) generiert die minimalen Kosten für das folgende Szenario? Geben Sie in der Lösung den Algorithmus **und** die dazugehörigen Kosten an.

Berechnen Sie einen Natural Join zwischen zwei Relationen $R[A, B]$ und $S[B, C]$, wobei $|R| = 4000$ Tupel und $|S| = 1000$ Tupel. Die Relationen sind auf 500 bzw. 1000 hintereinander liegenden Blöcken gespeichert. Der Buffer hat Platz für 21 Blöcke. Es existiert ein sparse B⁺-Baum Index auf $R.B$, wobei jeder Knoten im B⁺-Baum 20 Schlüssel speichern kann.

Lösung: _____

Aufgabe 11**1 Punkt**

Gegeben sind zwei Relationen $R[A, \underline{B}, \underline{C}]$ und $S[\underline{B}, \underline{C}, D]$ (d.h., das Paar (B,C) ist ein Schlüssel in R und S).

Formulieren Sie die folgende Anfrage so um, dass Selektionen und Projektionen so weit wie möglich nach unten geschoben werden (sodass der kleinstmögliche Join durchgeführt wird).

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

Aufgabe 12**1 Punkt**

Gegeben sind die folgende Relationen:

- $|R[A, B, C]| = 100000$, $V(R, A) = 30$, $V(R, B) = 50$, $V(R, C) = 150$
- $|S[B, C, D]| = 50000$, $V(S, B) = 100$, $V(S, C) = 40$, $V(S, D) = 50$

Die Werte in den Tupeln sind gleichverteilt und unabhängig.

Schätzen Sie die Kardinalität des Ergebnisses der folgenden Anfrage ab ($\sigma_{C=10 \wedge D=20}(S) \neq \emptyset$).

$$R \bowtie \sigma_{C=10 \wedge D=20}(S)$$

Lösung: _____