

Aufgabenblatt 2 Indexstrukturen

- Abgabetermin: **Montag, 25.11.2018 (23:59 Uhr)**
- Zur Prüfungszulassung muss ein Aufgabenblatt mit mind. 25% der Punkte bewertet werden und alle weiteren Aufgabenblätter mit mindestens 50% der Punkte.
- Die Aufgaben sollen in Zweiergruppen bearbeitet werden.
- Abgabesystem unter
<http://www.dcl.hpi.uni-potsdam.de/submit>
 - ausschließlich pdf-Dateien
 - eine Datei pro Aufgabe namens Aufgabe-<aufgabenNr>.pdf
 - jedes Blatt beschriftet mit Namen

Aufgabe 1: Indexstrukturen auf sequentiellen Dateien

Sei $R(\underline{A}, B, C)$ eine Relation. R umfasst 12 Tupel, die nach dem Attribut A sortiert in einer Datei gespeichert sind (sequentielle Datei). Innerhalb eines Blocks können zwei Tupel von R gespeichert werden.

Die nachfolgende Abbildung zeigt die Blockverteilung der Tupel von R in der sequentiellen Datei, wobei für jedes Tupel nur der Wert des Attributs A angegeben wurde.

(22, *, *)
(24, *, *)

(26, *, *)
(28, *, *)

(30, *, *)
(32, *, *)

(34, *, *)
(36, *, *)

(38, *, *)
(40, *, *)

(42, *, *)
(44, *, *)

- a) Indiziere die Datei mit einem zweistufigen Primärindex \mathcal{I} , wobei in jeder Stufe ein dünnbesetzter Index zu verwenden ist. Innerhalb eines Blocks können vier Schlüssel-Zeiger-Paare gespeichert werden. **3 P**
- b) Notiere zuerst in der Abbildung aus Teilaufgabe a) neben jedem Block der Indexstruktur und der sequentiellen Datei einen eindeutigen Bezeichner (z. B. eine Zahl). Betrachte nun die nachfolgenden SQL-Anfragen und gebe jeweils die eindeutigen Bezeichner der einzulesenden Blöcke an, die für die Anfrageausführung *unter Verwendung des zweistufigen Pri-*

märindex \mathcal{I} eingelesen werden müssen.

Hinweis: Beachte, dass A Primärschlüssel der Relation R ist. Zudem befinden sich keine Blöcke von R und keine Blöcke von \mathcal{I} im Hauptspeicher. 7 P

- 1) `SELECT * FROM R WHERE A = 34`
- 2) `SELECT * FROM R WHERE A > 36`
- 3) `SELECT COUNT(*) FROM R WHERE A = 38`
- 4) `SELECT COUNT(*) FROM R WHERE A = 40`
- 5) `SELECT MIN(A) FROM R`
- 6) `SELECT MAX(A) FROM R`
- 7) `SELECT AVG(A) FROM R`

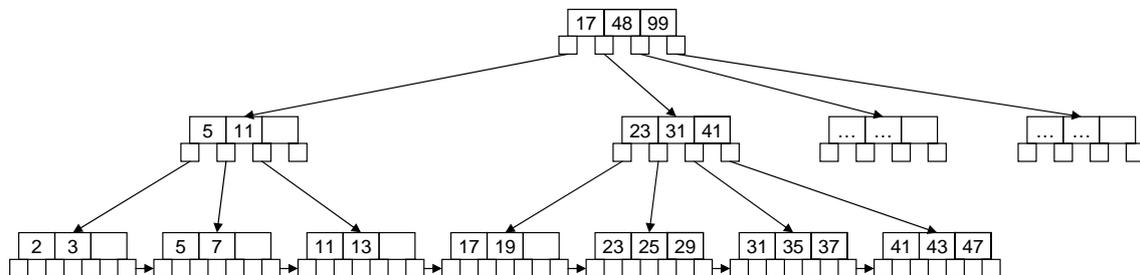
Aufgabe 2: Dichtbesetzte und dünnbesetzte Indexe

Betrachte Blöcke, die entweder 30 Datensätze oder 200 Schlüssel-Pointer-Paare aufnehmen können. Die Blöcke dürfen nur zu 80% gefüllt werden. Wie viele Blöcke werden in Abhängigkeit der Anzahl der Datensätze n benötigt für

- a) die Datensätze und einen dichtbesetzten Index? 2 P
- b) die Datensätze und einen dünnbesetzten Index? 2 P

Aufgabe 3: Einfügen und Löschen in B^+ -Bäumen

Gegeben sei der folgende B^+ -Baum:



- a) Füge die Schlüssel 1, 4, 18 und 20 (in dieser Reihenfolge) ein. Gib den B^+ -Baum nach dem Einfügen der 4 und der 20 an. Beschreibe kurz für jeden Schlüssel die notwendigen Schritte zum Einfügen. 10 P
- b) Lösche in dem Ergebnis aus Aufgabe a) nun die Schlüssel 1, 2, 3, 4, 5 und 7 (in dieser Reihenfolge). Gib den B^+ -Baum jeweils nach dem Löschen der 4 und 7 an. Beschreibe kurz für jeden Schlüssel die notwendigen Schritte zum Löschen. 10 P

Aufgabe 4: Korrekte B^+ -Bäume

Ist der folgende gegebene B^+ -Baum ein gültiger B^+ -Baum mit max. 4 Schlüssel (und damit max. 5 Pointern) pro Knoten? Falls nein, nenne alle Verletzungen der B^+ -Baum-Definition und begründe sie kurz. Die fehlenden Pointer der Blattknoten auf die Daten können ignoriert werden.

10 P

