

## Aufgabenblatt 2

— Indexstrukturen —

Ausgabe am 13.05.2008  
Abgabe bis 26.05.2008, 13.00 Uhr

### Aufgabe 1: Indexe auf sequentiellen Dateien (10 P)

Sei  $R$  eine Relation, die  $n$  Tupel enthält. Im Folgenden soll der Platzbedarf für die Speicherung von  $R$  und eines Indexes untersucht werden, dessen Suchschlüssel dem Primärschlüssel von  $R$  entspricht.

Angenommen ein Block kann entweder  $r$  Records von  $R$  oder  $k$  Schlüssel-Zeiger-Paare speichern. Dann sei  $D(n)$  die Anzahl der Blöcke, die für die Speicherung von  $R$  und eines *dichtbesetzten* Indexes benötigt wird. Analog sei  $S(n)$  die Anzahl der Blöcke, wenn statt eines dichtbesetzten ein *dünnbesetzter* Index verwendet wird. Berechnen Sie  $D(n)$  und  $S(n)$ . (4 P)

Nehmen Sie im Folgenden an, dass  $r = 3$  und  $k = 10$ . Sei  $\mathcal{I}_d$  eine mehrstufige Indexstruktur, so dass der Index der ersten Stufe dichtbesetzt ist und der Index der höchsten Stufe nur noch ein Schlüssel-Zeiger-Paar enthält. Analog sei  $\mathcal{I}_s$  eine Indexstruktur, in der der Index der ersten Stufe dünnbesetzt ist. Wieviele Blöcke werden für die Speicherung von  $\mathcal{I}_d$  bzw.  $\mathcal{I}_s$  benötigt (ohne Blöcke für die Speicherung von  $R$  selbst)? (6 P)

### Aufgabe 2: Sekundärindexe (9 P)

Betrachten Sie erneut die Relation `Filme` aus der Vorlesung. Angenommen, es existiert ein Sekundärindex auf dem Attribut `StudioName` und ein Sekundärindex auf dem Attribut `Jahr`. Die Relation enthält 105 Filme des Studios Babelsberg. Ferner sind 215 Filme aus dem Jahr 2007 gespeichert, wobei nur einer von diesen in Babelsberg aufgenommen wurde.

Angenommen ein Block kann entweder drei Records, 10 Schlüssel-Zeiger-Paare oder 50 Zeiger enthalten. Berechnen Sie die Anzahl an I/O-Operationen, um die Anfrage

```
SELECT *  
FROM Filme  
WHERE StudioName = 'Babelsberg' AND Jahr = 2007
```

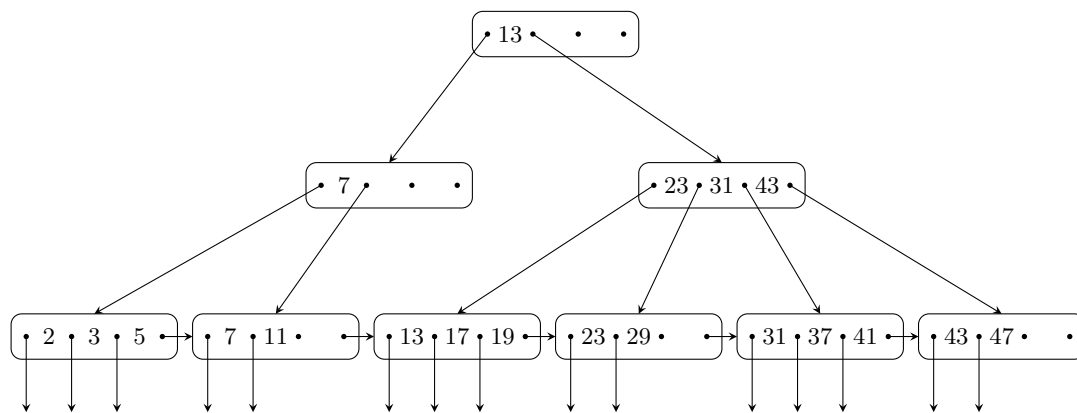
zu beantworten, wenn jeweils eine der folgenden Nebenbedingungen gilt. Nehmen Sie für die Berechnung den Fall an, dass beide Sekundärindexe bereits vollständig im Hauptspeicher vorhanden sind (nicht aber die Buckets und Records).

- (a) Für beide Sekundärindexe werden Buckets verwendet. (3 P)
- (b) Es wird nur der Sekundärindex auf dem Attribut **Studioname** verwendet (ohne Buckets). Nehmen Sie für die Berechnung an, dass es keine Babelsberg-Records gibt, die auf dem gleichen Block liegen. (3 P)
- (c) Es wird nur der Sekundärindex auf dem Attribut **Jahr** verwendet (ohne Buckets). Nehmen Sie für die Berechnung an, dass es keine 2007-Records gibt, die auf dem gleichen Block liegen. (3 P)

**Aufgabe 3: Operationen in B<sup>+</sup>-Bäumen**

(15 P)

Nehmen Sie an, dass ein Block 3 Schlüsselwerte und 4 Zeiger speichern kann. Betrachten Sie folgenden B<sup>+</sup>-Baum.



Führen Sie nacheinander folgende Operationen durch. Geben Sie den resultierenden Baum nach *jeder* Operation an! Teilbäume, die nach der Ausführung einer Operation unverändert sind, können Sie mit ... kennzeichnen.

- (a) Einfügen eines Records mit Schlüssel 6.
- (b) Einfügen eines Records mit Schlüssel 8.
- (c) Einfügen eines Records mit Schlüssel 12.
- (d) Einfügen eines Records mit Schlüssel 9.
- (e) Einfügen eines Records mit Schlüssel 10.
- (f) Löschen des Records mit Schlüssel 43.
- (g) Löschen des Records mit Schlüssel 41.
- (h) Löschen des Records mit Schlüssel 13.
- (i) Löschen des Records mit Schlüssel 5.

**Aufgabe 4: Aufzählung von  $B^+$ -Bäumen** (9 P)

Angenommen ein Knoten kann drei Schlüssel und vier Zeiger speichern (analog zu Aufgabe 3). Bestimmen Sie die Anzahl unterschiedlicher  $B^+$ -Bäume, wenn die zu indizierende Relation aus

- (a) 6 (2 P)
- (b) 10 (3 P)
- (c) 15 (4 P)

Records besteht.

**Aufgabe 5: Bulk-Loading** (10 P)

Gegeben sei eine Relation, die 20.000 Records umfasst. Die Relation wurde mittels eines  $B^+$ -Baums (als dichtbesetzter Index) indiziert. Zur Erstellung des Index wurde die in der Vorlesung vorgestellte *Bulk Loading*-Methode verwendet, wobei die Knoten jeder Ebene jeweils so voll wie möglich gefüllt wurden.

Nehmen Sie an, dass für die Speicherung eines Suchschlüssels 40 Byte und für die Speicherung eines Zeigers 10 Byte benötigt werden. Ferner sei angenommen, dass der gesamte Block für die Speicherung von Schlüssel-Zeiger-Paaren zur Verfügung steht (keine Headerinformationen). Die Größe eines Blocks beträgt 1000 Byte.

Beantworten Sie ausgehend von den angegebenen Informationen folgende Fragen:

- (a) Wie groß ist der Parameter  $n$ ? (2 P)
- (b) Wieviele Stufen hat der erzeugte  $B^+$ -Baum? (2 P)
- (c) Geben Sie für jede Stufe die Anzahl der Knoten auf dieser an! (2 P)
- (d) Wieviele Stufen hätte der  $B^+$ -Baum, wenn durch eine Komprimierung des Suchschlüssels für diesen durchschnittlich nur noch 10 Byte benötigt würden? (2 P)
- (e) Wieviele Stufen hätte der  $B^+$ -Baum, wenn keine Komprimierung der Suchschlüssel vorgenommen, dafür aber die Blöcke nur 50% befüllt würden? (2 P)