

Indexstrukturen

Tobias Bittner

2025-01-14

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

- Zugriff über den Inhalt (Suchschlüssel), nicht über die Satzadresse
- Berechnen der Satzadresse (bzw. Speicherposition) aus Schlüsselwert braucht neue Hilfsstruktur
- Index als Speicherstruktur mit Hashing für Suchschlüssel

Übersicht

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung**
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

Hausaufgabe 1: Fragen zur Vorlesung I

- a) Was versteht man unter einem (Such-)Schlüssel?
- b) Warum wird der Schlüsselzugriff zusätzlich zum Zugriff über Satzadressen in einem DBS benötigt?
- c) Hashing kann man in zwei Bereiche unterteilen:
 - i) Auswahl der Hashfunktion
 - ii) Überlaufbehandlung

Worauf muss man bei diesen Schritten für bestmögliche Leistung jeweils achten?

- d) Was sind die charakteristischen Eigenschaften eines B-Baums?
- e) Was ist der Vorteil eines B-Baums gegenüber binären Suchbäumen?
- f) Was unterscheidet den B*-Baum vom B-Baum? Was sind die Vor- und Nachteile?
- g) Welche Möglichkeiten fallen Ihnen ein, um Einträge wieder aus einem Index zu entfernen?
- h) Wann kann der Einsatz eines Bitmap-Index sinnvoll bzw. überhaupt nutzbar sein?

Übersicht

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung**
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

Hausaufgabe 2: Überlaufbehandlung I

- a) Was sind die Probleme des Hashings ohne Reorganisation in Bezug auf den Speicher?
- b) Welche Verfahren zur Behandlung von Überläufern kennen Sie? Was sind ihre Vor- und Nachteile?
- c) Wie lassen sich die Probleme zu vieler Überläufer trotz unbekannter Zahl zu erwartender Einträge vermeiden? Wie unterscheiden sich die verschiedenen Verfahren?

Übersicht

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1**
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

Hausaufgabe 3: Lineares Hashing 1 I

Gegeben ist die unten skizzierte lineare Hashtabelle mit $q = 5$ Buckets, die jeweils eine Größe $b = 2$ besitzen. Es wurden bisher die 4 vorgegebenen Schlüssel in die Tabelle eingefügt und noch keine Verdopplungen ausgeführt. Positionszeiger $p = 0$. Die Folge von Hashfunktionen sei $h_j(k) = k \bmod (2^j \cdot q)$, $j = 0, 1, \dots$

Fügen Sie die folgenden Schlüssel in der angegebenen Reihenfolge in die Tabelle ein. Verwenden Sie dabei kontrolliertes Splitting mit Schwellenwert $\alpha = 0,8$.

18, 50, 66, 2, 35, 51, 17, 22, 38, 85, 32, 16, 59, 9

Hausaufgabe 3: Lineares Hashing 1 II

p

0 1 2 3 4

15	11		23	29

Belegungsfaktor:

$$\beta = \frac{N}{(q \cdot 2^L + p) \cdot b} = \frac{4}{(5 \cdot 2^0 + 0) \cdot 2} = 0,4$$

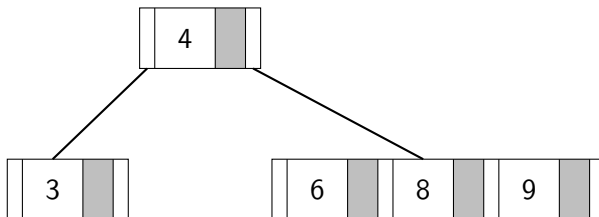
mit L Anzahl der schon vollständig durchgeführten Verdoppelungen

Übersicht

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen**
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

Hausaufgabe 4: Eigenschaften von B-Bäumen I

Begründen Sie, warum der folgende Baum kein B-Baum ist.



Übersicht

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum**
- 7 Hausaufgabe 6: B*-Baum
- 8 Klausur

Hausaufgabe 5: Einfügen und Löschen im B-Baum I

- a) Fügen Sie in einen anfangs leeren B-Baum mit $k = 2$ die Zahlen eins bis zwanzig in aufsteigender Reihenfolge ein.
- b) Löschen Sie nun den Eintrag 9 und zeichnen Sie den entstehenden Baum.

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum**

Hausaufgabe 6: B*-Baum I

Gegeben ist ein anfangs leerer B*-Baum. Die maximale Zahl der Einträge für einen inneren Knoten beträgt $2k$, $k = 2$. Die maximale Zahl der Einträge in einem Blattknoten beträgt $2k^*$, $k^* = 1$. Es sollen Tupel bestehend aus dem Schlüssel (einer Ganzzahl) und einem String der Länge 3 gespeichert werden. Führen Sie die im Folgenden angegebenen Einfüge- und Löschoperationen im B*-Baum durch. Wenn sich dessen Struktur dabei ändert, zeichnen Sie den B*-Baum neu!

- a) Fügen Sie die folgenden Tupel in der angegebenen Reihenfolge in den B*-Baum ein: $(1, fan)$, $(13, gct)$, $(7, xxe)$, $(8, lwc)$, $(22, vkw)$, $(5, wym)$, $(2, gzw)$, $(10, ycc)$
- b) Löschen Sie jetzt den Datensatz mit Schlüssel 10, danach den mit Schlüssel 13.

- 1 Wiederholung
- 2 Hausaufgabe 1: Fragen zur Vorlesung
- 3 Hausaufgabe 2: Überlaufbehandlung
- 4 Hausaufgabe 3: Lineares Hashing 1
- 5 Hausaufgabe 4: Eigenschaften von B-Bäumen
- 6 Hausaufgabe 5: Einfügen und Löschen im B-Baum
- 7 Hausaufgabe 6: B*-Baum