

Anfragebearbeitung und Indexstrukturen in Datenbanksystemen
WS 2013/14

Übungsblatt 4: gewichtete B-Bäume

Besprechung: 11.11.2013

Aufgabe 4-1 *gewichtete B-Bäume*

Gegeben sei der gewichtete 2B-Baum der Ordnung 1 aus der Vorlesung (Seite 35 im Skript) mit den Schlüsseln und Gewichten:

(a,5), (b,10), (c,8), (d,4) und (e,7).

- (a) Man füge in diesen Baum die Schlüssel (f,1) und (g,15) ein.
- (b) Anschließend erhöhe man das Gewicht des Schlüssels f um 15 (auf 16).

Aufgabe 4-2 *gewichtete B-Bäume*

- (a) Gegeben sei ein gewichteter 2B-Baum T und ein Schlüssel x_i in T . Man skizziere einen Algorithmus für die Operation **SPLIT**, der in $O(\log(\frac{\omega}{\omega_i}))$ Zeit den Baum T in x_i und zwei gewichtete Teilbäume T_1 und T_2 aufteilt, so daß alle Schlüssel in T_1 kleiner sind als x_i und alle Schlüssel in T_2 größer sind als x_i sind.
- (b) Gegeben sind zwei gewichtete 2B-Bäume T_1 und T_2 und ein Schlüssel x_i , wobei alle Schlüssel in T_1 kleiner als x_i sind und alle Schlüssel in T_2 größer als x_i sind. Man skizziere einen Algorithmus für die Operation **CONCATENATE**, die den Schlüssel x_i und die Bäume T_1 und T_2 zu einem gewichteten 2B-Baum zusammenfügt.

Für diese Aufgabe kann man die übrigen im Skript genannten Operationen benutzen. Deshalb ist jeweils ein sehr kurzer Algorithmus möglich (3 Zeilen).

Aufgabe 4-3 *gewichtete B-Bäume*

Wie im Skript angemerkt wurde, geht der gewichtete 2B-Baum nur auf die Wahrscheinlichkeit ein, daß man auf einen vorhandenen Schlüssel zugreift. Die Wahrscheinlichkeit, daß ein Schlüssel nicht gefunden wird, berücksichtigt der gewichtete 2B-Baum dagegen nicht.

- (a) Man erweitere das Konzept der gewichteten 2B-Bäume so, daß auch das Nichtfinden von Schlüsseln berücksichtigt wird.
- (b) Welche Vor- und Nachteile stellen sich durch diese Erweiterung ein ?