

Ergänzungen zu Kapitel 9 Folie 27 – Löschen im B-Baum

Allgemeiner Löschalgorithmus

• Löschen von x:

- Suche x
- Wenn x in Baum
 - Wenn x in Blatt
 - Entferne x
 - Wenn Anzahl Einträge im „Löschknoten“ $\geq k$ (nach Löschen)
 - FERTIG return gelöschterEntry (Entry mit key = x)
 - Ansonsten (**Underflow!**) :
 - Wenn Anzahl Einträge im „Löschknoten“ = k-1 und Anzahl Einträge im Nachbarn $> k$
 - **Rotation** und return gelöschterEntry
 - Wenn Anzahl Einträge im „Löschknoten“ = k-1 und Anzahl Einträge im Nachbarn = k
 - **Mischen** und return gelöschterEntry
 - Wenn x in innerem Knoten
 - Entferne x, d.h. ersetze x durch nächstkleineren bzw. nächstgrößerer Schlüssel (sind im linken Teilbaum ganz rechts unten bzw. im rechten ganz links unten)
 - Entferne den Ersatz Schlüssel aus entsprechendem Knoten (rekursiver Aufruf von delete!)
 - Dazu bietet es sich an eine delete Untermethode zu haben, der man direkt den Knoten aus dem der Schlüssel gelöscht werden soll übergeben kann (statt ihn erst zu suchen), weil der Schlüssel ja jetzt 2x im Baum ist und sonst der falsche gelöscht werden würde!
 - ansonsten return null

Wichtig: Mischen kann einen weiteren Underflow eine Ebene oben drüber hervorrufen, der sowohl rotation als auch mischen erfordern kann!

Und zum Mischen muss man noch sagen:

- Man kann mischen sowohl mit dem rechten als auch dem linken Nachbarn machen, das einzige was man beachten muss ist, dass der entsprechende Nachbar existiert, ansonsten hat man freie Wahl!
- Es kann sein, dass Mischen einen Unterlauf im Vaterknoten (wo ein Element rausgenommen wird!) erzeugt, dann muss mit diesem wieder auf Rotieren oder Mischen getestet werden. Hierbei kann es natürlich besonders eklig (je nach Datenstruktur) sein, sicher zu stellen die Zeiger auf die Nachfolgerknoten nicht zu verlieren!

Und zum Rotieren:

- Man kann auch hier mit rechtem oder linken Nachbar rotieren
Es bietet sich an mit dem zu rotieren, der mehr Elemente hat!
- Es sollen so viele Elemente rotiert werden, dass beide Knoten nachher genauso viele Elemente haben! Bei ungerader Anzahl soll der linke einen mehr haben!
- Beim rotieren in inneren Knoten muss man auch drauf achten keine Zeiger auf Nachfolgerknoten zu verlieren! (Wir übergeben unsererem rotieren/mischen dazu immer einen Zeiger auf das element des Mischvorgangs eine Rekursionsebene drunter, denn dieses element könnte auf einmal verloren gehen, bei einem k=1 baum, wenn im Vater knoten ein Unterlauf mit 0 Elementen entsteht!!!)

Ich hoffe das hilft einigermaßen
Gruß Jonas