



WIKIPEDIA
Die freie Enzyklopädie

Navigation

[Hauptseite](#)
[Themenportale](#)
[Von A bis Z](#)
[Zufälliger Artikel](#)

Mitmachen

[Artikel verbessern](#)
[Neuen Artikel anlegen](#)
[Autorenportal](#)
[Hilfe](#)
[Letzte Änderungen](#)
[Kontakt](#)
[Spenden](#)

Drucken/exportieren

[Buch erstellen](#)
[Als PDF herunterladen](#)
[Druckversion](#)

Werkzeuge

[Links auf diese Seite](#)
[Änderungen an verlinkten Seiten](#)
[Spezialseiten](#)

[Benutzerkonto anlegen](#)  [Anmelden](#)

Artikel [Diskussion](#)

Lesen

[Bearbeiten](#)

[Versionsgeschichte](#)

Suchen



2-3-4-Baum

Ein **2-3-4-Baum** ist in der [Informatik](#) eine [Datenstruktur](#), genauer ein [B-Baum](#) des Verzweigungsgrades 2, das heißt, er ist ein [Baum](#), in dem jeder [Knoten](#) zwei, drei oder maximal vier *Kinder* besitzt und entsprechend ein, zwei oder maximal drei Datenelemente speichert, die nach dem gewählten [Ordnungskriterium](#) aufsteigend sortiert sind. Er stellt damit zugleich einen speziellen [balancierten Suchbaum](#) dar.

Wie alle B-Bäume wird auch der 2-3-4-Baum häufig zur Speicherung großer Datenmengen verwendet. Das Suchen in diesen Bäumen ist mit einer [Laufzeit](#) von $O(\log n)$ möglich. Durch geschicktes Einfügen wird der 2-3-4-Baum stets [balanciert](#) gehalten.

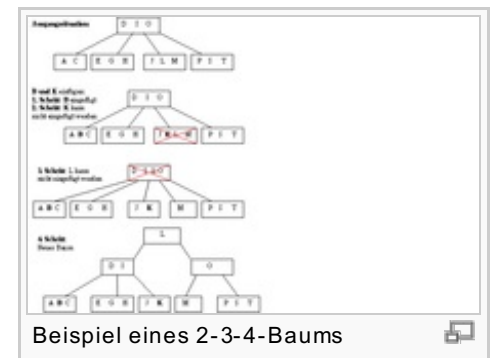
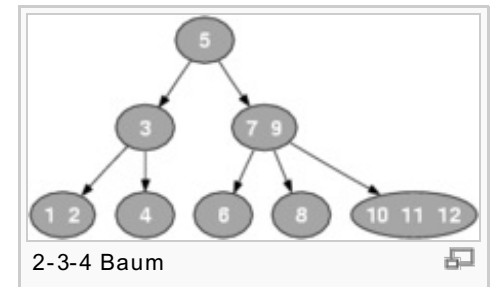
Inhaltsverzeichnis

- [1 Suchen](#)
- [2 Einfügen](#)
- [3 Löschen](#)
- [4 Varianten](#)
- [5 Literatur](#)
- [6 Weblinks](#)

Suchen [\[Bearbeiten\]](#)

Um in einem B-Baum und damit auch in einem 2-3-4-Baum zu suchen, wird ein einfacher [Algorithmus](#) angewendet. Beginnend beim kleinsten (linksten) Element des *Wurzelknotens*:

- Vergleiche, ob der gesuchte Schlüssel gleich dem aktiven Element ist.
 - Wenn ja, Suche beendet.
 - Wenn nein, gehe zu 2.
- Vergleiche, ob der gesuchte Schlüssel kleiner ist als das aktive Element im *aktiven Knoten*.
 - Wenn ja, verzweige zum Kindknoten, der links vom gerade überprüften Element angehängt ist, setze dessen kleinstes Element als aktives Element und gehe zu 1. zurück.
 - Wenn nein, markiere das nächstgrößere Element im aktiven Knoten als aktives Element und gehe zu 1. zurück. Gibt es kein



größeres Element mehr im aktiven Knoten, verzweige zum Kindknoten rechts des aktiven Element und setze dessen kleinstes Element als aktives Element und gehe zurück zu 1.

Einfügen [\[Bearbeiten\]](#)

- Ein Knoten wird mit Elementen aufgefüllt, bis er drei Elemente enthält (vgl. B im Beispiel).
- Wenn ein viertes Element aufgenommen werden soll, wird der Knoten gespalten in einen Knoten mit zwei Elementen (J K im Beispiel), einen Knoten mit einem Element (M im Beispiel) und ein mittleres Element (L im Beispiel), das in den *Elternknoten* aufgenommen wird (vgl. Schritt 2 im Beispiel).
- Ist der Elternknoten voll besetzt, wird das Element im Baum weiter nach oben gereicht. Erreicht das Element die Wurzel des Baumes und ist dieser schon mit drei Elementen besetzt, wird eine neue Wurzel nach gleicher Aufteilungsregel erzeugt (vgl. Schritt 4 des Beispiels).

Es gibt eine weitere Möglichkeit, neue Elemente einzufügen, die sich von obiger Methode darin unterscheidet, zu welchem Zeitpunkt ein *4-Knoten* aufgespalten wird (*Split-Operation*). Bei dieser Methode wird während des *Traversierens* des Baums jeder gefundene 4-Knoten aufgespalten, es wird also das mittlere Element nach oben gereicht. Die Split-Operation wird also im schlimmsten Fall gerade einmal durchgeführt, während die erstgenannte Methode im schlimmsten Fall $\log(n)$ Split-Operationen durchführen muss.

Löschen [\[Bearbeiten\]](#)

Das Löschen eines beliebigen Elements kann immer auf das Löschen eines Elements in einem *Blatt* zurückgeführt werden. Dazu merkt man sich die Position des Elements innerhalb des Knotens. Ist die Position i , so wird im *Unterbaum* i des Knotens das Blatt gesucht, das sich am weitesten rechts befindet, dort vertauscht man das größte Element mit dem zu löschenden Element. Nun braucht nur noch das Element aus dem Blatt gelöscht zu werden, wobei drei Fälle unterschieden werden müssen:

- Das Blatt besitzt mehr als ein Element. In diesem Fall kann das Element einfach entfernt werden.
- Das Blatt enthält nur ein Element. In einem *Nachbarknoten* (Knoten mit gleichem *Vorgänger*) gibt es aber mindestens zwei Elemente. Es kann ein Element vom Nachbarknoten *ausgeliehen* werden. Der Schlüssel wird in den *Vorgängerknoten* verschoben, wobei das *Vorgängerelement* des zu löschenden Elements an dessen Position verschoben wird und dieses ersetzt.
- Das Blatt hat nur ein Element. Auch die Nachbarknoten haben nur ein Element. Das Element wird entfernt und sein Vorgängerelement wird mit einem Nachbarelement zusammengelegt. Falls der Vorgängerknoten selbst nur ein einziges Element besitzt, wird dieselbe Operation auf höherer Ebene durchgeführt.

Varianten [\[Bearbeiten\]](#)

2-3-4-Bäume werden beispielsweise durch [Rot-Schwarz-Bäume](#) implementiert.

Literatur [\[Bearbeiten\]](#)

- D. Maier, S. C. Salveter: *Hysterical B-trees*. In: *Information Processing Letters* 12, 1981, S. 199-202

- S. Huddleston, K. Mehlhorn: *A New Data Structure for Representing Sorted Lists* . In *Acta Informatica* 17, 1982, S. 157-184

Weblinks [\[Bearbeiten\]](#)

- [Arbeitsweise eines 2-3-4-Baumes](#)  (englisch, [Java-Applet](#))

Kategorie: [Suchbaum](#)

Diese Seite wurde zuletzt am 3. April 2013 um 14:38 Uhr geändert.

[Abrufstatistik](#)

Der Text ist unter der Lizenz „[Creative Commons Attribution/Share Alike](#)“ verfügbar; Informationen zum Lizenzstatus eingebundener Mediendateien (etwa Bilder oder Videos) können im Regelfall durch Anklicken dieser abgerufen werden. Möglicherweise unterliegen die Inhalte jeweils zusätzlichen Bedingungen. Durch die Nutzung dieser Website erklären Sie sich mit den [Nutzungsbedingungen](#) und der [Datenschutzrichtlinie](#) einverstanden. Wikipedia® ist eine eingetragene Marke der Wikimedia Foundation Inc.

[Datenschutz](#) [Über Wikipedia](#) [Impressum](#) [Entwickler](#) [Mobile Ansicht](#)

