66115 / 2018 / Frühjahr

Thema 1 / Aufgabe 5

(*Binärer Suchbaum*)

Stichwörter: AVL-Baum

Hinweis: Wir betrachten in dieser Aufgabe binäre Suchbäume, bei denen jeder innere Knoten genau zwei Kinder hat. Schlüssel werden nur in den inneren Knoten gespeichert - die Blätter speichern keinerlei Informationen.

(a) Welche Eigenschaften muss ein binärer Suchbaum haben, damit er ein AVL-Baum ist?

Lösungsvorschlag

Er muss die zusätzliche Eigenschaft haben, dass sich an jedem Knoten die Höhe der beiden Teilbäume um höchstens eins unterscheidet

(b) Mit $n(h)$ bezeichnen wir die minimale Anzahl innerer Knoten eines AVL-Baums der Höhe h.

(i) Begründen Sie, dass $n(1) = 1$ und $n(2) = 2$.

(ii) Begründen Sie, dass $n(h) = 1 + n(h-1) + n(h-2)$.

(iii) Folgern Sie, dass $n(h) > 2^{\frac{h}{2}-1}$.

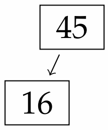
(c) Warum ist die Höhe jedes AVL-Baums mit n inneren Knoten $O(\log n)$?

(d) Fügen Sie die Elemente (45, 16, 79, 31, 51, 87, 49, 61) in der angegebenen Reihenfolge in einen anfangs leeren binären Suchbaum ein (ohne Rebalancierungen). Zeichnen Sie den resultierenden Suchbaum nach jeder Einfügeoperation.

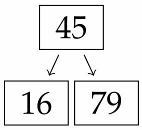
Lösungsvorschlag

Einfügen von „45“:

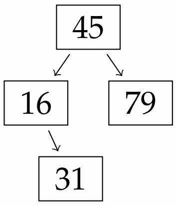
Einfügen von „16“:



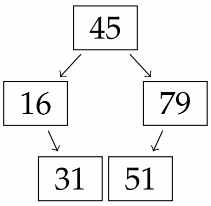
Einfügen von „79“:



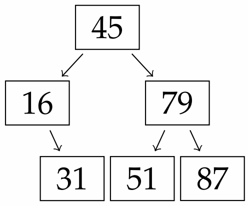
Einfügen von „31“:



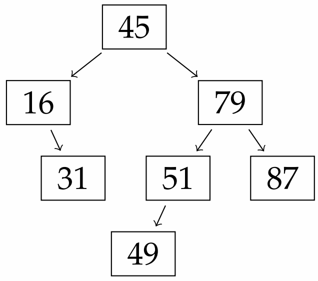
Einfügen von „51“:



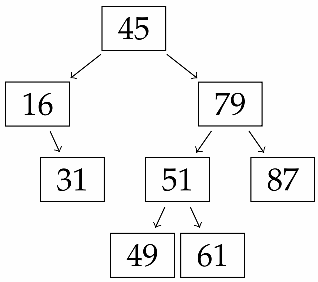
Einfügen von „87“:



Einfügen von „49“:



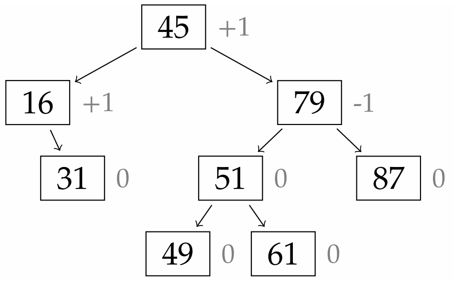
Einfügen von „61“:



(e) Ist der resultierende Suchbaum aus Teilaufgabe 5.4 ein AVL-Baum? Begründen Sie Ihre Antwort.

Lösungsvorschlag

Ja, wie in der untenstehenden Grafik zu sehen ist, unterscheiden sich die Höhe der Teilbäume von allen Knoten nur um höchstens eins.



(f) Das Einfügen in einen AVL-Baum funktioniert (zunächst) wie beim binären Suchbaum durch Erweitern eines äußeren Knotens w:

vor dem Einfügen von 54 nach dem Einfügen von 54

Anschließend wird die AVL-Baum Eingenschaft (falls notwending) durch eine (Doppel-)Rotation wiederhergestellt: Wenn z der erste Knoten auf dem Pfad P von w zur Wurzel ist, der nicht balanciert ist, y das Kind von z auf P und r das Kind von y auf P, und wenn (a, b,c) die Inorder-Reihenfolge von x, y, 2 ist, dann führen wir die Rotation aus, die benötigt wird, um b zum obersten Knoten der drei zu machen.

Die folgende Illustration zeigt den Fall, dass key(y) < key(x) < key(z), d. h. (a,b,c) = (y,x,z), wobei w ein Knoten in Ty ist.

Sei Ah die Höhe des Teilbaums 73. Für i = 0,1,2 sei h; die Höhe des Teilbaums T; und für v = 2,y,z sei h, die Höhe des Teilbaums mit der Wurzel v vor der Restrukturierung. Begründen Sie, dass

(i) $h_0 = h$

(ii) $h_1 = h - 1$

(iii) $h_2 = h$

(iv) $h_x = h + 1$

(v) $h_y = h + 2$

(vi) $h_z = h + 3$

(g) Welche Höhe haben die Teilbäume mit den Wurzeln x, y, z nach der Restrukturierung? Begründen Sie Ihre Antworten.

(h) Begründen Sie, dass die oben gezeigte Doppelrotation die AVL-Baum-Eigenschaft wiederherstellt.

(i) Beschreiben Sie, wie ein binärer Baum der Höhe h in einem Array repräsentiert werden kann. Wie viel Speicherplatz ist für so eine Darstellung erforderlich?

(j) Warum verwendet man bei der Implementierung von AVL-Bäumen eine verzeigerte Struktur und nicht eine Array-basierte Repräsentation?

Die Bschlangaul-Sammlung
Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.

Hilf mit! Die Hermine schafft das nicht alleine! Das ist ein Community-Projekt. Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an hermine.bschlangaul@gmx.net.Der TEX-Quelltext dieses Dokuments kann unter folgender URL aufgerufen werden: `https://github.com/hbschlang/lehramt-informatik/blob/main/Staatsexamen/66115/2018/09/Thema-1/Aufgabe-5.tex`