



Bruderbäume werden mit steigendem Grad der (gedachten Mehrweg-) Knoten effizienter bei der Suche

Ein balancierter Baum ist nicht unbedingt vollständig

In manchen Anwendungsfällen können lineare Listen bei der Suche effizienter sein als balancierte Suchbäume

In einem AVL-Baum unterscheidet sich die Anzahl der Knoten im linken Unterbaum maximal um 1 vom rechten Unterbaum

Die Auswertung eines arithmetischen Ausdrucks (dargestellt als Syntaxbaum) erfolgt "in order"

Gibt man die Knoten beliebiger Suchbäume "in order" aus, so ist die Ausgabe automatisch sortiert

Bei der Umwandlung von Mehrwegbäumen in Bruderbäume bleibt die logarithmische Suchzeit erhalten


#### 4) Einfügen in Suchbaum

Ein binärer Suchbaum für Integer soll aufgebaut werden. Zeichnen Sie jeweils die Bäume auf die sich zwischen den einzelnen Schritten ergeben.

- Fügen Sie dazu folgende Elemente schrittweise in den Baum ein (Es soll dabei gelten: Elemente des linken Unterbaums < Wurzelement < Elemente des rechten Unterbaums):  
6, 12, 4, 13, 9, 7, 8, 5, 3, 1, 20
- Fügen Sie das Element 7 ein. Geht das? Was müssen Sie ändern? Welche 2 Möglichkeiten gibt es dann? Welche Tiefe haben die beiden Bäume?
- Entfernen Sie nacheinander die Elemente 1, 9, 6, 7 vom Baum mit der geringeren Tiefe. Wie sieht dieser Baum nun aus? Lässt sich eindeutig entscheiden welches der beiden Elemente "7" entfernt werden muss?