# 4.3 AVL-Bäume

Unter der Balance eines Knotens versteht man ein Verhältnis, in dem seine beiden Unterbäume zueinander stehen.

Es sei u ein Knoten und es seien $UB_{links}$ und $UB_{rechts}$ seine beiden Unterbäume. Dann bezeichnet $|V_{UB_{links}}|$ die *Anzahl der Knoten* im linken Unterbaum von u und $|V_{UB_{links}}| + |V_{UB_{rechts}}|$ die Anzahl aller Knoten, die sich unterhalb des Knotens u befinden. Wie üblich sei h(T) die *Höhe* eines Baums T. Dann kann man z.B. Folgendes betrachten.



## Definition:

Es sei $\alpha$ eine Zahl aus dem reellen Intervall (0, ½].
Ein Knoten u heißt **$\alpha$-gewichtsbalanciert**, wenn gilt

$$\alpha \leq \frac{|V_{UB_{links}}| + 1}{|V_{UB_{links}}| + |V_{UB_{rechts}}| + 2} \leq 1 - \alpha$$

Den Ausdruck $\beta(u) = \dfrac{|V_{UB_{links}}| + 1}{|V_{UB_{links}}| + |V_{UB_{rechts}}| + 2}$

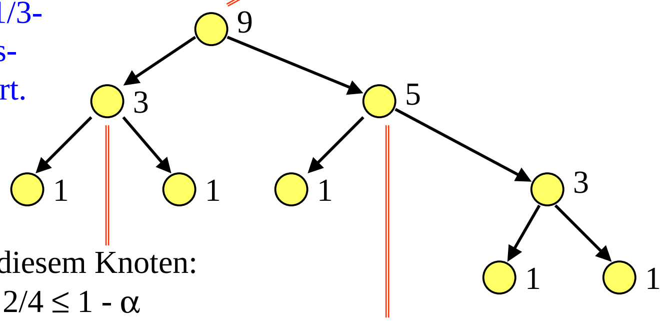
nennt man auch die Gewichts- oder Wurzelbalance von u.

Die Anzahlen der Knoten in den Unterbäumen werden hier jeweils um 1 erhöht, weil die Unterbäume leer sein können.

<u>Beispiel</u>: An jeden Knoten wird die Zahl der Knoten in dem Unterbaum, dessen Wurzel er ist, angegeben. Bestimme ein geeignetes $\alpha$.

Am Wurzelknoten gilt:
$\alpha \leq 4/10 \leq 1 - \alpha$ und $\alpha \leq 6/10 \leq 1 - \alpha$

Dieser Baum ist also 1/3-gewichts-balanciert.



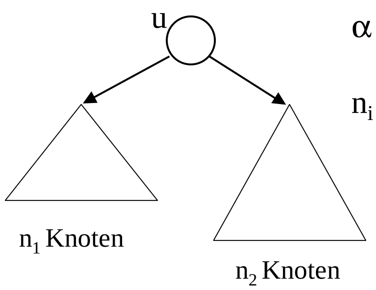

An diesem Knoten:
$\alpha \leq 2/4 \leq 1 - \alpha$
und $\alpha \leq 2/4 \leq 1 - \alpha$

An diesem Knoten gilt:
$\alpha \leq 2/6 \leq 1 - \alpha$
und $\alpha \leq 4/6 \leq 1 - \alpha$

usw.



Für $\alpha$-gewichtsbalancierte Bäume T mit n Knoten gilt an der Wurzel u:




$$\alpha \leq \frac{n_i + 1}{n + 1} \leq 1 - \alpha, \text{ d.h.:}$$

$$n_i \leq (1 - \alpha)\cdot(n + 1) - 1 \leq (1 - \alpha)\cdot n - \alpha$$

Beachte: Gesamtzahl der Knoten im Baum
$n = n_1 + n_2 + 1$

Dies gilt auch für das nächste Level, d.h.:

Anzahl der Knoten in einem Unterbaum zwei Level tiefer
$\leq (1 - \alpha)\cdot n_i - \alpha$
$\leq (1 - \alpha)\cdot((1 - \alpha)\cdot n - \alpha) - \alpha = (1 - \alpha)^2\cdot n - \alpha\,((1 - \alpha) + 1)$

Analog weiter einsetzen! Dies ergibt (beachte $\alpha > 0$):
Anzahl der Knoten in einem Unterbaum k Level tiefer
$\leq (1 - \alpha)^{\mathbf{k}}\cdot n - \alpha\,((1 - \alpha)^{\mathbf{k-1}} + (1 - \alpha)^{\mathbf{k-2}} + (1 - \alpha)^{\mathbf{1}} + (1 - \alpha)^{\mathbf{0}})$
$= (1 - \alpha)^{\mathbf{k}}\cdot n - \alpha\,(1 - (1 - \alpha)^{\mathbf{k}}) / (1 - (1 - \alpha))$
$= (1 - \alpha)^{\mathbf{k}}\cdot n - (1 - (1 - \alpha)^{\mathbf{k}})$
$< (1 - \alpha)^{\mathbf{k}}\cdot n$

Wenn $(1 - \alpha)^{\mathbf{k}}\cdot n < 2$ geworden ist, gibt es höchstens noch einen Knoten und dieser muss dann ein Blatt sein.



Aus $(1 - \alpha)^{\mathbf{k}}\cdot n < 2$ folgt mit $z := 1/(1 - \alpha)$:

$n < 2\cdot z^{\mathbf{k}}$, d.h., $\log(n/2) < k\cdot\log(z) = - k\cdot\log(1-\alpha)$

Suche das kleinste k, für das diese Ungleichung zutrifft:

$\mathbf{k = 1 - (\log(n) - 1) / \log(1-\alpha)} \in O(\log(n))$.

Dieses k ist eine obere Schranke für die Höhe des Baums.
Somit haben wir gezeigt:
Die Höhe in einem $\alpha$-gewichtsbalancierten Baum mit n Knoten ist höchstens $1 - (\log(n) - 1) / \log(1-\alpha)$ ,
d.h., die Höhe ist von der Größenordnung $O(\log(n))$.
Je mehr $\alpha$ sich der Zahl 0.5 nähert, um so mehr nähert sich die Höhe dem optimalen Wert $\log(n)$.

Kann man die Operationen FIND, INSERT und DELETE auf solchen $\alpha$-gewichtsbalancierten Bäumen so ausführen, dass diese Operationen schnell durchführbar sind (z.B. in O(log(n)) Schritten) und dass nach Ausführung jeder Operation der entstandene Baum wieder ein $\alpha$-gewichtsbalancierter Baum ist?

Ja, das geht tatsächlich.

Wir führen dies hier nicht durch, sondern verweisen auf die Literatur.

Statt dessen behandeln wir die höhenbalancierten Bäume.



## Definition:

Ein binärer Baum heißt **AVL-Baum** (oder höhenbalancierter Baum), wenn für jeden Knoten u gilt:

$h(UB_{rechts}) - h(UB_{links}) =$

( Höhe des rechten Unterbaums von u -

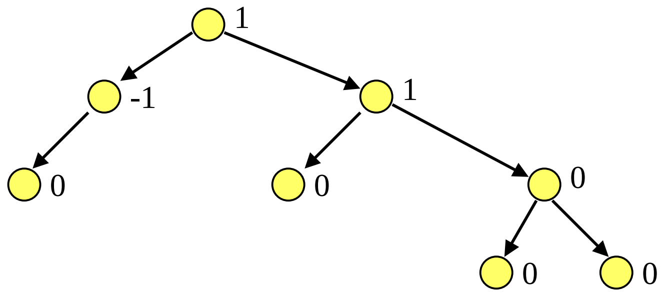
Höhe des linken Unterbaums von u ) $\in \{-1, 0, 1\}$,

d.h., für jeden Knoten unterscheiden sich die Höhen seiner Unterbäume höchstens um 1.

$h(UB_{rechts}) - h(UB_{links})$ heißt auch der Balancefaktor von u.

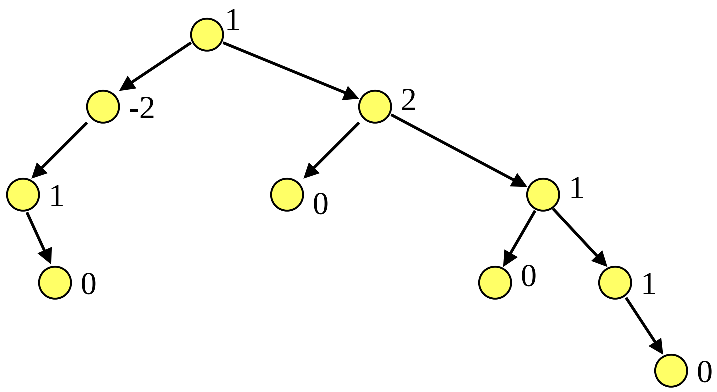
AVL-Bäume wurden benannt nach ihren beiden Erfindern Adelson-Velski und Landis.

Beispiel: Der bereits oben betrachtete Baum ist höhenbalanciert, d.h. ein AVL-Baum. Wir tragen die Balancefaktoren an jedem Knoten ein:






Beispiel: dagegen ist folgender Baum *nicht* höhenbalanciert:



Datentypen für AVL-Bäume,
vgl. Skript Plödereder, Folie 132:

```
type Knotentyp;
type Baumtyp is access Knotentyp;
type Balancetyp is (-1, 0, 1);
type Knotentyp is record
    Schluessel: integer;
    Balance: Balancetyp;
    UBLinks, UBRechts: Baumtyp;
    end record;
```

Hinweis: Herr Plödereder verwendet:
type Balancetyp is (LinksL, Neutral, RechtsL)



FIND: Wie beim Suchbaum.

INSERT:
1. Füge den Schlüssel x als neues Blatt in den AVL-Baum ein.
2. Gehe den Suchpfad zurück und ändere die Balancefaktoren.

DELETE:
1. Finde den Schlüssel x im AVL-Baum und entferne seinen Knoten wie bei einem Suchbaum.
2. Gehe den Weg zur Wurzel zurück ab der untersten Stelle, an der eine Veränderung statt fand, und korrigiere entlang des Pfads die Balancefaktoren (evtl. muss man bis zur Wurzel alle Balancefaktoren ändern).

Diese Operationen erfordern höchstens O(log(n)) Schritte.