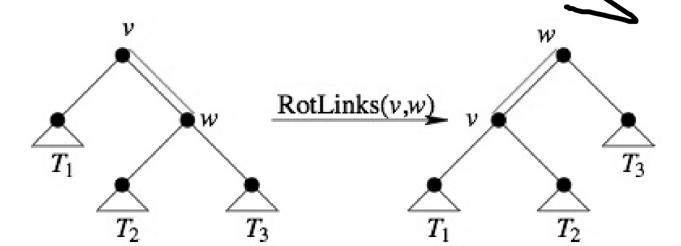
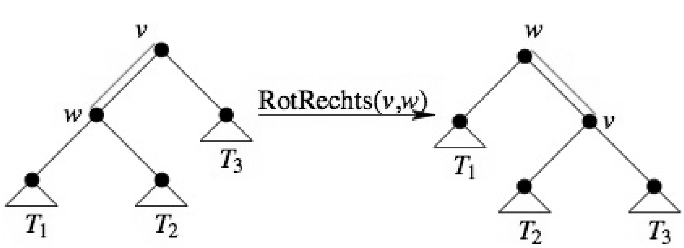
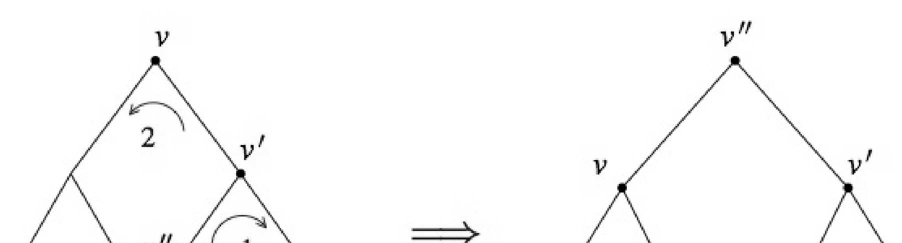
Zur Erinnerung:







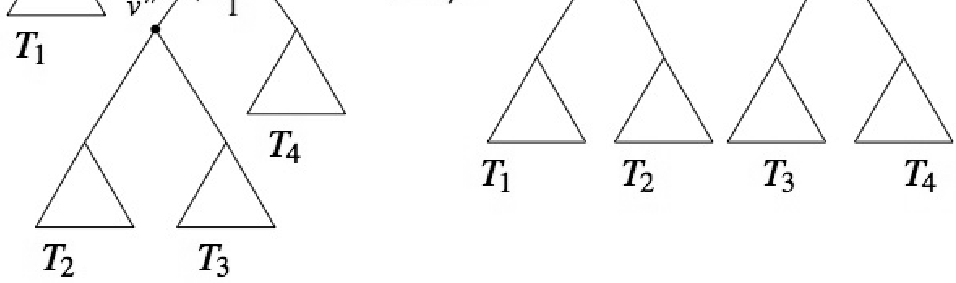


Abbildung 5.6: Doppellinksrotation DRotLinks(v, v', v'').

Lemma 5.2 (Rotationslemma für AVL-Bäume) *Sei T ein Baum mit Wurzel v. Der linke und der rechte Teilbaum T_ℓ, T_r von T seien AVL-Bäume. Die Wurzel v sei geringfügig außer Balance, d.h. $|\beta(v)| = 2$. Dann folgt:*

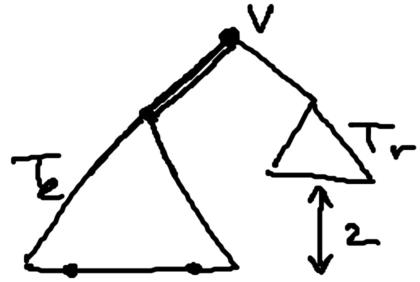
a) *T kann durch eine Rotation bzw. Doppelrotation in einen AVL-Baum T' überführt werden mit $h(T') \le h(T)$.*

b) *Die Art der Rotation (einfach links, ..., doppelt rechts) kann mit $O(1)$ Aufwand ermittelt werden.*

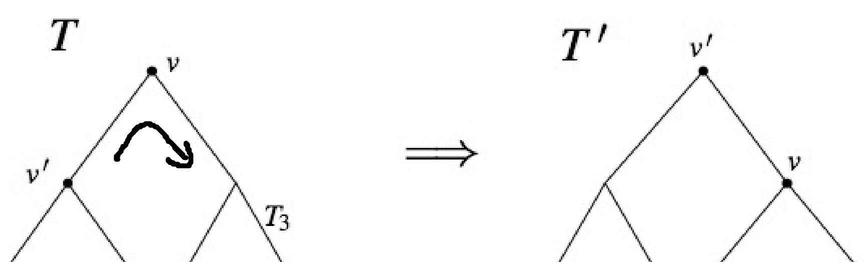
c) *Die Rotation kann in $O(1)$ Aufwand durchgeführt werden.* ✓

d) *Alle veränderten Balancen in T' können mit $O(1)$ Aufwand aus denen in T berechnet werden.*

<u>Beweis:</u> Unterscheide 4 Fälle in Abhängigkeit von den ersten beiden Kanten des Weges von der Wurzel v in die tiefste Tiefe des Baumes



1. Fall: **1.) Der Weg von v in die größte Tiefe geht links – links (LL)**



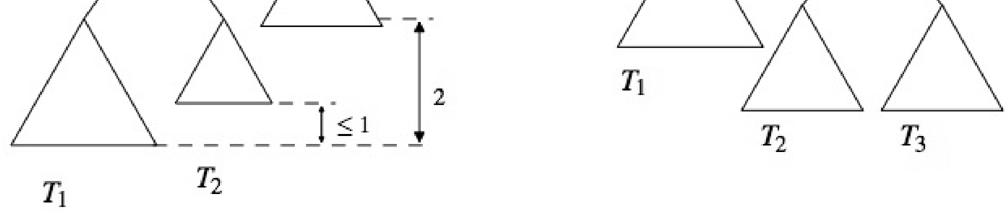


Abbildung 5.8: Rechtsrotation bei dem Fall LL.

Veränderte Situation in T':

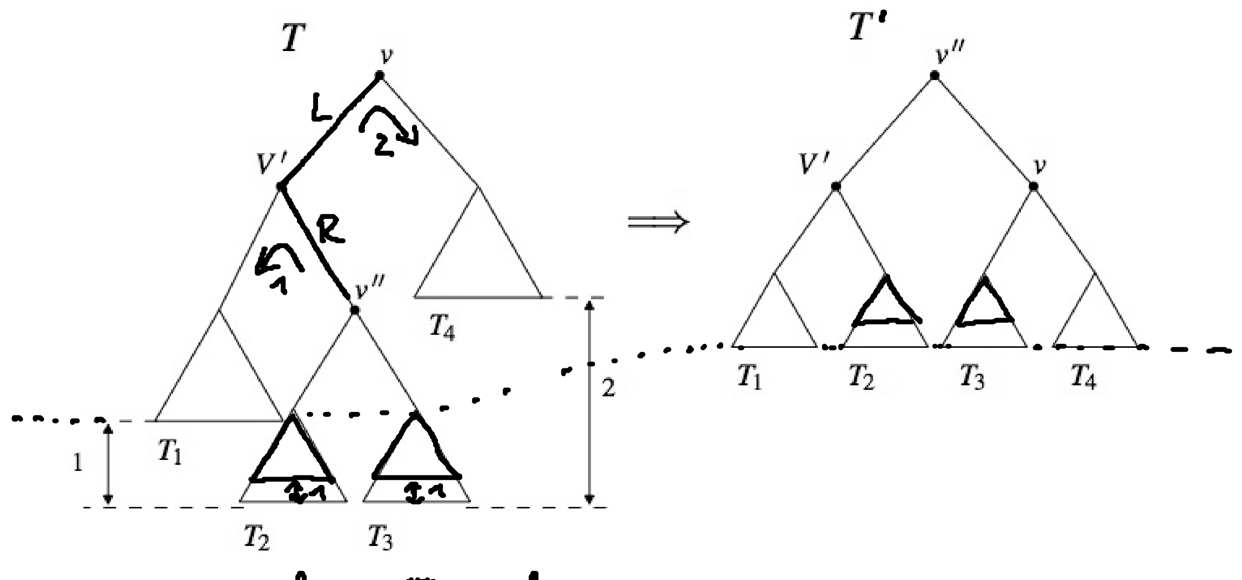
$$\beta'(v) = \begin{cases} -1 & \text{falls } \beta(v') = 0 \\ 0 & \text{falls } \beta(v') = -1 \end{cases}$$

$$\beta'(v') = \begin{cases} 1 & \text{falls } \beta(v') = 0 \\ 0 & \text{falls } \beta(v') = -1 \end{cases}$$

$$\beta'(u) = \beta(u) \quad \text{, für alle } u \neq v, v'$$

$$h(T') = \begin{cases} h(T) & \text{falls } \beta(v') = 0 \\ h(T) - 1 & \text{falls } \beta(v') = -1 \end{cases}$$

<u>2. Fall:</u> Der Weg in die tiefste Tiefe geht LR und nicht LL.



oder oder
keines von beiden

Abbildung 5.9: Doppelrechtsrotation bei dem Fall LR und nicht LL.

Veränderung von T' gegenüber T:

$$\beta'(v'') = \begin{cases} 0 & \text{falls } \beta(v'') = 0 \\ 0 & \text{falls } \beta(v'') = 1 \\ 0 & \text{falls } \beta(v'') = -1 \end{cases}$$

$$\beta'(v') = \begin{cases} 0 & \text{falls } \beta(v'') = 0 \\ -1 & \text{falls } \beta(v'') = 1 \\ 0 & \text{falls } \beta(v'') = -1 \end{cases}$$

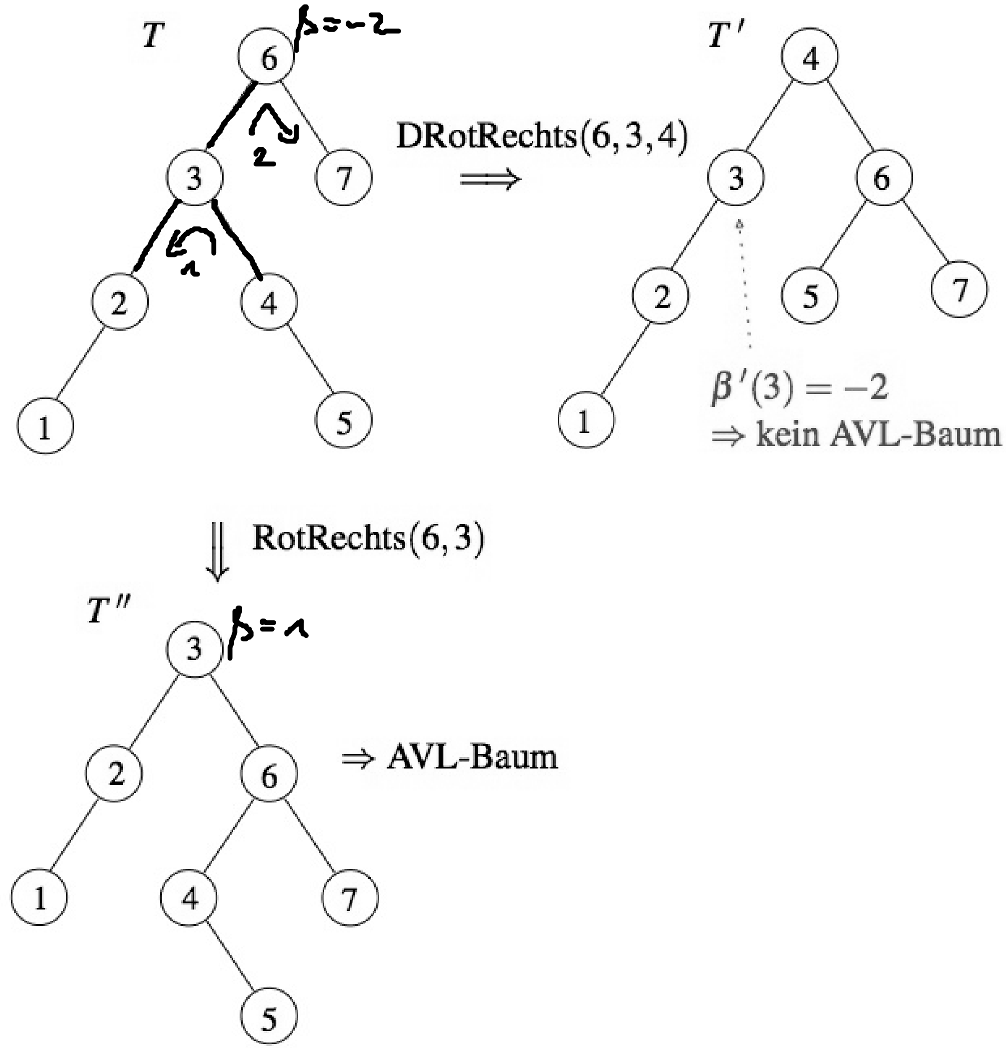
$$\beta'(v) = \begin{cases} 0 & \text{falls } \beta(v'') = 0 \\ 0 & \text{falls } \beta(v'') = 1 \\ 1 & \text{falls } \beta(v'') = -1 \end{cases}$$

$$\beta'(u) = \beta(u) \quad \text{, für alle } u \neq v, v', v''$$

$$h(T') = h(T) - 1$$

Beachte: Die neuen Balancen in T' und die neue Höhe $h(T')$ können im Fall 1 und 2 jeweils in $\mathcal{O}(1)$ berechnet werden.

Bemerkung: Falls der Weg in die tiefste Tiefe sowohl LL als auch LR gehen kann, so muss wie in Fall 1 verfahren werden, denn:



Fall 3: RR, verfahre analog zu Fall 1, d.h. RotLinks.

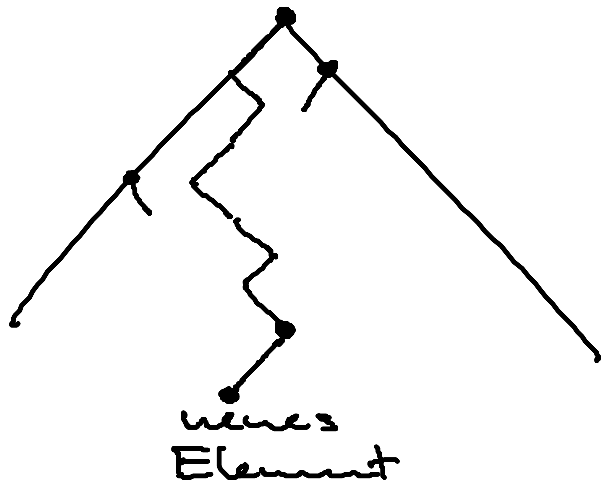
Fall 4: RL aber nicht RR, verfahre analog zu Fall 2, d.h. DRotLinks. $\square$

Basisoperationen in AVL-Bäumen

1) Suchen: Wie bei allgemeinen Suchbäumen

Aufwand $\mathcal{O}(h(T)) \underset{\substack{\uparrow \\ \text{wegen AVL-Baum}}}{=} \mathcal{O}(\log n)$

2) Einfügen: Suche nach Schlüssel des neuen Elements endet erfolglos im Blatt. Hänge neues Element als linkes oder rechtes Kind an dieses Blatt.



Beachte: Balancen haben sich höchstens im Knoten entlang der orangenen Spur geändert.

$\Rightarrow$ gehe von neuem Element entlang dieser Spur zurück zur Wurzel und update die Balancen. Falls $\beta(v) \in \{-2, 2\}$, repariere dies sofort mit Hilfe des Rotationslemmas.

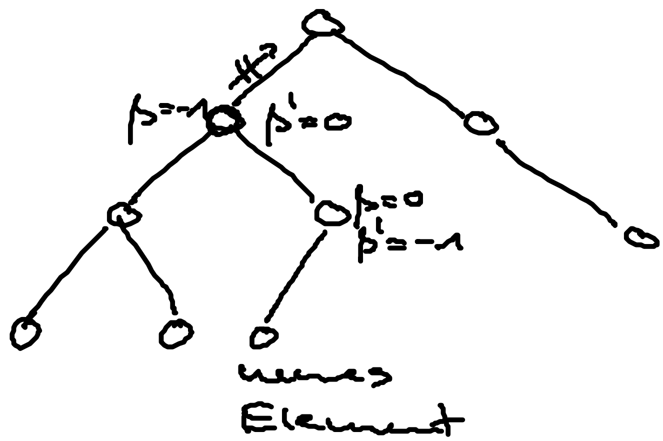
Aufwand: Suchen $\mathcal{O}(\log n)$

Updaten und Korrigieren der Balancen entlang der Spur: $\mathcal{O}(\log n)$.

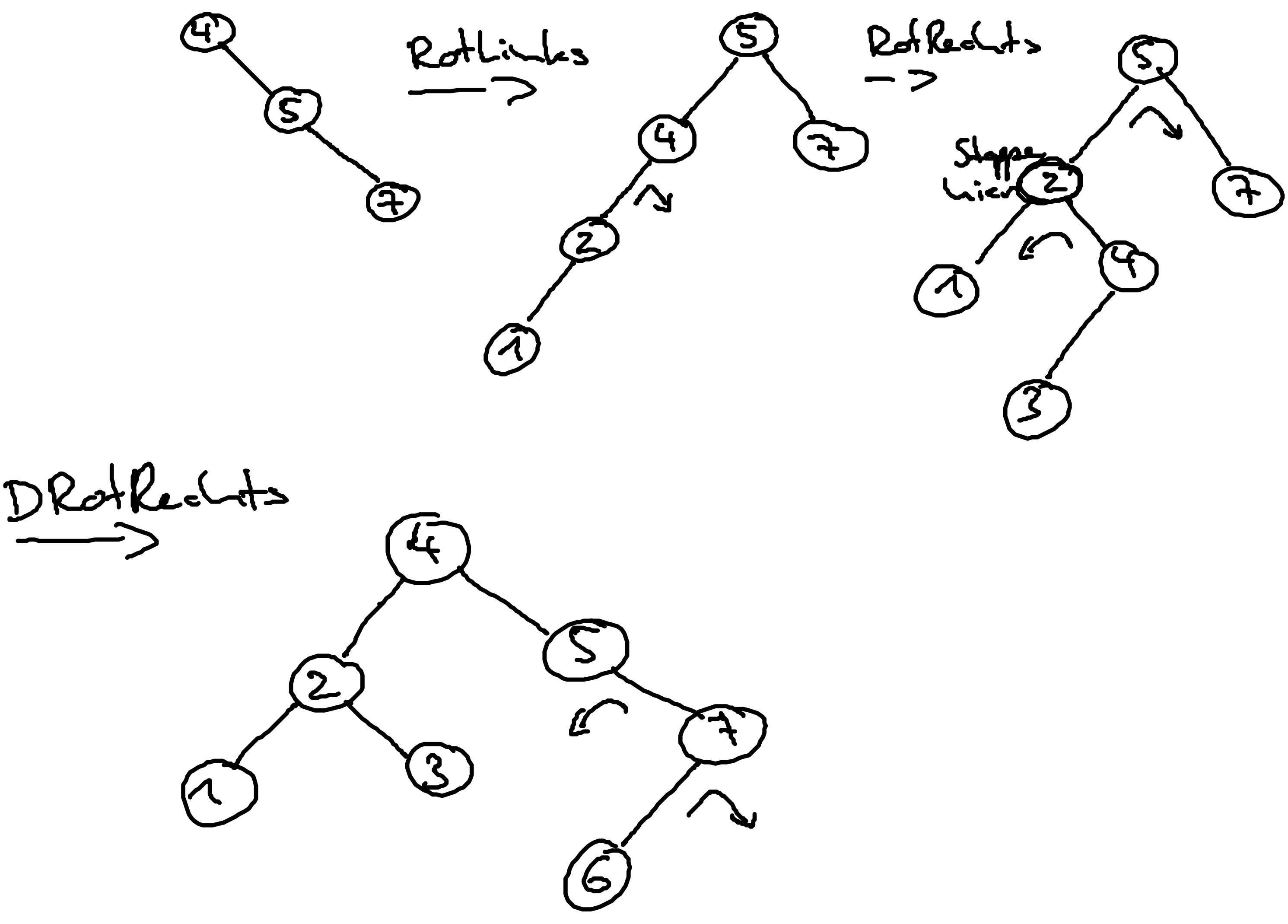
Praktische Verbesserung: Stellt man beim Verfolgen der Spur fest, dass sich die

Tiefe des Teilbaums an einem Knoten v nicht verändert hat, so kann man an diesen abbrechen.

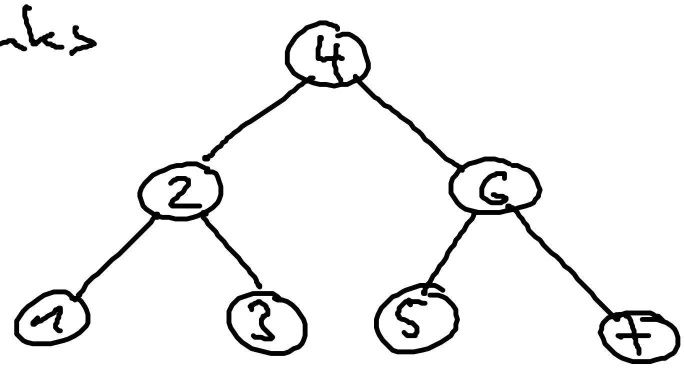
Bsp:



Bsp: Einfügen von 4, 5, 7, 2, 1, 3, 6 in leeren AVL-Baum:

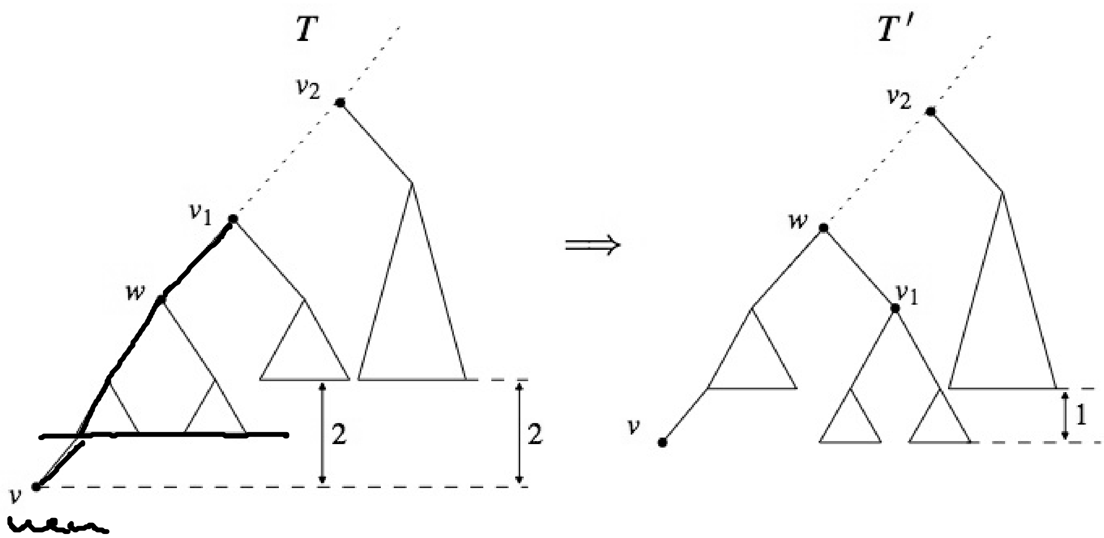


DRotLinks $\longrightarrow$



Satz: Findet man beim Verfolgen der Spur nach oben einen Knoten v mit $\beta(v) \in \{2, -2\}$ und behebt dies mit Hilfe des Rotationslemmas, so kann man danach stoppen, da die Höhe des Teilbaums an diesem Knoten wieder der Ausgangshöhe entspricht.

Beweisskizze:



Es sei v_1 der erste gefundene Knoten

mit $\beta(v_1) \in \{-2, 2\}$. $\rightarrow$ Höhe des Teilbaums an v_1 ist um eins gewachsen (durch v).

Unterscheide jetzt zwei Fälle: LL und LR (RR, RL symmetrisch!)

LR: Die Tiefe des Baumes schrumpft um 1 (siehe Fall 2 Rotationslemma).

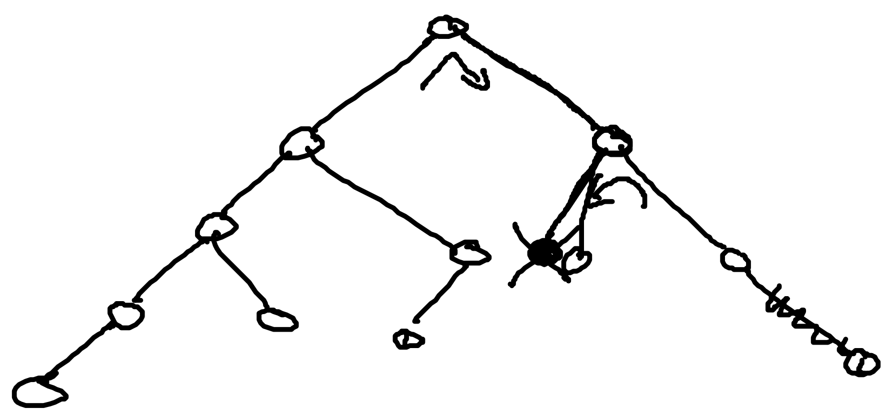
LL: Bemerke, dass die beiden Teilbäume von w vor dem Einfügen von v perfekt balanciert waren. Nach Fall 1 im Rotationslemma schrumpft daher die Tiefe des Teilbaums an dem Knoten um 1. ☐

Löschen: Genau wie in allgemeinen Suchbäumen. Aber: Balance-Eigenschaft möglicherweise verletzt. Gehe daher entlang der Spur vom gelöschten Element zurück zur Wurzel, update β und repariere. Aufwand: $\mathcal{O}(\log n)$

<u>Bemerkung</u>: Leider gilt das schöne Resultat für das Einfügen <u>nicht</u> für das Löschen.

<u>Denn:</u> Es gibt AVL-Bäume, so dass nach dem Löschen eines Blattes an jedem Knoten der Spur das Rotationslemma angewendet werden muss

<u>Bsp:</u>



Worst-Case Beispiel: Lösche das Blatt minimaler Tiefe in einem extremalen AVL-Baum.