

## AVL-Bäume (30 Punkte)

Fügen Sie in einen zunächst leeren AVL-Baum nacheinander (in dieser Reihenfolge) die Zahlen 10, 5, 6, 12, 14, 8 ein. Geben Sie alle Zwischenschritte an.

Gute Erklärvideos:

<https://www.youtube.com/watch?v=RGxOJ-syDkI>

<https://studyflix.de/informatik/avl-baum-1434>

In jedem Schritt Balance-Faktor **BF** für alle Knoten berechnen:

für jeden Unterbaum ausrechnen: Höhe rechter Teilbaum - Höhe linker Teilbaum

Also: Knoten mit einem Kind rechts = 1 (rechtes Kind) - 0 (kein linkes Kind) = 1

Knoten mit einem Kind links = 0 (kein rechtes Kind) - 1 (linkes Kind) = -1

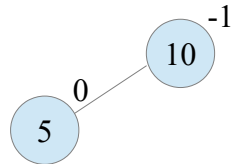
Grundsätzlich:

- umsortieren, sobald **BF** an einer Stelle  $> 1$  oder  $< -1$
- Transformation war richtig, wenn danach alle BF 0 oder 1 und die Kriterien des binären Suchbaums erfüllt sind (linkes Kind  $<$  Elternknoten  $<$  rechtes Kind)

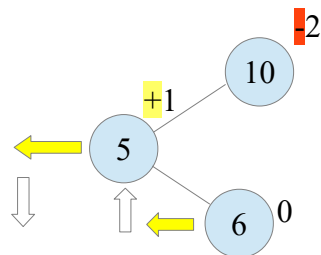
1)



2)



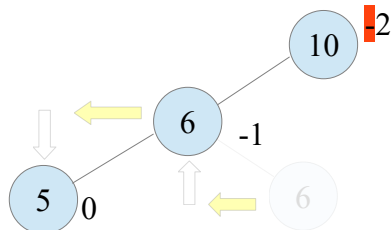
3)



AVL-Kriterium verletzt: **links-rechts**  
Drehung benötigt

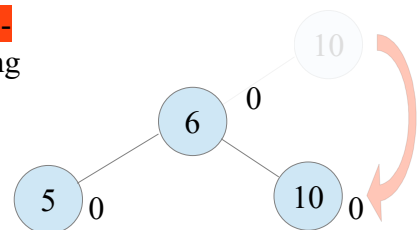
3a)

**Links**drehung  
um 5



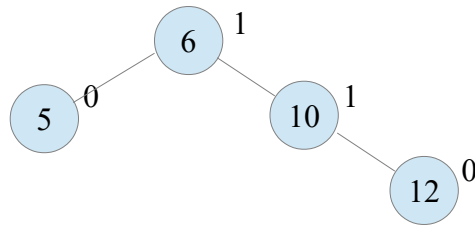
3b)

**Rechts-**  
drehung  
um 6

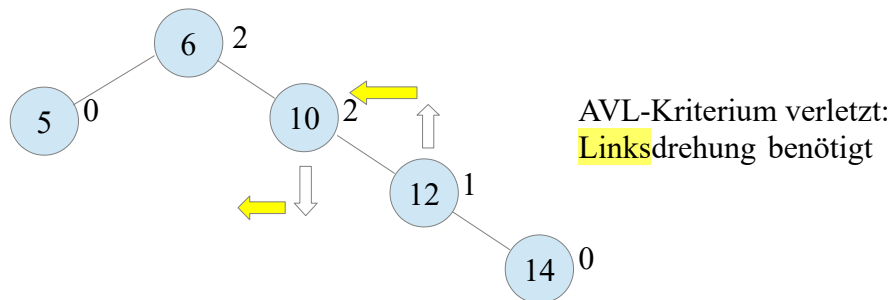


- **+/** = **links** / **rechts** → Links- Rechtsdrehung bei **+**1 im Kind & **-**2 im Elternknoten
  - 3) & 3a) Linksdrehung von Kind des Kindknotens um den Kindknoten → Kindknoten (Knoten mit **+**1) und Kind des Kindknotens wandern jeweils eine Stelle nach links (und dann hoch bzw. runter)
- 3b) Rechtsdrehung des Elternknotens (Knoten mit **-**2) um den Kindknoten

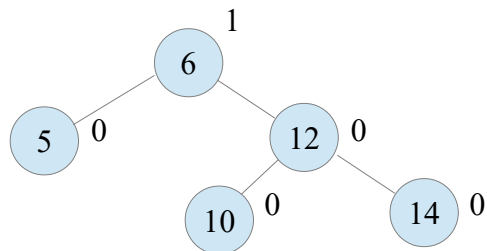
4)



5)

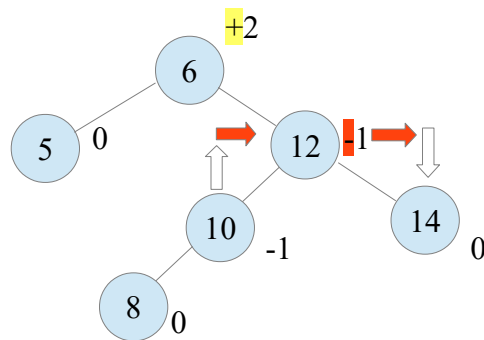


5a)



- $+$  = links  $\rightarrow$  Linksdrehung bei  $+$ 1 im Kind (Wert 12) &  $+$ 2 im Elternknoten (Wert 10)
  - Drehung um den Kindknoten (BF 1, Wert 12)
  - eigentlich links-links Drehung: der Elternknoten (BF 2, Wert 10) wandert nach links unten, der Kindknoten nach links oben
    - die Positionen ihrer anderen Kinder/Unterbäume (Wert 14) bleiben erhalten/werden positionsgetreu mit verschoben

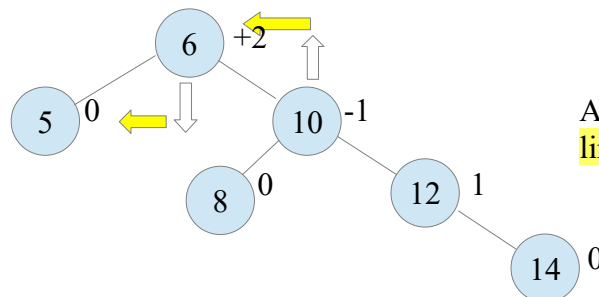
6)



AVL-Kriterium verletzt:  
rechts-links Drehung  
benötigt

- $- / + = \text{rechts} / \text{links}$  → Rechts- Linksrotation  $-1$  im Kind &  $+2$  im Elternknoten
  - 6) Rechtsdrehung des Kindknotens (Wert 12) um das Kind des Kindknotens (Wert 10)
    - Kind des Kindknotens (Wert 10) wandert 1 hoch und 1 nach rechts,
    - Kindknoten (Wert 12) wandert 1 nach rechts und 1 nach unten
    - die Positionen ihrer anderen Kinder/Unterbäume (Wert 14) bleiben erhalten/werden positionsgetreu mit verschoben

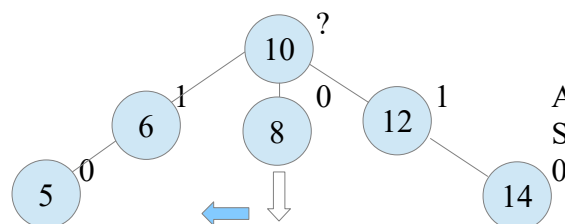
6a)



AVL-Kriterium verletzt: rechts-  
links Drehung benötigt

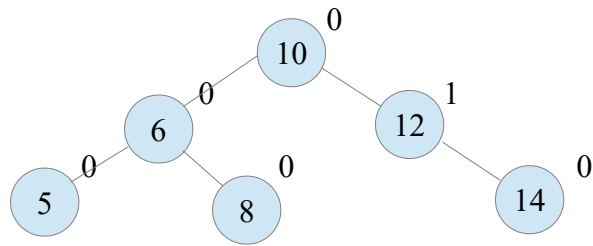
- 6a) Linksdrehung des Elternknotens um den Kindknoten:
  - Kindknoten (Wert 10) wandert 1 nach oben, 1 nach links
  - Elternknoten (Wert 6) wandert 1 nach unten, 1 nach links

6b)



AVL-Kriterium verletzt: binärer  
Suchbaum

6c)



Blatt, das die AVL-Kriterien verletzt wird als Kind an einen Knoten seiner Ebene eingefügt, so dass die Kriterien des binären Suchbaums erfüllt sind (linkes Kind < Elternknoten < rechtes Kind)