Definition 5.4.1:

Es sei α eine Zahl aus dem reellen Intervall (0, ½]. Ein Knoten u eines Binärbaums heißt α-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. Wenn ein Knoten α-gewichtsbalanciert ist und $0 \leq \alpha' \leq \alpha \leq ½$ gilt, dann ist der Knoten auch α'-gewichtsbalanciert.

Für das Folgende beachten Sie, dass hier "Höhe" und "Tiefe" synonym verwendet werden (5.2.4). T bezeichnet die Tiefe.

Definition 5.4.5:

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

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

("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 (= Tiefen) seiner Unterbäume höchstens um 1.

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

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

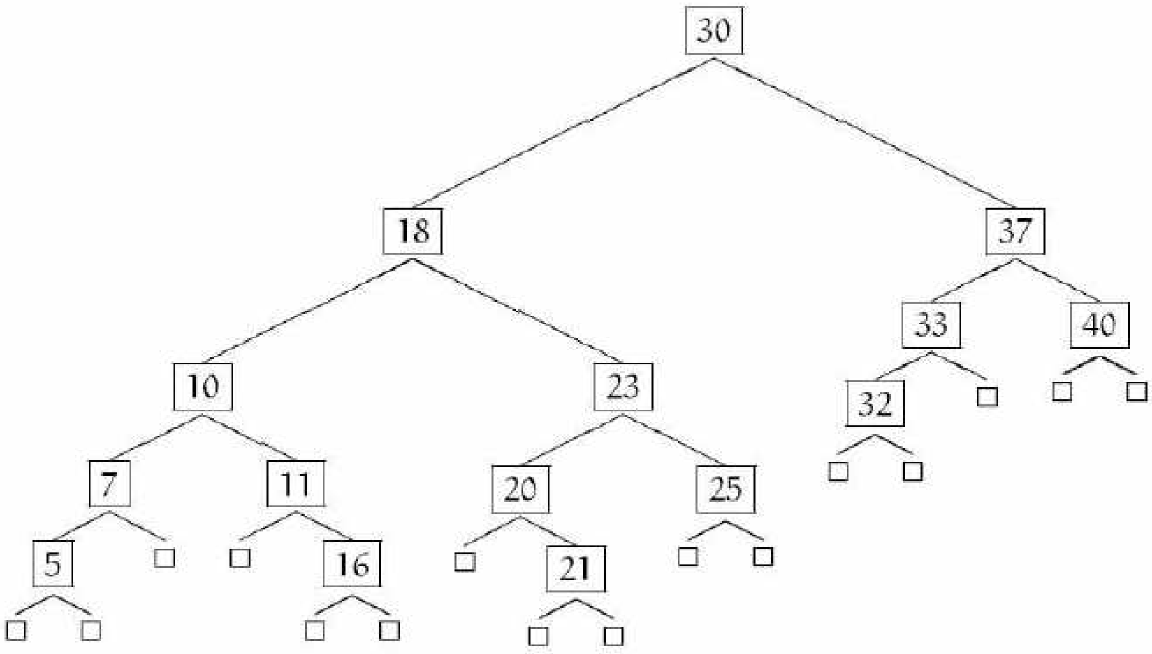
Aufgabe 1: (AVL-Bäume teilweise ehemalige Klausuraufgabe)

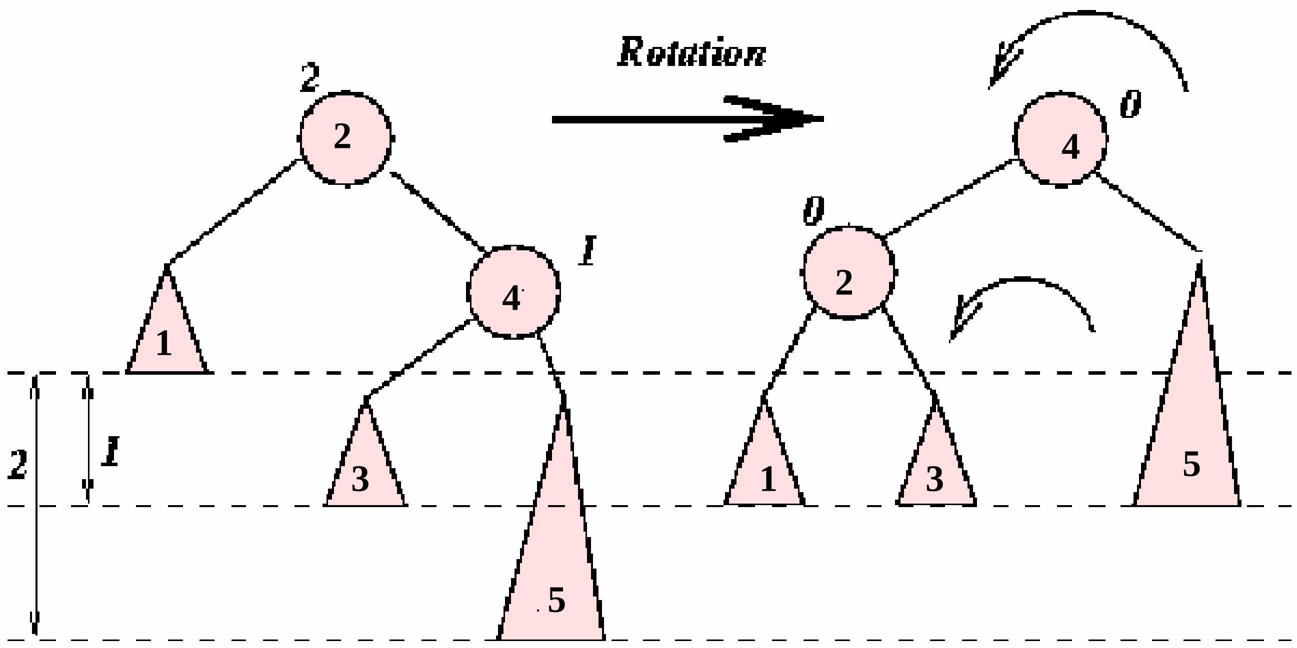
Ein AVL-Baum ist ein binärer Suchbaum, in dem sich für jeden Knoten p die Höhe der beiden Teilbäume von p um höchstens 1 unterscheidet. Solche Bäume heißen höhenbalanciert.

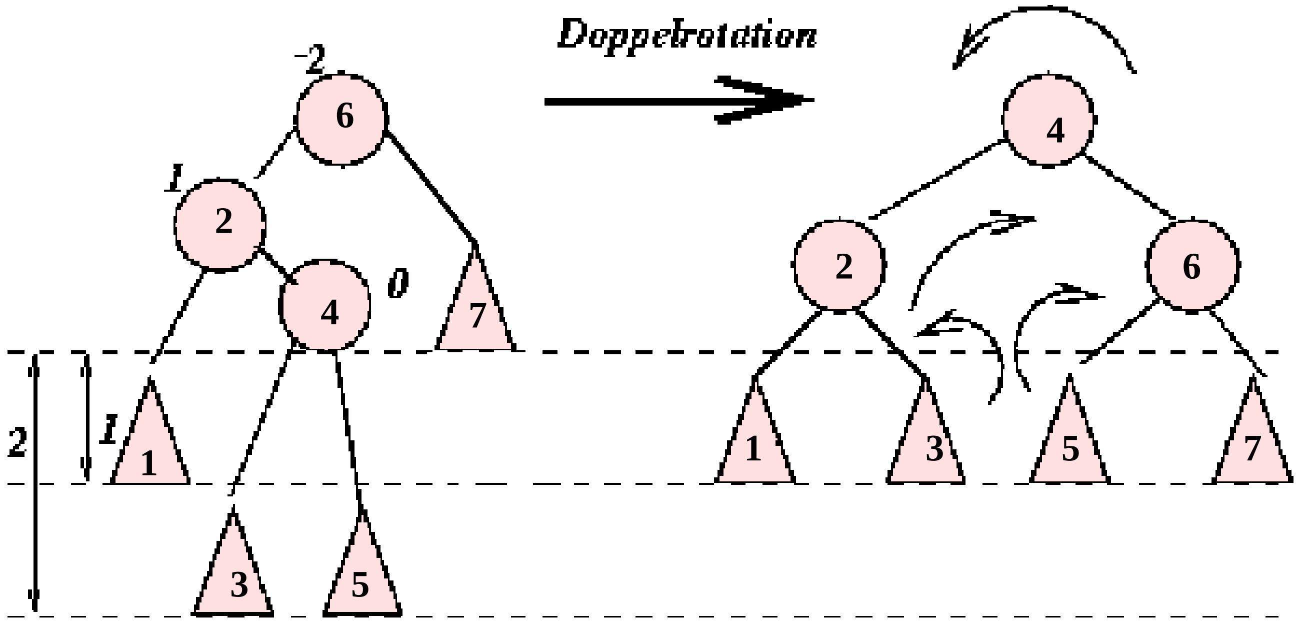
a) Fügen Sie folgende Schlüssel in der angegebenen Reihenfolge in einen AVL-Baum ein: S = (45; 25; 15; 13; 12; 11; 8; 7; 50; 53; 35; 63; 52; 67; 9; 10)

Geben Sie dabei bei jedem Einfügeschritt den entstandenen Baum sowie die angewandten Rotationen an.

b) Geben Sie für den AVL-Baum an, in welcher Reihenfolge die Schlüssel möglicherweise eingefügt wurden, wenn man davon ausgeht, dass keine Rotationen zur Rebalancierung notwendig waren.




Rotation


Doppelrotation

c) Löschen Sie aus dem obigen Baum sukzessiv die Schlüsselwerte 16, 30, 40 und 11. Geben wieder nach jedem Löschvorgang den entstandenen Baum sowie die benutzten Rotationen an.

d) Fügen Sie sukzessiv die Schlüssel 10, 20, 30, 40, 50, 60 und 70 in einen anfangs leeren AVL-Baum ein. Löschen Sie sukzessiv die Schlüssel 10, 20 und 40 aus dem Baum.

e) Bauen Sie sukzessive aus folgenden Schlüsseln einen AVL-Baum auf:

4, 5, 7, 2, 1 ,3, 6

f) Fügen Sie in einen anfangs leeren AVL-Baum nacheinander die Monatsnamen ein. Gehen Sie von einer lexikographischen Ordnung aus.

Aufgabe 2 (Höhe von **AVL-Bäumen**)

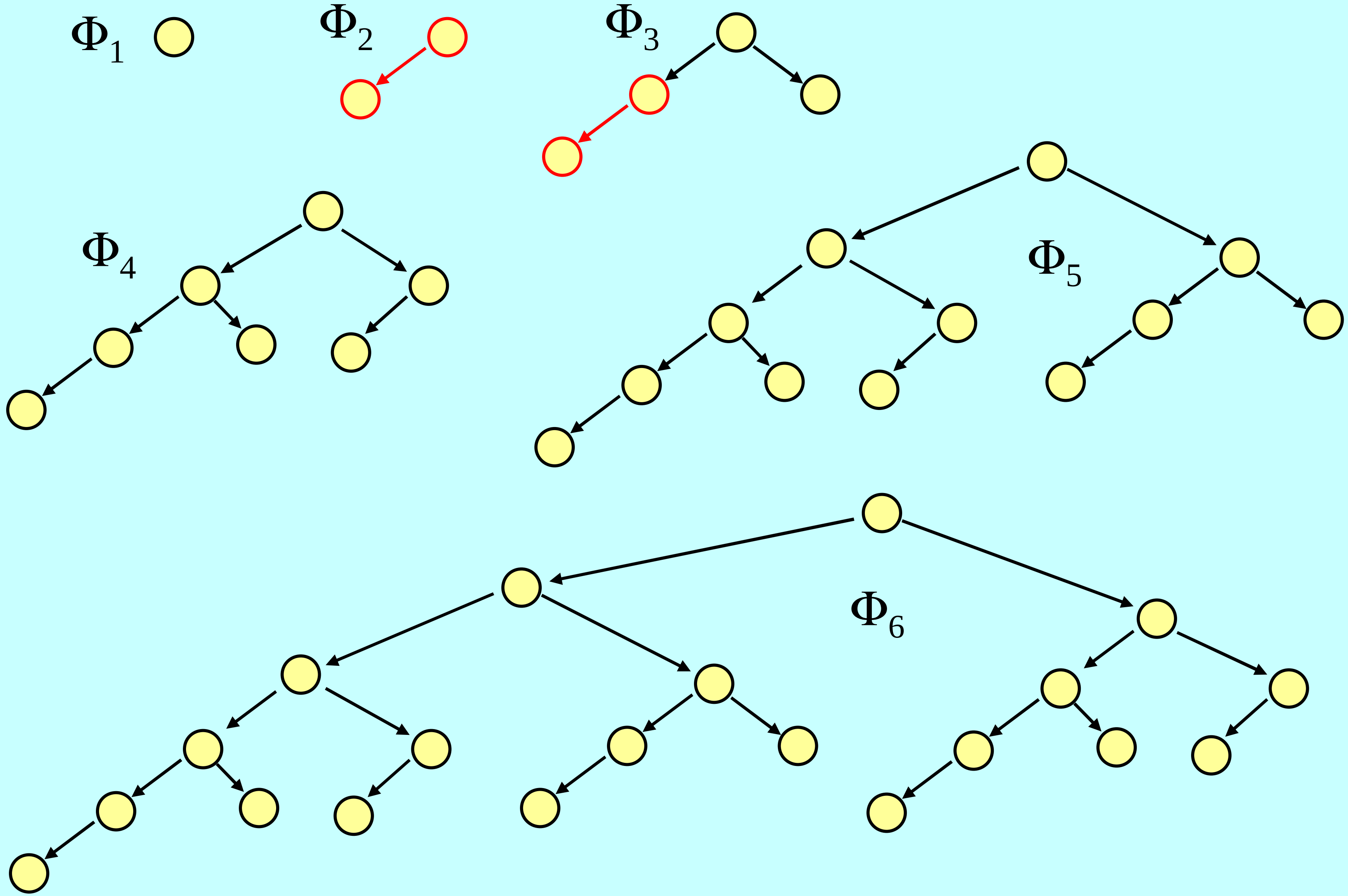
a) Gegeben sei ein AVL-Baum mit 10^5 Knoten, die vollständig im Hauptspeicher Platz finden, und mit einer Verarbeitungszeit von 10^{-6} Sekunden pro Vergleich. Wie lang dauert die erfolgreiche Suche nach einem Schlüssel im schlechtesten Fall?

b) Geben Sie einen AVL-Baum der Höhe 5 an, der möglichst wenig Knoten besitzt.

c) Zeichnen Sie einen AVL-Baum mit Tiefe 4, bei dem der rechte Teilbaum der Wurzel möglichst wenige Knoten enthält und der linke möglichst viele. Wie groß ist der Unterschied in der Anzahl der Knoten?

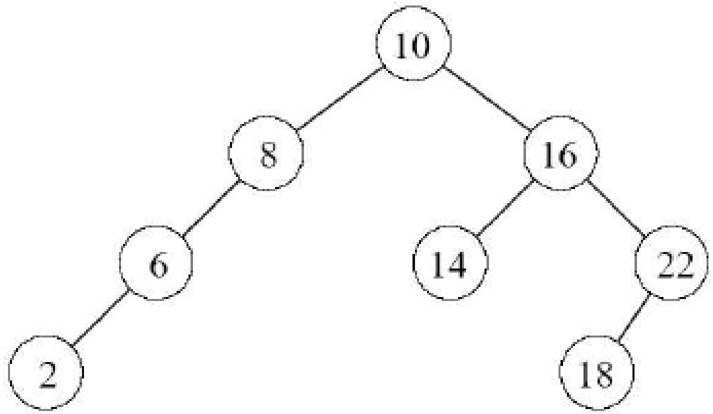
d) Ist ein AVL-Baum auch α-gewichtsbalanciert oder umgekehrt?

Fibonaccibäume AVL-Bäume maximaler Höhe



Aufgabe 3 (AVL-Bäume - ehemalige Klausuraufgabe)

Gegeben sei folgender AVL-Baum direkt nach dem Einfügen eines neuen Elements:



Ist die AVL-Eigenschaft in diesem Baum verletzt? Falls ja, führen Sie die notwendigen Maßnahmen durch, um die AVL-Bedingung wieder herzustellen, und zeichnen Sie den Baum in seinem korrigierten Zustand.

Führen Sie danach folgende Operationen in diesem AVL-Baum in der angegebenen Reihenfolge durch und zeichnen Sie den Baum nach jeder Operation. Achten Sie darauf, dass die AVL- Eigenschaft immer erhalten bleibt.

a) Fügen Sie 19 in den AVL-Baum ein.

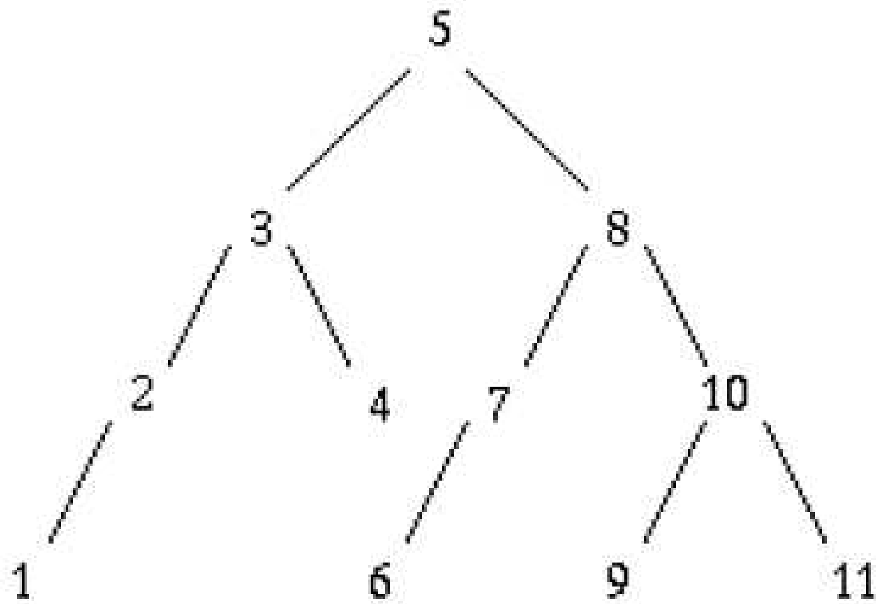
b) Fügen Sie 7 in den AVL-Baum ein.

c) Löschen Sie 22 aus dem AVL-Baum.

d) Fügen Sie 13 in den AVL-Baum ein.

Aufgabe 4 (Löschen in **AVL-Bäumen**)

Gegeben sei folgender AVL – Baum:



Löschen Sie aus dem Baum sukzessive die folgenden Schlüssel: 4, 8, 6, 5, 2

Definition B-Bäume

Definition 3.2.5.2:

Es sei $m \geq 2$ (teilweise verwenden wir auch $m=1$) eine natürliche Zahl. Ein geordneter nicht-leerer Baum, in dem jeder Knoten eine sortierte Folge von Schlüsseln aus einer geordneten Menge enthält, heißt
B-Baum der Ordnung m, wenn für alle Knoten gilt:

1. Jeder Knoten enthält höchstens 2m Schlüssel.
2. Die Wurzel enthält mindestens einen Schlüssel.
3. Jeder andere Knoten enthält mindestens m Schlüssel.
4. Ein Knoten mit k Schlüsseln besitzt entweder genau k+1 Kinder oder kein Kind.
5. Alle Blätter (=Knoten ohne Kinder) besitzen das gleiche Level.
6. B erfüllt die verallgemeinerte Suchbaumeigenschaft.

Einfügen in B-Bäumen

Beim Einfügen ermittelt man zunächst das Blatt, in das der neue Schlüssel s einzutragen ist. Befinden sich weniger als 2m Schlüssel in diesem Blatt, so füge man den neuen Schlüssel s sortiert ein.

Anderenfalls liegt ein Überlauf im Knoten vor und das Blatt muss in zwei Blätter mit je m Schlüsseln aufgespalten werden, wobei der mittelste der 2m+1 Schlüssel an den Elternknoten weitergereicht und dort eingetragen wird. Eventuell muss nun auch der Eltern-knoten aufgespalten werden usw. Hierbei kann von unten nach oben ein Aufspalten bis zur Wurzel hin erfolgen.

Wird die Wurzel aufgespalten, so wird eine neue Wurzel mit genau einem Schlüssel erzeugt. Der Aufwand ist wie bei der Suche, jedoch muss ggf. der Pfad bis zur Wurzel zurückverfolgt werden. Hinzu kommt das Einsortieren der Schlüssel in die Knoten.

Aufgabe 5: (AVL Bäume)

a) Geben Sie den AVL-Baum an, der durch Einfügen der Schlüssel 1; 2; 1; 9; 12; 11; 9; 10; 3; 4; 5 in einen leeren Baum entsteht.

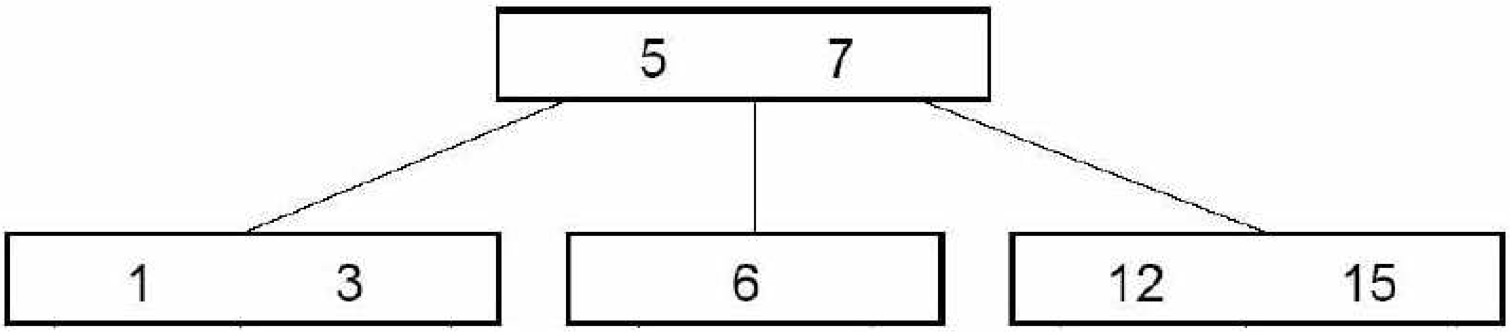
b) Gibt es zu jedem AVL-Baum T eine Einfügereihenfolge der Schlüssel, die zu diesem Baum führt, **ohne** dass Rotationen stattfinden? Falls ja, geben Sie einen Beweis an, andernfalls ein Gegenbeispiel.

Aufgabe 6: (AVL Bäume)

Fügen Sie die Monatsnamen nacheinander in einen lexikographisch geordneten, anfangs leeren AVL-Baum ein. Wenn immer eine Rebalancierung notwendig ist, zeichnen Sie den Baum.

Aufgabe 7 (Einfügen eines Knotens in einen B-Baum)

Gegeben ist folgender B-Baum der Ordnung 1.



Fügen Sie den Schlüssel 14 ein, und strukturieren Sie den Baum ggf. um. Zeichnen und erklären Sie die einzelnen Schritte des Einfügens.

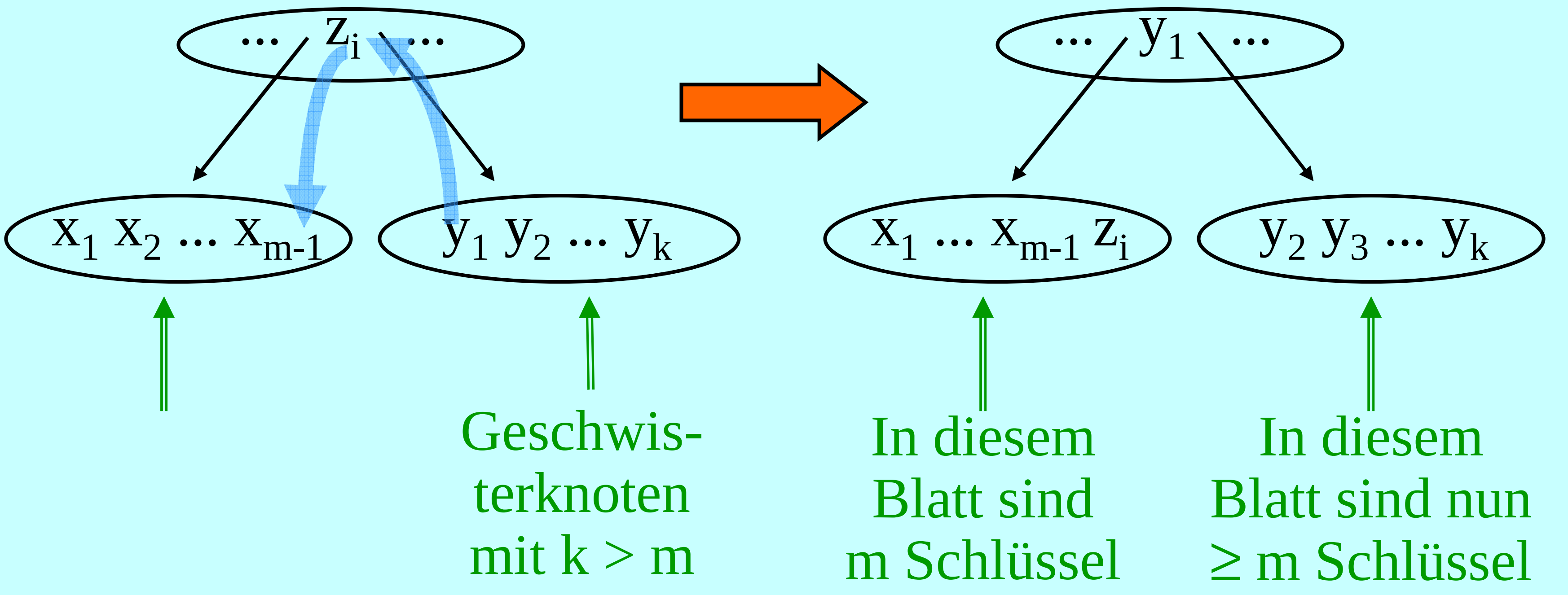
Löschen in B-Bäumen

Beim Löschen wird zunächst der Knoten, in dem der gesuchte Schlüssel s steht, ermittelt. Ist dieser Knoten kein Blatt, so wird der Schlüssel s durch seinen Inorder-Nachfolger s' ersetzt (Erinnerung: wie findet man diesen?). Da der Inorder-Nachfolger in einem Blatt steht, muss nun dieser Schlüssel s' im Blatt gelöscht werden.

Fall 1: Besitzt das Blatt mindestens m+1 Schlüssel, so wird der Schlüssel s' gelöscht und man ist fertig.
Fall 2: Besitzt das Blatt genau m Schlüssel "es liegt hier ein "Unterlauf" vor), so prüft man, ob ein Geschwisterknoten mit mindestens m+1 Schlüsseln existiert. Ist dies der Fall, so führt man einen Ausgleich unter Verwendung des zugehörigen Schlüssels des Elternknotens durch. Hierbei genügt eine Verschiebung von 2 Schlüsseln.

Löschen in B-Bäumen (2 Teil)

Genauer: Das Blatt besitzt m Schlüssel und mindestens ein Geschwisterknoten besitzt mindestens m+1 Schlüssel. Dann wird der zugehörige Schüssel des Elternknotens in das Blatt geschoben und an seine Stelle wird der nächstgelegene Schlüssel aus dem Geschwisterknoten gesetzt. Skizze:



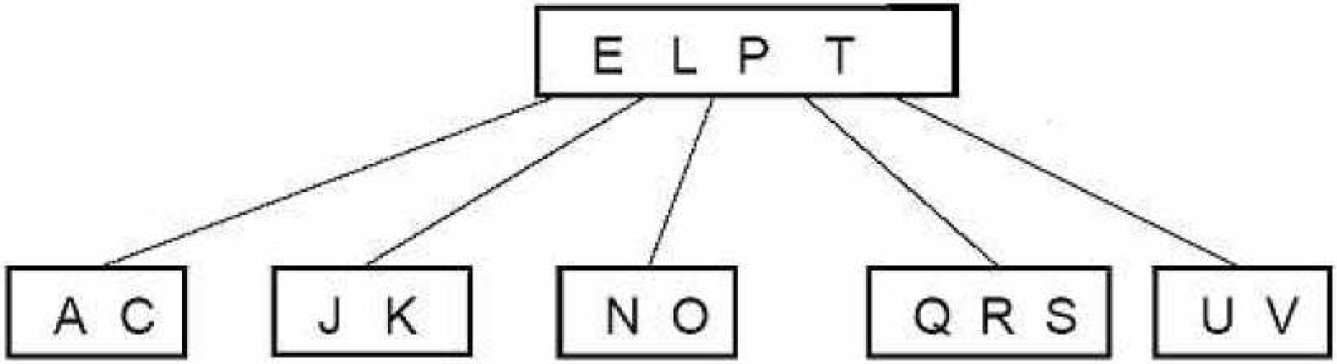
Aufgabe 8 (Aufbau von B-Bäumen)
a) Fügen Sie die Folge 1; 2; 3; 4; 15; 14; 7; 8; 9; 12; 11 in dieser Reihenfolge in einen anfangs leeren B-Baum der Ordnung 2ein. Zeichnen Sie den B-Baum jeweils nach dem Einfügen eines Elements.

b) Zeichnen Sie einen B-Baum der Ordnung 1, in den die Folge

1; 3; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16

in aufsteigend - sortierter Reihenfolge eingefügt wird. Was beobachten Sie beim Einfügen?

Aufgabe 9 (Löschen in B-Bäumen)
Gegeben ist folgender B-Baum



a) Welche Ordnung hat der Baum? Was ist die minimale und maximale Schlüsselzahl bei dieser Ordnung?

b) Zeigen Sie die Ergebnisse der Delete - Operationen für die Schlüssel C, P und V (in dieser Reihenfolge).

Aufgabe 10 (Höhe in B-Bäumen)

Bei einem B-Baum vom Grad *m* haben die Knoten zwischen *m* und *2m* Elemente; die Wurzel hat zwischen 1 und *2m* Elemente. Alle Blätter haben gleiche Tiefe.
Zeigen Sie, dass in einem B-Baum vom Grad *m* bei einer Zahl von *n* Knoten die Höhe *h* immer begrenzt ist durch

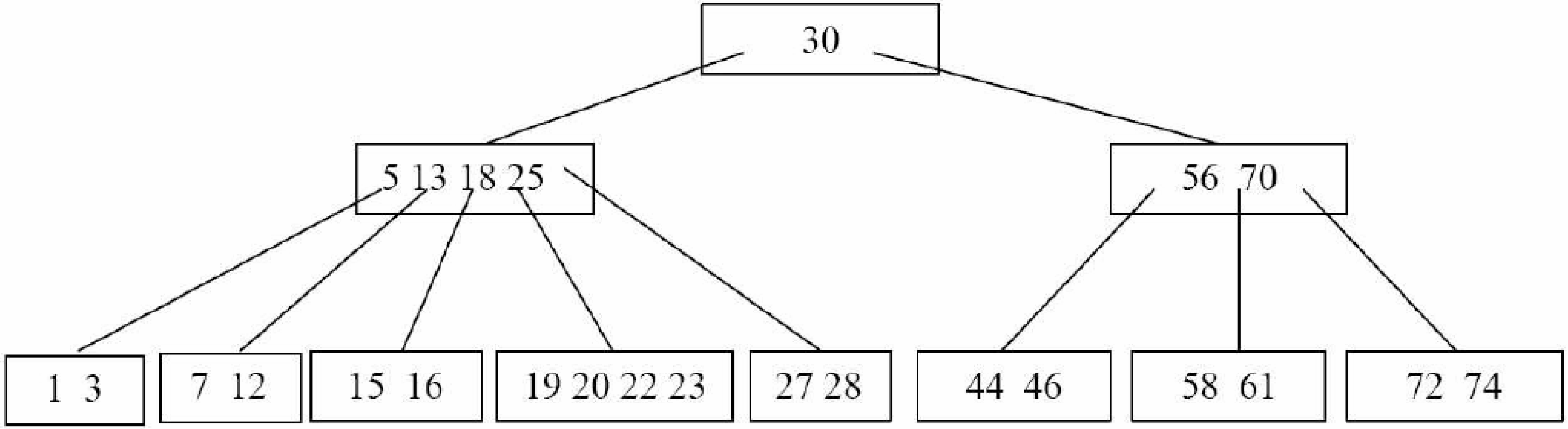
$$h \leq 1 + \log_{(m+1)}\left(\frac{n+1}{2}\right)$$

Berechnen Sie damit die maximale Höhe eines B-Baumes vom Grad 128 für eine Million bzw. eine Milliarde Knoten.

Aufgabe 11 (Eigenschaften von B-Bäumen)

Gegeben ist folgende Schlüsselmenge: { 1, 3, 5, 6, 7, 12, 15 }
a) Welche minimale Höhe hat ein B-Baum der Ordnung m=1, der diese Schlüssel speichert?
b) Gibt es eine größere Schlüsselmenge, für die die minimale B- Baum -Höhe denselben Wert wie unter a) hat?
c) Welche maximale Höhe hat ein B - Baum der Ordnung m=1, der diese Schlüssel speichert?

Aufgabe 12 (Operationen auf B-Bäumen)



Führen Sie die folgenden Operationen jeweils auf dem gegebenen B-Baum der Ordnung m = 2 durch.

a) Einfügen von 24

b) Löschen von 44

Aufgabe 13 (Aufbau von B-Bäumen)

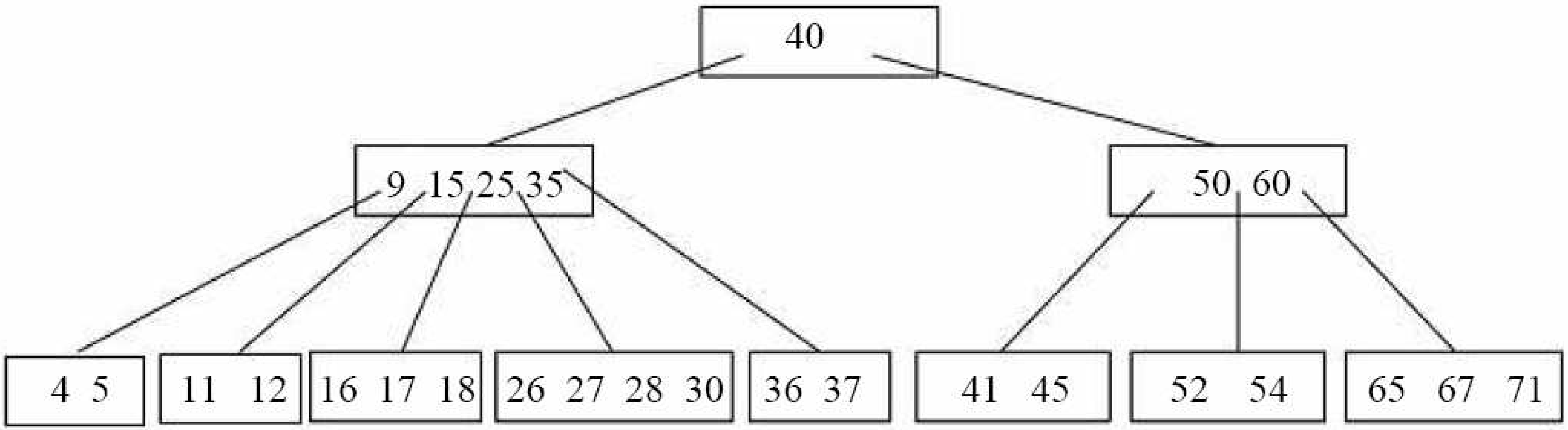
a) Konstruieren Sie einen (zu Beginn leeren) B-Baum der Ordnung 2 durch Einfügen der folgenden Schlüssel:

48, 23, 35, 66, 7, 3, 71, 12, 55, 2, 1, 9, 10, 25, 39, 42, 91, 84, 74.

b) Löschen Sie in dem entstandenen B-Baum folgende Schlüssel:

35, 9, 55, 84, 91, 23, 12 , 7, 10, 3, 25, 39, 1 ,2,71.

Aufgabe 14: (B-Bäume – ehemalige Klausuraufgabe)



Welche Ordnung hat der gegebene B-Baum? Führen Sie die Operationen ‚Einfügen von 29' und ‚Löschen von 41' durch (9P).