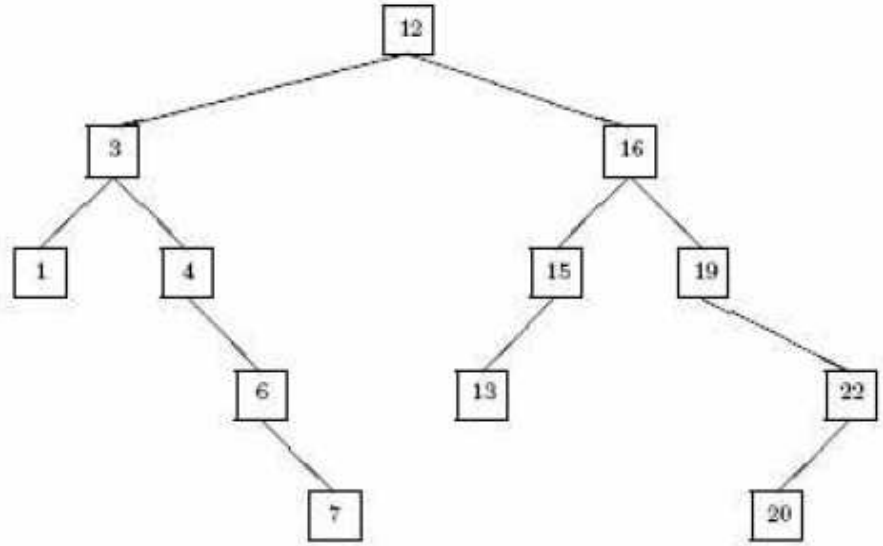
Aufgabe 1 (Interpolationssuche)

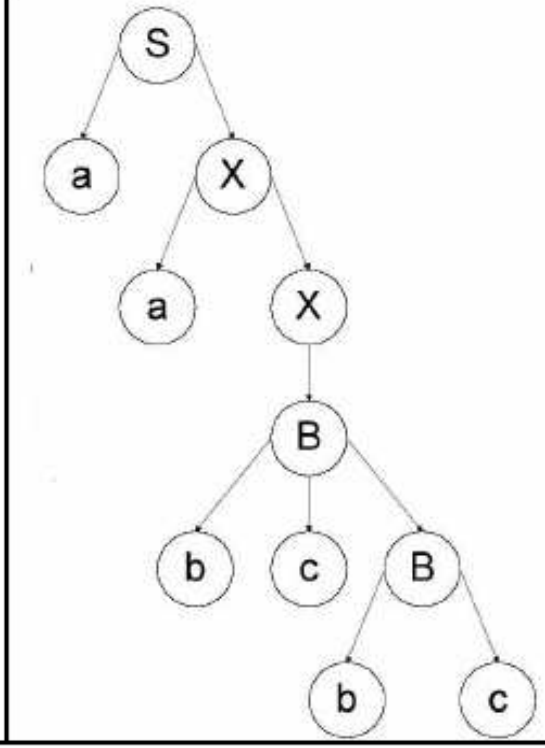
Gegeben sei eine geordnete Liste (2, 3, 5, 6, 8, 9, 12, 14, 22, 23, 24, 25, 29, 31, 34, 35, 37, 40, 44, 46, 52, 53, 58, 59, 62, 84, 85, 86, 88, 89).
Suchen Sie (von Hand) die Elemente 8, 28 und 62 mit Interpolationssuche. Geben Sie in jedem Schritt den berechneten Index und das zughörige Element an. Wie viele Schritte sind jeweils nötig?

Aufgabe 2: (Wiederholung Suchbäume)
Geben Sie von dem nebenstehenden binären Suchbaum an, welche Werte jeweils ausgegeben werden, wenn dieser nach den Strategien Preorder, Inorder, Postorder oder Levelorder traversiert wird.
Bauen Sie einen anfänglich leeren binären Suchbaum auf durch sukzessives Einfügen der Werte 12, 3, 4, 18, 20, 1, 7, 6, 19. Löschen Sie danach die Knoten 12 und 3. Erläutern Sie Ihr Vorgehen.



Aufgabe 3: (Binäre Suchbäume)
Fügen Sie die Schlüssel 4, 10, 9, 2, 5, 3, 1, 7, 8, 6 in dieser Reihenfolge in einen anfangs leeren Suchbaum ein. Zeichnen Sie den fertigen Suchbaum!
Löschen Sie jetzt den Knoten 4. Beschreiben Sie Ihr Vorgehen.

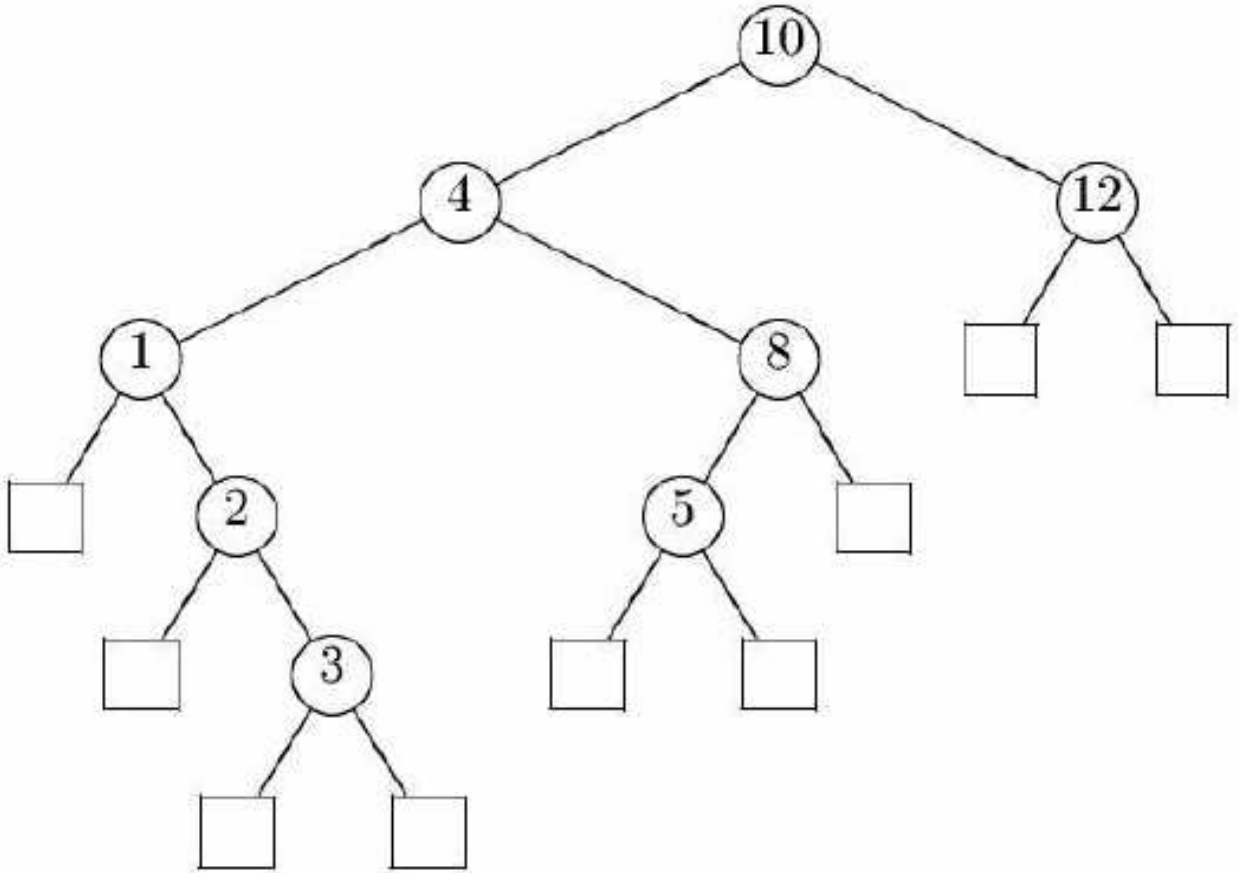
Aufgabe 4: (Binarisierung)
Binarisieren Sie folgenden Baum:



Aufgabe 5: (Baumdurchläufe)
Von den Suchbäumen B_i ($i=1,2,3$) sei folgendes bekannt. Der Preorderdurchlauf von B_1 ist (3,1,2,5,4,6,7), der Levelorderdurchlauf von B_2 ist (6,3,7,4,9,2,5,8,1), der Postorderdurchlauf von B_3 ist (1,3,2,4,5,8,7,6). Stellen Sie B_i ($i=1,2,3$) graphisch dar.

Aufgabe 6: (Suchbäume)

Es sei der folgende Suchbaum t gegeben.



a) Geben Sie für diesen Baum die interne Pfadlänge an?

b) Was ist die durchschnittliche interne Pfadlänge von t?

c) Führen Sie in dem Binärsuchbaum der Reihe nach die folgenden Operationen durch: Einfügen(11), Einfügen(6), Entfernen(12), Entfernen(4).

Innere Pfadlänge

Zu jedem Baum mit Wurzel w gehört die Levelfunktion

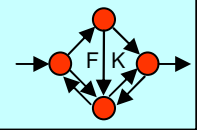
level: V $\rightarrow$ IN0 , rekursiv definiert durch

level(w) = 1 und
level(x) = level(vater(x)) + 1 für x $\neq$ w.

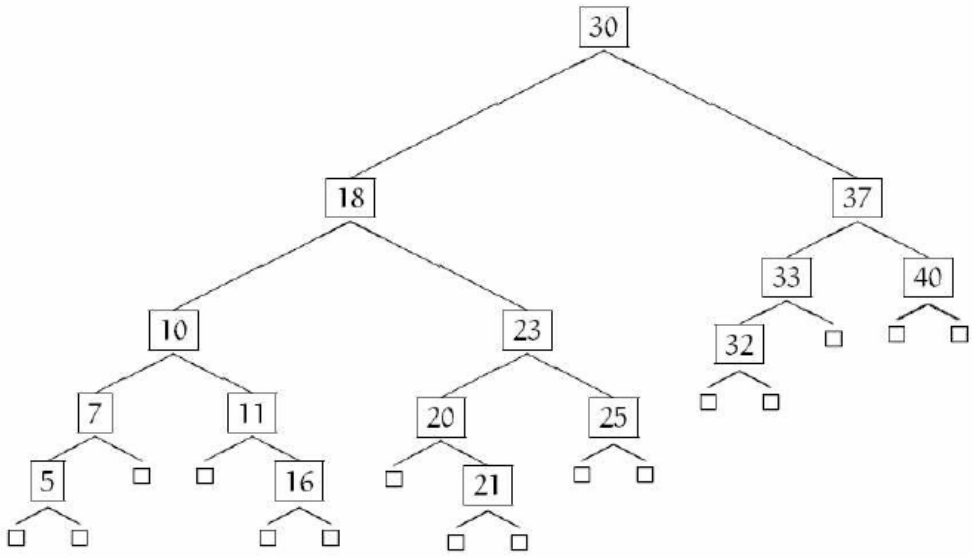
Definition 3.2.2.13: Die innere Pfadlänge Pf eines beliebigen gerichteten Baumes mit n Knoten $v_1, ..., v_n$ ist definiert als:

$$Pf(B) = \sum_{i=1}^{n} level(v_i) .$$

Die mittlere Suchzeit MS eines gegebenen Baumes B mit n Knoten ist dann MS(B) = Pf(B)/n.



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.



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.