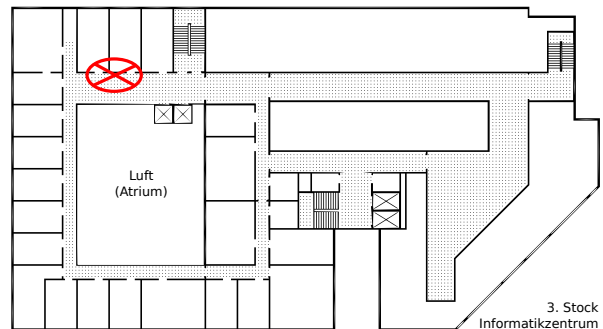


Hausaufgabenblatt 4

Abgabe der Lösungen bis zum Donnerstag, den 15.01.2026 um 11:00 Uhr im Hausaufgabenschrank bei Raum IZ 337 (siehe Skizze rechts). Es werden nur mit einem dokumentenechten Stift (kein Rot!) geschriebene Lösungen gewertet.

Schreibe auf die Abgabe unbedingt deinen Namen, Matrikel- und Gruppennummer! Mehrere Blätter tackern!



Hausaufgabe 1 (Binäre Suchbäume):

(4+4 Punkte)

- a) Wir nennen einen Knoten im Suchbaum *Blatt*, wenn dieser keine Kinder besitzt. Ein Knoten mit Kindern ist ein *innerer Knoten*. Sei nun b die Anzahl der Blätter in einem binären Suchbaum und i die Anzahl an inneren Knoten.

Zeige oder widerlege: In einem *vollen* (nicht zwingend vollständigen!) binären Suchbaum mit mindestens einem Knoten gilt $i = b - 1$.

- b) Betrachte einen binären Suchbaum T , in welchem kein Element doppelt vorhanden ist. Angenommen, die Suche nach einem Element mit Wert k in T endet in einem Blatt. Dieser Suchpfad von der Wurzel zum Blatt teilt die Elemente von T in die zwei Mengen A_L und A_R , die links bzw. rechts vom Suchpfad liegen. Wir nehmen an, dass $A_L \neq \emptyset$ und $A_R \neq \emptyset$ gelten.

Zeige oder widerlege: Für alle $l \in A_L, r \in A_R$ gilt: $l < r$.

Hausaufgabe 2 (Einfach verkettete Listen):

(3+2 Punkte)

Aus der Vorlesung kennen wir bereits die Operation LIST-DELETE für doppelt verkettete Listen. Wir wollen diese Operation nun für einfach verkettete Listen betrachten, bei denen jedes Element lediglich seinen Nachfolger kennt:

- a) Gib in Pseudocode eine Funktion an, die LIST-DELETE, wie sie aus Vorlesung und Skript bekannt ist, für einfach verkettete Listen realisiert.
- b) Ist die Funktion LIST-DELETE für einfach verkettete Listen mit Laufzeit $\mathcal{O}(1)$ realisierbar? Begründe!

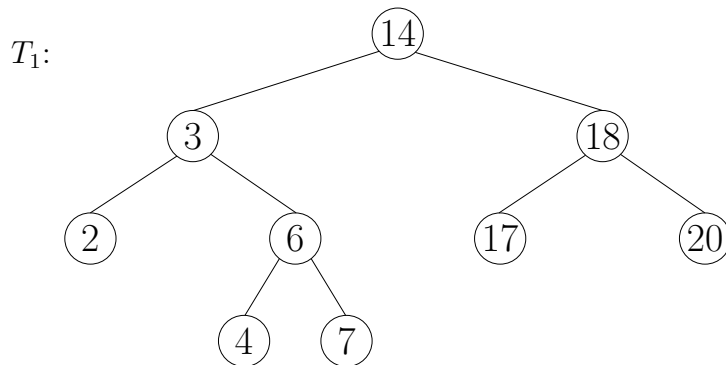
Hausaufgabe 3 (AVL-Bäume):

(2+3+2 Punkte)

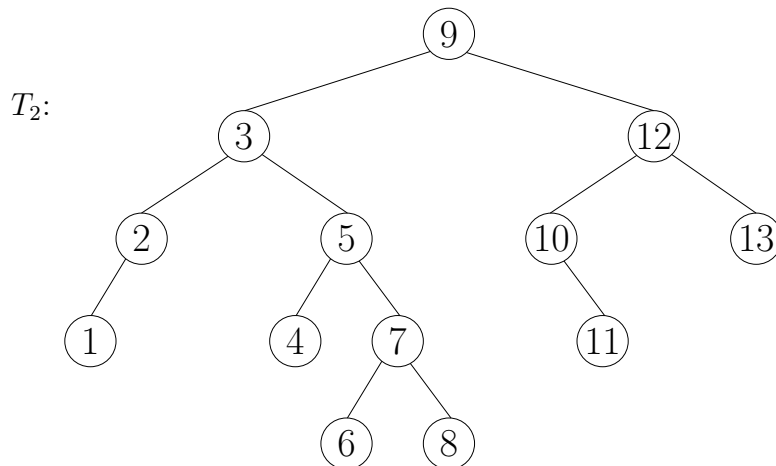
Betrachte in den Aufgabenteilen a) und b) den Baum, der in der jeweiligen Abbildung dargestellt ist. Führe die Operation des jeweiligen Aufgabenteils und die damit verbundenen Restrukturierungsmaßnahmen zum Erhalt der AVL-Eigenschaft auf dem entsprechenden Baum aus. Zeichne dabei das Resultat nach jeder einzelnen ausgeführten Operation INSERT, DELETE und RESTRUCTURE in einen separaten Baum.

(Hinweis: AVL-Bäume werden in den Vorlesungen 15 und 16 sowie in Übung 4 behandelt.)

a) INSERT($T_1, 5$)



b) DELETE($T_2, 13$)



c) Für welche Werte von $x \in \{1, 2, 4, 6, 8, 9, 11\}$ führt die Operation INSERT(T_3, x) zu einer Restrukturierung des Baumes? (Ohne Begründung.)

