

Bäume (21 Punkte)

Arithmetische Ausdrücke lassen sich auch als Bäume darstellen. So kann der Ausdruck:

$$7 * 2 + (5 + 4) * 3$$

durch den Baum in Abbildung 1 repräsentiert werden.

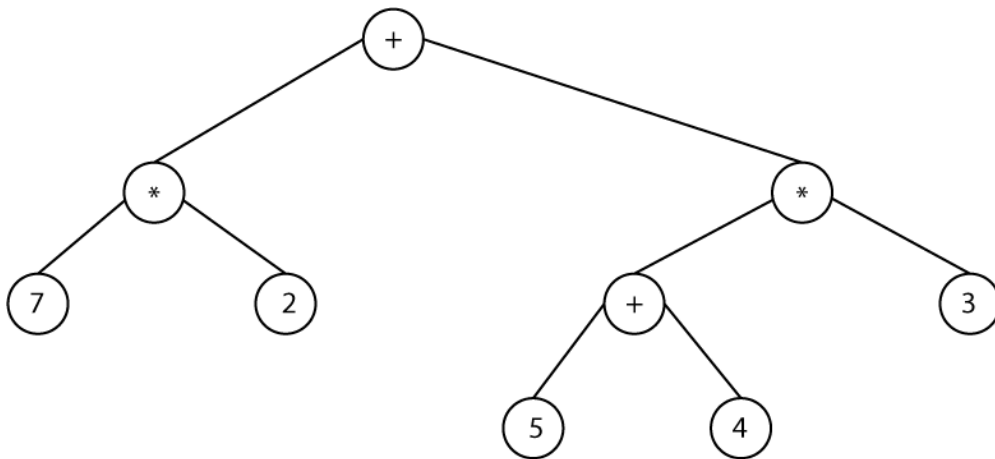


Abbildung 1: Darstellung eines arithmetischen Ausdrucks in Form eines Baums

Dabei speichert jeder Knoten entweder ein Operanden oder einen Operator. Die Blattknoten enthalten dabei die Operanden.

Klammern werden nicht explizit im Baum gespeichert, da ihre Bedeutung durch die Struktur des Baumes repräsentiert wird.

Die Wurzel enthält dabei immer den zuletzt angewendeten Operator.

Gegeben sei der folgende arithmetische Ausdruck:

$$9 + (3 * 5 - 4) + (7 * 2 / (8 * 4 - 2) + 4) * 5$$

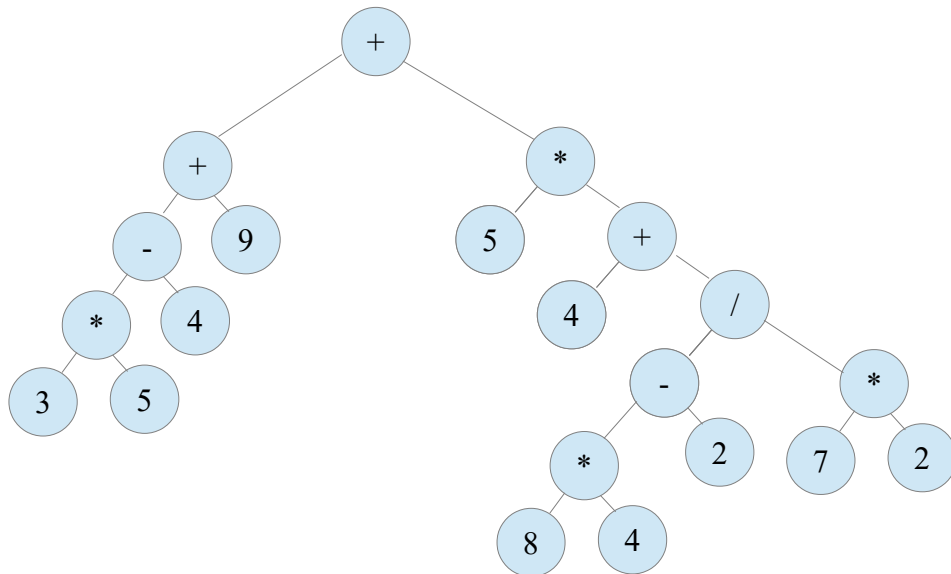
Erstellen Sie hierzu nun einen binären Baum wie in Abbildung 1.

Es soll dabei die Java-Auswertungsreihenfolge verwendet werden.

Meine eingereichte Lösung:

Wie immer: **keine Garantie** auf Richtigkeit oder Vollständigkeit.

$$9 + (3 * 5 - 4) + (7 * 2 / (8 * 4 - 2) + 4) * 5$$



a) Wie viele Blätter hat der Baum? (2 Punkte)

11

b) Welche Höhe besitzt der Baum? (3 Punkte)

7

c) Durchlaufen Sie den Baum jeweils mit dem angegebenen Verfahren und geben Sie den Inhalt der Knoten (ohne Leerzeichen, Komma und Klammern, also **nur Operatoren und Operanden**) an: (16 Punkte)

in-order: (links / aktuell / rechts)

3*5-4+9+5*4+8*4-2/7*2

pre-order: (aktuell / links / rechts)

(zuerst den **aktuellen** Knoten, anschließend den **linken** Nachfolger, danach den **rechten** Nachfolger)

++-*3549*5+4/-*842*72

post-order: (links / rechts / aktuell)

35*4-9+5484*2-72*/+*+

level-order: (Geschwister einer Ebene / von links nach rechts / nächste Ebene)

++*-95+*44/35-*27284