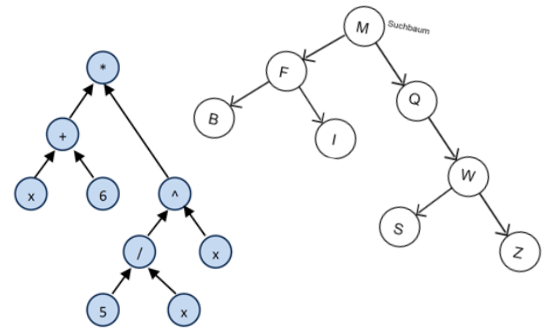


Bäume (in der Informatik)



Bäume gehören in der Informatik zu den wichtigsten Datenstrukturen. Der Baum wird sehr häufig für alle Arten von Problemen verwendet, wie zum Beispiel für die Suche, das Verwalten von Datenmengen oder für die Speicherung einer geometrischen Struktur. Die Datenstruktur an sich ist sehr simpel, gibt dafür einen Raum für sehr viele Variationen und verschiedene Implementierungen. Die folgenden Definitionen sind allerdings zunächst theoretischer Natur und von der konkreten Umsetzung in eine Programmiersprache unabhängig:

Definitionen

1. iterativ

Ein **Baum** bezeichnet in der Informatik einen speziellen gerichteten Graphen, der genau einen Knoten ohne direkten Vorgänger hat (die **Wurzel**) und in dem jeder **Knoten** von der Wurzel aus nur durch einen einzigen **Pfad** erreicht werden kann. Der direkte Vorgänger eines Knotens heißt **Vater**, ein direkter Nachfolger **Sohn**. Knoten ohne Söhne werden auch **Blätter** genannt.

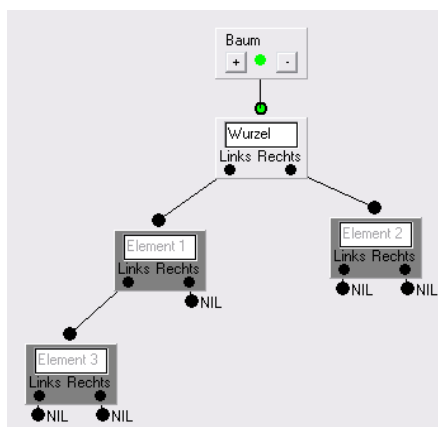
(Speziell: Ein Baum wird **Binärbaum** genannt, wenn jeder Knoten höchstens 2 Söhne hat.)

2. rekursiv

Ein Baum T ist entweder leer oder besteht aus einer Wurzel w mit n Teilbäumen t_1, t_2, \dots, t_n ($n \in \mathbb{N}$).

Dynamische Datenstruktur

Graphisch mögliche Darstellungen von Bäumen finden sich z.B. oben rechts auf diesem Blatt wieder. In Richtung Implementierung allerdings wollen wir uns von nun an an folgender Struktur orientieren, die sich an die Veranschaulichung der bekannten Listenstrukturen anlehnt:



```
public class BinaryTree<ContentType> {
    private class BTNode<CT> {
        private CT content;
        private BinaryTree<CT> left, right;
        ...
    }
    private BTNode<ContentType> node;
    ...
}
```