Wurzelbäume

Da ein Baum nie leer sein darf, können wir darin stets einen fixen Knoten als *Wurzelknoten* wählen. Enfernen wir den Wurzelknoten (zusammen mit allen anliegenden Kanten) so bleibt ein Wald übrig, dessen Zusammenhangskomponenten wieder Bäume sind. Für jede diese Zusammenhangskomponenten wählen wir als Wurzelknoten denjenigen, der mit dem entfernten Knoten verbunden war und fahren so rekursiv fort.

8.3.11 Definition. Ein Baum, in dem ein Knoten als Wurzelknoten ausgezeichnet ist, heißt ein *Wurzelbaum*

Jeder Wurzelbaum zerfällt auf eindeutige Weise in einen Wurzelknoten und eine Menge von Wurzelbäumen (Teilbäume).

Die Wurzelknoten der Teilbäume bezeichnet man auch als *Nachfolger* oder *Söhne* der Wurzel des Gesamtbaumes. Knoten, die keine Nachfolger haben, heißen *Blätter*.

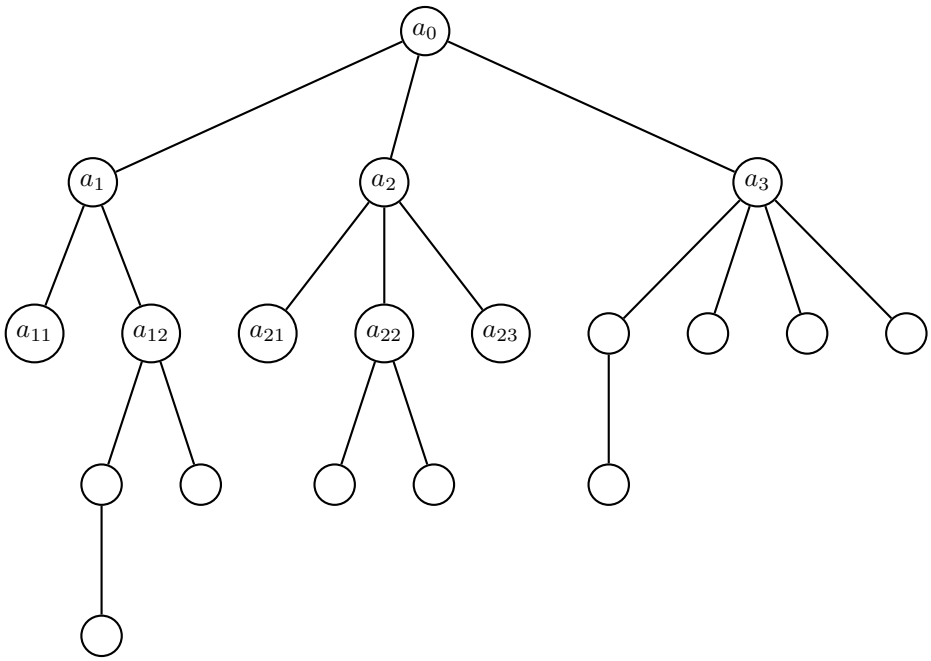

Abbildung 8.1: Rekursiver Aufbau eines Baumes

Die obige Konstruktion entspricht einem rekursivem Aufbau: Ein Baum besteht aus einem Wurzelknoten und einer (möglicherweise leeren) Menge weiterer Bäume (den *Teilbäumen*). Damit kann man Bäume als hierarchische Struktur und als Verallgemeinerung von Listen auffassen; Listen entsprechen dann genau denjenige Bäumen, in denen jeder Knoten höchstens einen Nachfolger hat.

Bäume treten immer dann auf, wenn eine Struktur aus gleichartigen Unterstrukturen besteht, bzw. wenn sich ein Problem in Teilprobleme zerlegen läßt. Typische Beispiele: Sortier- und Suchprobleme.

Binärbäume

8.3.12 Definition. Ein *binärer Baum* ist entweder leer, oder er besteht aus einem Wurzelknoten und einem linken und einem rechten binären (Teil)baum.

Formal könnte man diese Definition etwa so schreiben:

$$\mathrm{BTree}\, A = \{\, \varnothing \,\} + \mathrm{Btree}\, A \times A \times \mathrm{Btree}\, A.$$

Dies macht die Analogie zur Definition von Listen noch deutlicher (bei letzteren fehlt lediglich der zweite Teilbaum).

Prinzipiell sind binäre Bäume einfach Bäume, bei denen jeder Knoten höchstens zwei Nachfolger hat. Aber nicht genau: Im Unterschied zu Bäumen wird bei binären Bäumen zwischen dem linken und dem rechten Nachfolger unterschieden. Insbesondere sind

und

als verschiedene binäre Bäume zu betrachten. Außerdem kann ein Binärbaum, so wie ein Wald, aber im Gegensatz zu einem Baum, leer sein. Tatsächlich kann man jedem binären Baum auf natürliche Weise einen Wald zuordnen und umgekehrt:

- Interpretiert man den linken Nachfolger in einem binären Baum als „ältester Sohn“ und den linken Nachfolger als „nachfolgender Bruder“, so bekommt jeder Knoten eine entsprechende Menge Söhne, wodurch ein Wald entsteht.
- Zeichnet man umgekehrt in einem Wald nur zu einem Sohn eine Kante, und dafür von jedem Sohn eine zum nächsten, so entsteht ein binärer Baum.

Dies heißt allerdings nicht, daß Bäume und binäre Bäume damit zwei Varianten desselben Konzeptes wären. Denn ein Baum besteht ja aus einem Knoten und einer *Menge* von Teilbäumen. Die Interpretation als binärer Baum legt aber eine Reihenfolge der Teilbäume fest (nächstälterer Bruder). Insbesondere sind die Bäume

und

als gleich zu betrachten, führen aber mit der obigen Zuordnung zu den unterschiedlichen binären Bäumen:

und

Entsprechend dem rekursiven Aufbau eines binären Baumes, kann man eine Rekursion oder Induktion ähnlich wie für Listen führen, mit dem Unterschied, daß eben beide Zweige berücksichtigt werden müssen.

Zum Beispiel erhalten wir analog zur Länge von Listen die Anzahl der Knoten in einem binären Baum:

- Der leere Baum hat 0 Knoten;
- Die Anzahl der Knoten in einem nicht-leeren binären Baum ist um 1 höher als die Summe der Anzahl der Knoten in den beiden Teilbäumen.

Ersetzt man in dieser Definition „Summe“ durch „Maximum“, so erhält man die *Höhe* des binären Baumes.

8.3.13 Definition. Ein binärer Baum ist *sortiert* falls er entweder leer ist oder wenn beide Teilbäume sortiert sind und alle Elemente im linken Teilbaum kleiner als der Wurzelknoten sowie alle Elemente im rechten Teilbaum größer als der Wurzelknoten sind.

Es ist leicht, eine Funktion tree2list zu schreiben, welche zu einem sortierten binären Baum die sortierte Liste derselben Elemente bestimmt:

- tree2list von einem leeren Baum ergibt die leere Liste;
- ansonsten bildet man tree2list vom linken Teilbaum, hängt den Wurzelknoten an, und dann tree2list vom rechten Teilbaum.

8.3.14 Definition. Diese Vorgangsweise, daß man zuerst den linken Teilbaum behandelt, dann den Wurzelknoten, dann den rechten Teilbaum, nennt man ganz allgemein *in-order*. Analog gibt es *pre-order* (zuerst der Wurzelknoten, dann die Teilbäume) und *post-order* (zuerst die Teilbäume, dann der Wurzelknoten) sowie *double-order* (zuerst der Wurzelknoten, dann die beiden Teilbäume, dann nocheinmal der Wurzelknoten).

Das Berechnen von arithmetischen Ausdrücken (lassen sich in recht natürlicher Weise als Bäume darstellen) ist ein typisches Beispiel, bei dem post-order gefragt ist. Beim Suchen nach einem bestimmten Element in einem binären Baum (z.B. nach einer Datei mit einer bestimmten Eigenschaft) dagegen verwendet man pre-order. Hierarchisch organisierte Arbeitsabläufe funktionieren üblicherweise nach double-order: Zuerst delegiert der Chef Teilprobleme an die direkt unterstellten Mitarbeiter (und diese delegieren weiter) und später, wenn diese mit Ihrer Arbeit fertig sein, muß alles noch irgendwie kombiniert werden.