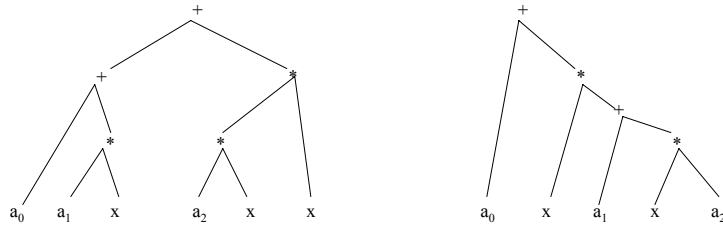


**Lösungen zu Übungsblatt 2: Datenstrukturen, WS 2001/02, Prof. R. Bayer**

**Lösung 3.1:**  $a_0 + a_1x + a_2x^2 = a_0 + x(a_1 + xa_2)$



Der linke Baum ist niedriger und buschiger, kann deshalb in 3 Schritten in kürzerer Zeit ausgeführt werden, braucht dafür aber 2 Prozessoren, die parallel arbeiten können, und insgesamt 5 Operationen.

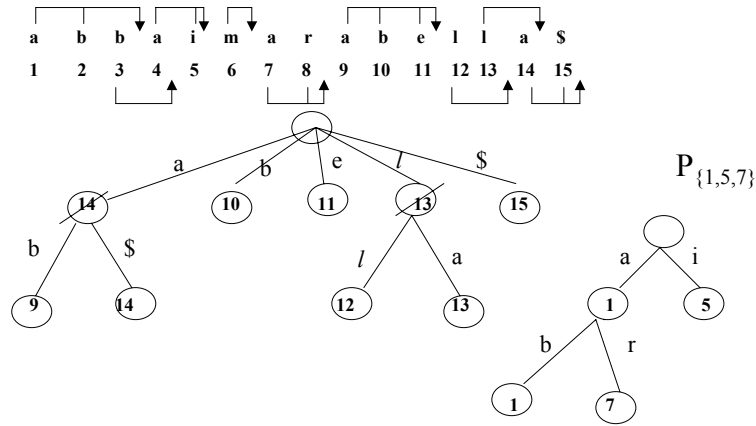
Der rechte Baum ist höher und schlanker, braucht deshalb 4 Berechnungsschritte für 1 Prozessor, ist insgesamt also billiger, dauert aber länger.

**Lösung 3.2:**

**Quicksort:** Die Kaskadenrekursion führt meist zu einem buschigen Baum, der viel Parallelität erlaubt. Jede Wahl eines Pivot-Elementes führt zu 2 Folgeprozessen.

**Mergesort:** Paare von schon sortierten Teilfolgen können in parallelen Prozessen zusammengemischt werden.

Lösung 4.1 im folgenden ist nur ein Teilbaum gezeichnet, der beim rechts-links Verfahren entsteht



### Lösung 4.2

$\log_{26} 10^8 \sim 6$  die Längen von Positionsidentifikatoren sollten im Bereich 6 bis ca 20 Zeichen liegen, Beispiele:

- mich dür...
- es ist vollbr...
- du sollst nicht t...
- vater unser ...
- das ist mein l...

### Lösung 4.3

Bei 3 Oktaven mit Halbtönen bekommt man 48 verschiedene Töne, nimmt man die Tonlängen hinzu mit ca 10 verschiedenen Längen, so bekommt man ein sehrgroßes Alphabet von ca 480 Zeichen für Melodien. Die Länge der Positionsidentifikatoren wird dann  $\log_{480}$  was natürlich sehr klein wird.