

---

## Effiziente Algorithmen und Datenstrukturen I

---

Abgabetermin: 19.12.2003 vor der Vorlesung

### Aufgabe 1

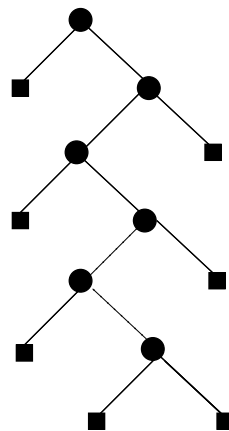
Seien  $T$  und  $T'$  zwei unterschiedliche binäre Suchbäume auf den gleichen  $n$  Schlüsseln. Zeigen Sie, dass man mit  $O(n)$  Rotationen  $T$  in  $T'$  umwandeln kann.

### Aufgabe 2

Seien  $U$  und  $T$  zwei  $(2, 4)$ -Bäume, die  $m$  bzw.  $n$  Schlüssel speichern. Die Schlüssel im Baum  $T$  seien alle höchstens so groß wie die Schlüssel im Baum  $U$ . Zeigen Sie, wie man mit einem Zeitaufwand von  $O(\log n + \log m)$  die beiden Bäume  $T$  und  $U$  zu einem  $(2, 4)$ -Baum  $Z$  vereinen kann, der alle  $n + m$  Schlüssel enthält.

### Aufgabe 3

Sei  $T$  ein Splay-Tree auf  $n$  Knoten,  $n$  ungerade. Geben Sie eine Sequenz von Zugriffen auf  $T$  an, die  $T$  durch die entsprechenden Splayoperationen in einen Baum umwandelt, der die im Bild angegebene „Form“ hat. (Der Pfad der inneren Knoten alterniert zwischen rechten und linken Kinder.)



### Aufgabe 4

Gegeben sei ein  $n \times n$ -Array  $A$  mit Einträgen aus  $\{0, 1\}$  so, dass in einer Zeile die 1-Einträge immer vor den 0-Einträgen stehen. Beschreiben Sie ein Verfahren, das mit Zeitaufwand  $O(n \log n)$  die Anzahl der 1-Einträge zählt.