Datenstrukturen und Algorithmen — *Übung 8*
Frühlingssemester 2011 — *Abgabetermin 21. April 2011*
Institut für angewandte Mathematik — *Adrianus Kleemans [07-111-693]*

Theoretische Aufgaben: Rot-Schwarz-Bäume

Aufgabe 1

Da die Wurzel vorher rot war, müssen die beiden Unterknoten (falls vorhanden) schwarz sein. Die Eigenschaft 4 würde deshalb bei einer Schwarzfärbung nicht verletzt werden, weshalb der resultierende Baum ein gültiger Rot-Schwarz-Baum sein muss.

Aufgabe 2

Da jeder Pfad von einem Knoten zu seinem Blattknoten gleichviele schwarze Knoten haben muss, kann nur die Anzahl roter Knoten variieren. Beim kürzesten Pfad entspricht dies nur schwarzen Knoten, beim längsten gibt es gleichviele beider Farben. Da es höchstens gleichviel schwarze wie rote Knoten geben kann, ist auch die Länge des längsten Pfads höchstens doppelt.

Aufgabe 3

- **Minimale Anzahl:** Vom Buch: $2^h - 1$ innere Knoten sind das Minimum.
- **Maximale Anzahl:** Bei maximaler Knotenanzahl muss jeder Pfad die maximale Länge ($2 \cdot$ Schwarzhöhe) haben. Daraus ergibt sich eine Knotenanzahl von $2^{2 \cdot h} - 1$.

Aufgabe 4

Listing 1: Aufgabe 4

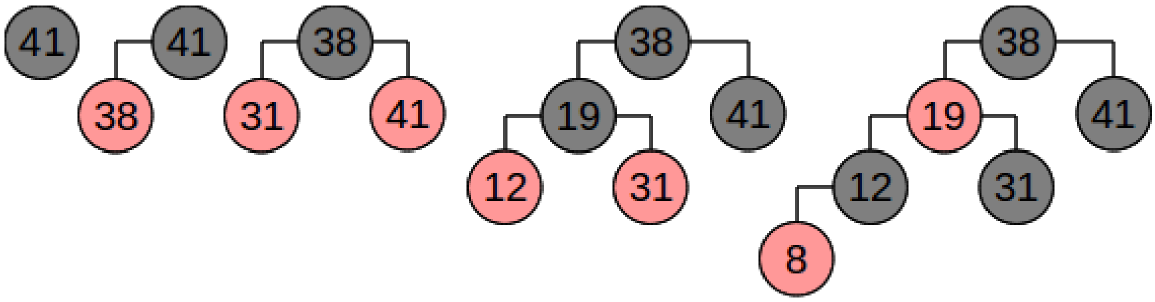
```
right-rotate(T, x)
y = x.links              // bestimme y
x.links = y.rechts        // mache y's rechter Teilbaum zu x's linkem

if y.rechts != T.nil
  y.rechts.vater = x
y.vater = x.vater         // setze y's Vater auf x's Vater
if x.vater == T.nil
  T.wurzel = y
else if x == x.vater.rechts
  x.vater.rechts = y
else x.vater.links = y
y.rechts = x              // mache x zu y's rechtem Kind
x.vater = y
```

Aufgabe 5

- Für jedes linke Kind kann die Operation right-rotate ausgeführt werden, solange, bis alle in den rechten äusseren Pfad (der Pfad, in dem alle Knoten ausser der Wurzel rechte Kinder sind) integriert wurden. Dies muss nicht für die Wurzel geschehen, deshalb reichen $n - 1$ Rechtsrotationen.
- Wenn man vom degenerierten Baum, der nur aus einem rechten Pfad (wie oben beschrieben) besteht, die Rechtsrotationen mit Linksrotationen wieder rückgängig macht, so erhält man wieder den ursprünglichen Baum. Da nun jeder Baum in die degenerierte Form umwandelbar ist, gilt auch das umgekehrte, aus dem degenerierten baum lässt sich in $n - 1$ Linksrotationen jeder gewünschte Baum bauen. Die Gesamtlaufzeit ist deshalb $2 \cdot (n - 1) = 2n - 2$, welches in $\Theta(n)$ liegt.

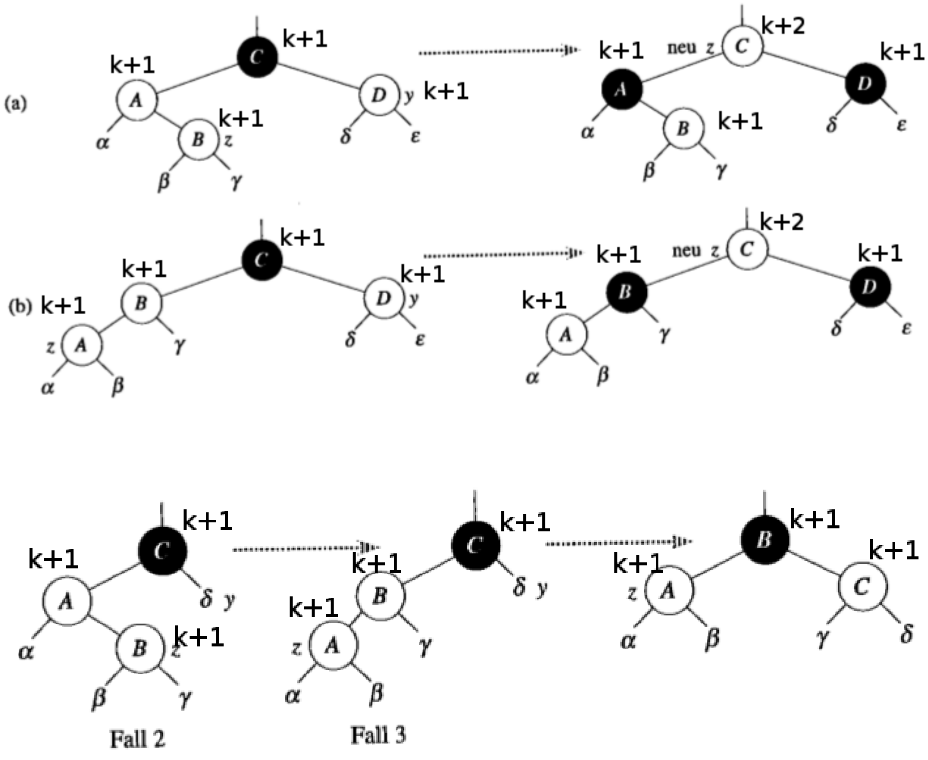
Aufgabe 6

Es werden gemäss der Problemstellung nur Rot-Schwarz-Bäume gezeichnet und keine temporär erstellten Zwischenbäume.



Aufgabe 7

Da die Wurzeln der Teilbäume auch schwarz sind, kommt bei den unteren Knoten als Schwarzhöhe zu k schon eins dazu. Die Transformation erhält die Eigenschaft 5. Über dem Knoten C oder allgemein dem



angezeigten Teilbaum ist die Höhe konstant k+2 geblieben.

Aufgabe 8

Nein, durch die verschiedenen Fälle kann es zu Umformungen oder zu Umfärbungen kommen. Der resultierende Baum muss nicht derselbe sein; ein right- oder left-rotate verformt ihn oder das Herstellen der Eigenschaft 4 kann einzelne Knoten verfärben.