6. Übung Informatik I

Marcus Rickert

30. Dezember 1995

Aufgabe 1

Siehe Listing **info6.pas**

Aufgabe 2

Für die Operationen *Links-Rotation, Rechts-Rotation und Rechts-Links-Rotation* wurden die Beschriftungen der Vorlesung übernommen. Für die 4. Operation *Links-Rechts-Rotation* kann die Beschriftung untenstehender Zeichnung entnommen werden. Die Parameter der Prozeduren sind jeweils Zeiger auf die die Beschriftung tragenden Knoten.

Links-Rotation

PROCEDURE links-rotation (v,w)
  BEGIN
  v→rechter_sohn:=w→linker_sohn
  w→linker_sohn:=v
  END

Rechts-Rotation

PROCEDURE rechts-rotation (v,w)
  BEGIN
  w→linker_sohn:=v→rechter_sohn
  v→rechter_sohn:=w
  END

Rechts-Links-Rotation

```
PROCEDURE rechts-links-rotation (x,v,w)
    BEGIN
    rechts-rotation(x,w)
    links-rotation(v,x)
    END
```

Links-Rechts-Rotation

```
PROCEDURE links-rechts-rotation (x,v,w)
    BEGIN
    links-rotation(v,x)
    rechts-rotation(x,w)
    END
```

Aufgabe 3

Gegeben sei ein binärer Baum. Es ist folgende Äquivalenz zu zeigen:

(A) Baum rang-balanciert $\iff$ (B) Baum 'rot-schwarz'-balanciert

$\underline{A \Longrightarrow B}$

Algorithmus für die 'Einfärbung' der Knoten

```
Für alle Blätter
    BEGIN
    Bis zur Wurzel
        BEGIN
        Merke Rang des Knotens (Sohn)
        Gehe eins nach oben (zum Vater)
        Wenn Kante schon gefärbt[1]
            Nächstes Blatt
        Wenn Rang des Vaters ungleich (größer) dem des Sohnes
            Färbe Kante schwarz
        Sonst
            Färbe Kante rot
        END
    END
```

zu (i)

Klar nach Algorithmus und A-(iii).

zu (ii)

Nach A-(ii) ist der Rang eines Knoten um mindestens 1 größer als der Rang des Enkels. Wegen der Monotonie A-(i) kann sich der Rang um höchstens zwei unterscheiden und die Sprünge des Ranges nach oben sind immer gleich +1 oder null. Das bedeutet aber, daß zwischen Knoten und Enkel höchstens ein Sprung von null liegen kann und damit wird nur höchstens eine Kante rot gefärbt.

zu (iii)

Da es laut Vorraussetzung eine eindeutige Numerierung für den rang-balancierten Baum gibt,wegen der Monotonie A-(i) und weil nach A-(iii) alle Blätter Rang null haben, muß für jeden Knoten die Anzahl der Einser-Sprünge zu allen Blätter gleich sein. Die Anzahl der Einser-Sprünge entspricht jedoch laut Algorithmus der Anzahl der schwarzen Kanten.

$\underline{A \Longleftarrow B}$

Algorithmus für die Verteilung des Ranges:

Für alle Blätter
    BEGIN
    Setze Rang des Blattes auf null
    Setze Zähler auf Null
    Bis zur Wurzel
        BEGIN
        Gehe eins nach oben (zum Vater)
        Wenn Kante zum Vater schwarz war
            Erhöhe Zähler um eins
        Wenn Vater schon numeriert[2]
            Gehe zum nächsten Blatt
        Sonst
            Setze Rang des Vaters auf den Wert des Zählers
        END
    END

[1] dann würde sie jetzt die gleiche Farbe erhalten, denn diese hängt nicht von der Vorgeschichte ab, sondern nur vom Rangunterschied Vater-Sohn

[2] dann würde der Vater den gleichen Rang erhalten, da laut B-(iii) die Anzahl der schwarzen Kanten zu allen Blättern dieses Vaters gleich sind

zu (i)

Klar nach Algorithmus, da der Zähler nur bei jedem Schritt um 1 oder 0 erhöht wird.

zu (ii)

Da wegen B-(ii) für jeden Knoten auf dem Pfad zum Großvater höchstens eine rote Kante liegen darf, muß mindestens eine schwarz sein. Damit wird der Zähler um mindestens eins erhöht und der Rang unterscheidet sich um mindestens eins.

zu (iii)

Klar nach Algorithmus und B-(i).