Prof. Andreas Goerdt
Frank Schädlich

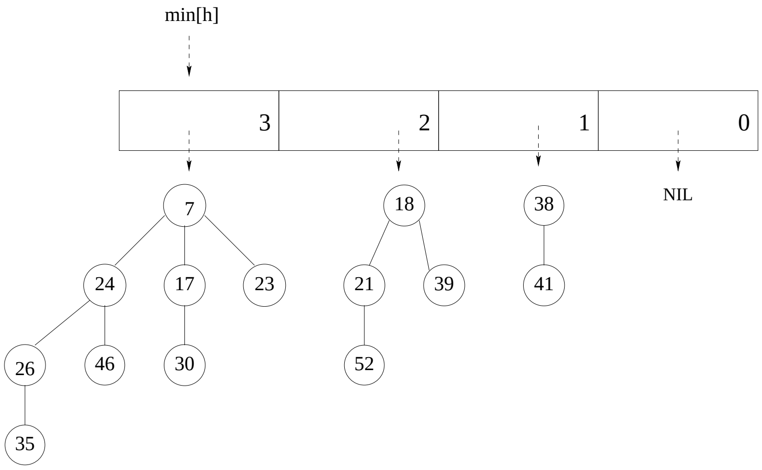
TU Chemnitz
Sommersemester 2003

Theoretische Informatik III

2. Übung

1. **Aufgabe**

   (a) Führen Sie DeleteMin(h) auf dem folgenden Binomialen Heap h aus:



   (b) Zeigen Sie, wie h aus dem leeren Heap entstehen kann.

2. **Aufgabe**

   Beweisen Sie folgenden Satz mit den Mitteln der Vorlesung:

   Eine Folge $\sigma = \{\sigma_1, \sigma_2, \cdots, \sigma_n\}$ von

   - n_1-mal $Insert$
   - n_2-mal $DeleteMin$
   - n_3-mal $Minimum$,

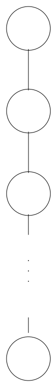
   in einer beliebigen Reihenfolge auf dem anfangs leeren binomialen Heap ausgeführt, besitzt einen Zeitaufwand $O(n_1 + n_2 \cdot \log n_1 + n_3) = O(n \log n)$ mit $n = n_1 + n_2 + n_3$.

3. **Aufgabe**

   Führen Sie eine amortisierte Analyse für die Addition von n Einsen in der Ternärdarstellung durch.

4. **Aufgabe**

(a) Geben Sie für jedes $n > 0$ eine Folge von Operationen für den Fibonacci-Heap an, so daß ein Baum der Art



mit n Knoten entsteht.

(b) Zeigen Sie, daß im `worst case` die Operationen DeleteMin(h), Cut(i, h), DecreaseKey(i, j, h) im Fibonacci-Heap h mit n Knoten ein Laufzeitverhalten von $\Omega(n)$ haben.

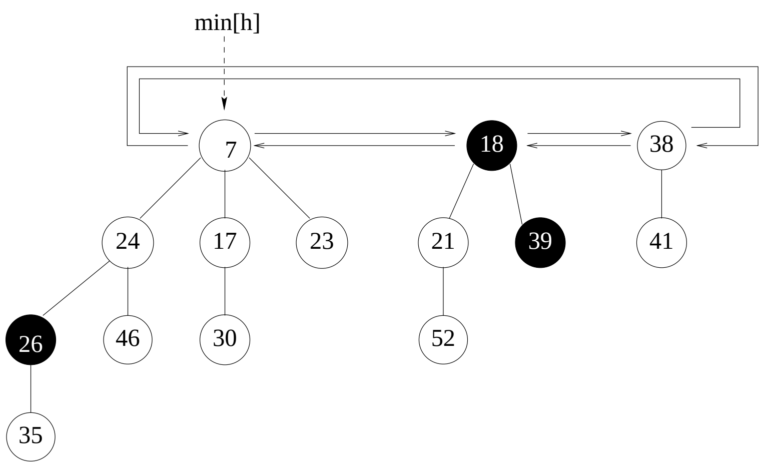
5. **Aufgabe**

Beweisen Sie formal die Richtigkeit der in der Vorlesung über die amortisierten Zeiten der Operationen auf dem Fibonacci-Heap getroffenen Aussagen. Verwenden Sie dabei die in der Vorlesung eingeführte Potentialfunktion:

$$\Phi(h) = \text{Anzahl der Bäume } + 2 \cdot \text{Anzahl der markierten Knoten}$$

6. **Aufgabe**

(a) Führen Sie DeleteMin(h) auf folgenden Fibonacci-Heap h aus:



(b) Zeigen Sie, wie h aus dem leeren Heap entstehen kann.