Prof. Andreas Goerdt
Frank Schädlich

TU Chemnitz
Sommersemester 2003

# Theoretische Informatik III

## 3. Übung

**1. Aufgabe**

Gegeben ist der Baum
$T = a \to b \to c \to d \to e \to f \to g$ .
Wir führen die Operationen `Find(`$a$`)`, `Find(`$b$`)`, $\cdots$, `Find(`$g$`)` in irgendeiner Reihenfolge mit Wegkompression auf $T$ aus. Die Zeit der Operation `Find(`$x$`)` sei die Anzahl der durchlaufenen Knoten.

(a) Welche Gestalt hat der Baum am Ende (unabhängig von der Reihenfolge der Ausführung)?

(b) Welche Reihenfolge der `Find`-Operationen ergibt eine minimale Laufzeit (mit Beweis der Minimalität)?

**2. Aufgabe**

Wir betrachten eine beliebige Folge von `Union`- und `Find`-Operationen, wobei alle `Unions` vor allen `Finds` ausgeführt werden. Zeigen Sie: Mit Wegkompression (mit und ohne `Union-by-size`) ist die Zeit für diese Folge von Operationen linear in der Länge der Folge.

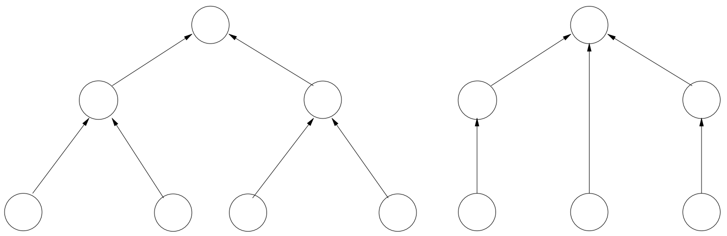
Hinweis: Betrachten Sie die Menge der Knoten, die einen Großvater haben.

**3. Aufgabe**

Geben Sie eine Implementierung von `Delete(`$x$`)` an, so daß `Delete(`$x$`)` das Element $x$ aus seiner Menge unserer `Union-Find-Datenstruktur` löscht und die Laufzeit gemäß Vorlesung für `Unions` und `Finds` mit der Heuristik `Union-by-size` sowie Wegkompression erhalten bleibt.

**4. Aufgabe**

Welcher der beiden Bäume kann bei der Heuristik `Union-by-size` entstehen?



Begründen Sie Ihre Antwort!