

1. Übung zur Vorlesung:

Ausgewählte Kapitel aus "Algorithmen und Datenstrukturen"

Wintersemester 2009/10

6. November 2009

Aufgabe 1.1:

Zeigen Sie für das *Union-Find*-Problem, dass jede Folge von UNION-Operationen auf einer Anfangspartition aus n Blöcken die Länge $n - 1$ hat.

Aufgabe 1.2:

Untersuchen Sie die *worst-case* Kosten einer einzelnen UNION-Operation für Lösung 1.2 (relate the smaller half) aus der Vorlesung.

Aufgabe 1.3:

Betrachten Sie folgende Variante der Weighted-Union-Regel: Mache die Wurzel des Baumes mit der kleineren *Höhe* zum Kind der Wurzel des Baumes mit der größeren Höhe. Zeigen Sie, dass durch $n - 1$ Unions maximal Bäume der Höhe $O(\log n)$ entstehen.

Aufgabe 1.4:

Zeigen Sie, dass in der Analyse von Union-Find nach der Ausführung aller Unions (d. h. im Baum T) gilt: Für jede Kante x, y ist $Rang(x) < Rang(y)$.

Aufgabe 1.5:

Das *Split-Find*-Problem ist wie folgt definiert:

Verwalte eine Einteilung der Zahlen $\{1, \dots, n\}$ in disjunkte Intervalle, die am Anfang nur aus dem Intervall $[1, n]$ besteht, unter folgenden Operationen:

FIND(i):

liefert das Intervall, das die Zahl i enthält.

SPLIT(i):

ersetze das Intervall $[a, b] = \text{FIND}(i)$ durch die beiden Intervalle $[a, i]$ und $[i + 1, b]$.

Entwickeln Sie eine Datenstruktur, die jede FIND-Operation in Zeit $O(1)$ und jede Folge von SPLIT-Operationen möglichst effizient unterstützt.