technische universität dortmund

SoSe 2017

C. Sohler
A. Krivošija, A. Rey, H. Sandvoß

DAP2 – Präsenzübung 13

Besprechung: 19.07.2017 — 21.07.2017

Präsenzaufgabe 13.1: (Union-Find Datenstruktur)

a) Wir verwalten eine Union-Find Datenstruktur, wie sie in der Vorlesung definiert wurde. Zeichnen Sie die Datenstruktur nach jedem `Union`-Befehl:

```
Make-Set(a)  Make-Set(g)  Make-Set(h)  Union(g,h)
Make-Set(o)  Make-Set(i)  Union(a,o)   Union(i,h)
Make-Set(z)  Union(i,z)   Make-Set(j)  Union(j,a)
Union(a,i)
```

b) Nehmen wir an, es wurde statt doppelverketter Liste als Basis für die Union-Find Datenstruktur ein AVL-Baum verwendet. Beschreiben Sie die Konstruktion der Datenstruktur. Was wären die Laufzeiten für die `Make-Set`, `Find` und `Union` Operationen in dieser Version der Union-Find Struktur? Begründen Sie Ihre Antworten.

c) Nehmen wir an, die Basisstruktur für Union-Find ist eine binäre Max-Halde (Max-Heap). Beschreiben Sie die Konstruktion der Datenstruktur. Was wären die Laufzeiten für die `Make-Set`, `Find` und `Union` Operationen in dieser Version der Union-Find Struktur? Begründen Sie Ihre Antworten.

Zur Erinnerung: Ein binärer Max-Heap ist ein binärer Baum, sodass seine Wurzel den maximalen Wert im gesamten Baum enthält, und seine linke bzw. rechte Teilbäume ebenfalls binäre Heaps sind. Für Details sehen Sie bitte die Vorlesungsfolien über Heaps (angeboten als Zusatzmaterial).

Präsenzaufgabe 13.2: (Zweiter Übungstest, Aufgabe 1)

Kreuzen Sie jede Beziehung an, die gültig ist. Es gibt pro vollständig richtig ausgefüllter Zeile einen Punkt.

f	g	$f(n) \in O(g(n))$	$f(n) \in \Omega(g(n))$	$f(n) \in \Theta(g(n))$
$\frac{1}{10} \cdot 4^n$	$4n^4 + 3n^3$			
$\sqrt[4]{n}$	$(\ln n^2)^3 + 10$			
$n^5 - 5n^3 - 10n$	$10n^5 + 100n^4 + 1000n^3$			
$\sqrt[3]{n} \log^3 n$	$\frac{2}{3}n$			

Präsenzaufgabe 13.3: (Zweiter Übungstest, Aufgabe 2)

Gegeben ist die folgende Rekursionsgleichung:

$$T(n) = \begin{cases} 16 \cdot T\left(\frac{n}{4}\right) + n^3 & \text{, wenn } n > 1 \\ 1 & \text{, sonst.} \end{cases}$$

Sie können im Folgenden annehmen, dass n eine Viererpotenz ist.

a) Leiten Sie eine Funktion $f(n)$ her, für die $T(n) \in \Theta(f(n))$ gilt.

b) Zeigen Sie für die in Aufgabenteil a) angegebene Funktion f, dass $T(n) \in \mathcal{O}(f(n))$ gilt. Nutzen Sie dazu Induktion.

Präsenzaufgabe 13.4: (Zweiter Übungstest, Aufgabe 3)

Die Lufttemperaturen in einer Messstation auf Grönland wurden jeden Tag über n Tage gemessen und die Werte, die alle ganze Zahlen sind, in dem Array $A[1..n]$ gespeichert. Es fiel auf, dass die Temperatur zuerst bis zum Tag k, $1 < k < n$, immer strikt gesunken ist, und ab Tag k immer strikt gestiegen ist. Wir suchen den Index des Tags k.

Wenn beispielsweise das Array $A = \{5, 2, -1, -3, 0, 2\}$ ist, wollen wir den Index $k = 4$ bestimmen.

a) Entwerfen Sie einen Algorithmus, der den Index k des minimalen Elements in dem Array A bestimmt. Geben Sie den Algorithmus auch in Pseudocode an. Für die volle Punktzahl wird ein Algorithmus erwartet, dessen Laufzeit durch $\mathcal{O}(\log n)$ beschränkt ist.

b) Analysieren Sie die Laufzeit Ihres Algorithmus.

c) Beweisen Sie die Korrektheit Ihres Algorithmus.

Präsenzaufgabe 13.5: (Zweiter Übungstest, Aufgabe 4)

Die Aufgabe wurde im Heimblatt 13 als Aufgabe 3 gestellt.