05.11.2001

Theoretische Informatik 1, WS 2001/2002

Übungsblatt 3

Die hier vorgestellten Lösungen sind nicht immer richtig.
Aufgabe 3.1

(a)
Entwicklung der Rekursionsgleichung:

$$\begin{aligned} T(n) &= n^{\frac{1}{2}} \cdot T\left(n^{\frac{1}{2}}\right) + n \\ &= n^{\frac{1}{2}} \left[n^{\frac{1}{2^2}} \cdot T\left(n^{\frac{1}{2^2}}\right) + n^{\frac{1}{2}} \right] + n \\ &= n^{\frac{3}{2^2}} \cdot T\left(n^{\frac{1}{2^2}}\right) + n + n \\ &= n^{\frac{3}{2^2}} \left[n^{\frac{1}{2^3}} \cdot T\left(n^{\frac{1}{2^3}}\right) + n^{\frac{1}{2^2}} \right] + 2n \\ &= n^{\frac{7}{2^3}} \cdot T\left(n^{\frac{1}{2^3}}\right) + 3n \\ &= \ldots \end{aligned}$$

Vermutung:

Behauptung: $T(n) = n^{\frac{2^k-1}{2^k}} \cdot T\left(n^{\frac{1}{2^k}}\right) + k \cdot n$

Beweis durch Induktion über k:

I.V.:

$$k = 0$$
$$T(n) = n^0 \cdot T(n) + 0 \cdot n = T(n)$$

$$k = 1$$
$$T(n) = n^{\frac{1}{2}} \cdot T\left(n^{\frac{1}{2}}\right) + n$$

I.S.:

$$k \to k+1$$

$$\begin{aligned} T(n) &= n^{\frac{2^k-1}{2^k}} \cdot T\left(n^{\frac{1}{2^k}}\right) + k \cdot n \\ &= n^{\frac{2^k-1}{2^k}} \left[n^{\frac{1}{2^{k+1}}} \cdot T\left(n^{\frac{1}{2^{k+1}}}\right) + n^{\frac{1}{2^k}} \right] + k \cdot n \\ &= n^{\frac{2^{k+1}-1}{2^{k+1}}} \cdot T\left(n^{\frac{1}{2^{k+1}}}\right) + n + k \cdot n \\ &= n^{\frac{2^{k+1}-1}{2^{k+1}}} \cdot T\left(n^{\frac{1}{2^{k+1}}}\right) + n(k+1) \end{aligned}$$

Zurückführung auf Basisfall:

$$T(2) = 1$$

k soll so gewählt werden, dass in der bewiesenen Vermutung der Basisfall verwendet werden kann.

$$n^{\frac{1}{2^k}}=2 \quad \Leftrightarrow \quad \log_n 2=\frac{1}{2^k} \quad \Leftrightarrow \quad \frac{1}{\log_n 2}=2^k \quad \Leftrightarrow \quad \log_2\left(\frac{1}{\log_n 2}\right)=k \quad \Leftrightarrow \quad \log_2(\log_2 n)=k$$

Mit $k=\log_2(\log_2 n)$ und $2^{\log_2(\log_2 n)}=(\log_2 n)^{\log_2 2}=\log_2 n$ folgt:

$$T(n)=n^{\frac{\log_2 n-1}{\log_2 n}}\cdot T(2)+\log_2(\log_2 n)\cdot n=n^{\frac{\log_2 n-1}{\log_2 n}}+\log_2(\log_2 n)\cdot n$$

(b)

Entwicklung der Rekursionsgleichung:

$$\begin{aligned}
T(n)&=2\cdot n^{\frac{1}{2}}\cdot T\left(n^{\frac{1}{2}}\right)+n\\
&=2\cdot n^{\frac{1}{2}}\left[2\cdot n^{\frac{1}{2^2}}T\left(n^{\frac{1}{2^2}}\right)+n^{\frac{1}{2}}\right]+n\\
&=2^2\cdot n^{\frac{3}{2^2}}\cdot T\left(n^{\frac{1}{2^2}}\right)+2n+n\\
&=2^2\cdot n^{\frac{3}{2^2}}\left[2\cdot n^{\frac{1}{2^3}}\cdot T\left(n^{\frac{1}{2^3}}\right)+n^{\frac{1}{2^2}}\right]+2n+n\\
&=2^3\cdot n^{\frac{7}{2^3}}T\left(n^{\frac{1}{2^3}}\right)+2^2n+2n+n\\
&=2^3\cdot n^{\frac{7}{2^3}}T\left(n^{\frac{1}{2^3}}\right)+7n\\
&=\ldots
\end{aligned}$$

Vermutung:

Behauptung: $T(n)=2^k\cdot n^{\frac{2^k-1}{2^k}}\cdot T\left(n^{\frac{1}{2^k}}\right)+n\cdot(2^k-1)$

Beweis durch Induktion über k:

I.V.:

$$k=0$$
$$T(n)=2^0 n^0\cdot T(n)+n\cdot 0=T(n)$$

$$k=1$$
$$T(n)=2\cdot n^{\frac{1}{2}}\cdot T\left(n^{\frac{1}{2}}\right)+n$$

I.S.:

$$k\to k+1$$

$$\begin{aligned}
T(n)&=2^k\cdot n^{\frac{2^k-1}{2^k}}\cdot T\left(n^{\frac{1}{2^k}}\right)+n(2^k-1)\\
&=2^k\cdot n^{\frac{2^k-1}{2^k}}\left[2\cdot n^{\frac{1}{2^{k+1}}}\cdot T\left(n^{\frac{1}{2^{k+1}}}\right)+n^{\frac{1}{2^k}}\right]+n(2^k-1)\\
&=2^{k+1}\cdot n^{\frac{2^{k+1}-1}{2^{k+1}}}\cdot T\left(n^{\frac{1}{2^{k+1}}}\right)+2^k\cdot n+2^k\cdot n-n
\end{aligned}$$

$$=2^{k+1}\cdot n^{\frac{2^{k+1}-1}{2^{k+1}}}\cdot T\left(n^{\frac{1}{2^{k+1}}}\right)+2\cdot 2^k\cdot n-n$$
$$=2^{k+1}\cdot n^{\frac{2^{k+1}-1}{2^{k+1}}}\cdot T\left(n^{\frac{1}{2^{k+1}}}\right)+n(2^{k+1}-1)$$

<u>Zurückführung auf Basisfall:</u>

$T(2)=1$

k soll so gewählt werden, dass in der bewiesenen Vermutung der Basisfall verwendet werden kann.

$$n^{\frac{1}{2^k}}=2 \quad\Leftrightarrow\quad \log_n 2=\frac{1}{2^k} \quad\Leftrightarrow\quad \frac{1}{\log_n 2}=2^k \quad\Leftrightarrow\quad \log_2\left(\frac{1}{\log_n 2}\right)=k \quad\Leftrightarrow\quad \log_2(\log_2 n)=k$$

Mit $k=\log_2(\log_n 2)$ folgt:

$$T(n)=\log_2(n)\cdot n^{\frac{\log_2 n-1}{\log_2 n}}\cdot T(2)+n(\log_2 n-1)=\log_2(n)\cdot n^{\frac{\log_2 n-1}{\log_2 n}}+n(\log_2 n-1)$$

(c)

<u>Entwicklung der Rekursionsgleichung:</u>

$$T(n)=2T\left(\frac{n}{2}\right)+n\log_2 n$$
$$=2\left[2T\left(\frac{n}{2^2}\right)+\frac{n}{2}\log_2\left(\frac{n}{2}\right)\right]+n\log_2 n$$
$$=2^2T\left(\frac{n}{2^2}\right)+n\log_2\left(\frac{n}{2}\right)+n\log_2 n$$
$$=2^2\left[2T\left(\frac{n}{2^3}\right)+\frac{n}{2^2}\log_2\left(\frac{n}{2^2}\right)\right]+n\log_2\left(\frac{n}{2}\right)+n\log_2 n$$
$$=2^3T\left(\frac{n}{2^3}\right)+n\log_2\left(\frac{n}{2^2}\right)+n\log_2\left(\frac{n}{2}\right)+n\log_2 n$$
$$=\dots$$

<u>Vermutung:</u>

Behauptung: $T(n)=2^k\cdot T\left(\frac{n}{2^k}\right)+n\sum_{i=0}^{k-1}\log_2\left(\frac{n}{2^i}\right)$

<u>Beweis durch Induktion über k:</u>

I.V.:

$k=0$
$T(n)=2^0\cdot T(n)+n\cdot 0=T(n)$

$k=1$

$$T(n) = 2 \cdot T\left(\frac{n}{2}\right) + n \log_2 n$$

I.S.:

$$k \to k+1$$

$$\begin{aligned}
T(n) &= 2^k \cdot T\left(\frac{n}{2^k}\right) + n \sum_{i=0}^{k-1} \log_2\left(\frac{n}{2^i}\right) \\
&= 2^k \left[2 \cdot T\left(\frac{n}{2^{k+1}}\right) + \frac{n}{2^k} \log_2\left(\frac{n}{2^k}\right) \right] + n \sum_{i=0}^{k-1} \log_2\left(\frac{n}{2^i}\right) \\
&= 2^{k+1} \cdot T\left(\frac{n}{2^{k+1}}\right) + n \log_2\left(\frac{n}{2^k}\right) + n \sum_{i=0}^{k-1} \log_2\left(\frac{n}{2^i}\right) \\
&= 2^{k+1} \cdot T\left(\frac{n}{2^{k+1}}\right) + n \sum_{i=0}^{k} \log_2\left(\frac{n}{2^i}\right)
\end{aligned}$$

<u>Zurückführung auf Basisfall:</u>

$$T(1) = 1$$

k soll so gewählt werden, dass in der bewiesenen Vermutung der Basisfall verwendet werden kann.

$$\frac{n}{2^k} = 1 \quad \Leftrightarrow \quad 2^k = n \quad \Leftrightarrow \quad k = \log_2 n$$

Mit $k = \log_2 n$ folgt:

$$\begin{aligned}
T(n) &= 2^{\log_2 n} \cdot T\left(\frac{n}{2^{\log_2 n}}\right) + n \sum_{i=0}^{\log_2 n - 1} \log_2\left(\frac{n}{2^i}\right) = n \cdot T(1) + n \sum_{i=0}^{\log_2 n - 1} \log_2\left(\frac{n}{2^i}\right) \\
&= n \cdot + n \sum_{i=0}^{\log_2 n - 1} \log_2\left(\frac{n}{2^i}\right)
\end{aligned}$$

<u>Aufgabe 3.2:</u>

Die Datenstruktur besteht aus zwei Stacks und einer Variablen m zum Speichern des kleinsten Elements. Zu Beginn sind beide Stacks leer und m ist kein Wert zugewiesen.

Beim Speichern einer Zahlen auf dem Stack wird folgendermaßen vorgegangen:

1. Lege Zahl x mit push(x) auf den Stack.
2. Vergleiche x mit dem Wert von m. Wenn x < m , dann übergebe m den Wert von x. (Falls m noch kein Wert zugewiesen, wird m den Wert von x zugewiesen.)
3. Kopiere den Wert von m auf den Hilfsstack.

Laufzeit:

push(x) auf den Stack $\rightarrow$ O(1)
Vergleich von x mit m $\rightarrow$ O(1)
Variable m evtl. den Wert von x zuweisen $\rightarrow$ O(1)

push(x) wird also in konstanter Zeit durchgeführt.

Beim Entfernen der obersten Zahl vom Stack (pop) wird folgendermaßen vorgegangen:

1. Entferne die oberste Zahl vom Stack und vom Hilfsstack.
2. Vergleiche m mit der obersten Zahl des Hilfsstacks. Wenn die oberste Zahl auf dem Hilfsstack größer ist als m, dann übergebe m den Wert dieser Zahl.

Laufzeit:

pop vom Stack → O(1)
pop vom Hilfsstack → O(1)
Vergleich von m mit oberstem Wert des Hilfsstacks → O(1)
Variable m evtl. neuen Wert zuweisen → O(1)

pop wird also in konstanter Zeit durchgeführt.

Beim Aufruf von minfind() muss nur der Wert von m ausgelesen werden. Das geschieht in konstanter Zeit.