Datenstrukturen (DS)

Sommersemester 2017

Prof Dr. Martin Hoefer
Niklas Hahn

GOETHE UNIVERSITÄT FRANKFURT AM MAIN
Institut für Informatik
Algorithmen und Komplexität

Übung 2

Ausgabe: 02.05.2017
Abgabe: 16.05.2017

Aufgabe 2.1. *Master-Theorem* (9 Punkte)

Benutzen Sie das Master-Theorem, um die Laufzeitklasse folgender rekursiver Funktionen $T(n)$ zu berechnen. Es kann angenommen werden, dass $T(1)$ konstant und n eine Potenz von b ist.

a) $T(n) = 2T(\frac{n}{16}) + \sqrt{7n}$

b) $T(n) = 27 \cdot T(\frac{n}{3}) + n^2 \log_2 n$

c) $T(n) = 9 \cdot T(\frac{n}{3}) + (5 \log_3 n)\sqrt{n} + (2^{2 \log_2 n})$

Aufgabe 2.2. *Potenzieren* (8 Punkte)

Betrachten Sie die folgenden beiden Varianten, die Potenz a^n für $a \in \mathbb{N}_{>0}, n \in \mathbb{N}$ zu berechnen:

Variante 1:

```
int potA(int a, int n) {
  if (n == 0) {return 1;}
  return a*potA(a,n-1);
}
```

Variante 2:

```
int potB(int a, int n) {
  if (n == 0) {return 1;}
  if (n % 2 == 1) {
    return a*potB(a,n-1);}
  return potB(a*a,n/2);
}
```

Bestimmen Sie die Laufzeiten der beiden Funktionen asymptotisch exakt.

Bitte wenden!

Aufgabe 2.3. *Pseudocode* (12 Punkte)

Gegeben ist ein Array der Länge n von Zahlen $(A[1], ..., A[n])$.

Seien $i, k \in \{1, ..., n\}$ und $i + k \leq n$. Das Array $(A[i], ..., A[i + k - 1])$ wird ein *Teilarray* von A mit Länge k genannt. Der Durchschnittswert eines Arrays $(A[i], ..., A[i + k - 1])$ ist die Summe aller Zahlen geteilt durch die Länge des Arrays, in diesem Fall also $\frac{1}{k} \sum_{j=i}^{i+k-1} A[j]$.

Beschreiben Sie jeweils einen Algorithmus in Pseudocode, geben eine kurze Begründung für die Korrektheit und analysieren seine *worst-case* Laufzeit.

a) Bestimmen Sie, ob die Summe aller Elemente mit geradem Index mindestens so groß wie die Summe aller Elemente mit ungeradem Index ist.

b) Zählen Sie, wie oft die einzelnen Elemente im Array vorkommen. Geben Sie am Ende ein Element aus, das am häufigsten vorkommt, sowie die absolute Häufigkeit des Elements im Array.

c) Berechnen Sie den größten Durchschnittswert aller Teilarrays.

Hinweis: Die Laufzeit der Algorithmen muss nicht optimal sein.

Aufgabe 2.4. (8 Punkte)

Wir betrachten folgenden Algorithmus für eine ganzzahlige Eingabe $n \geq 0$:
Algorithmus A:

```
var sum = 0;
for (i = 0 ; i ≤ n ; i++){
    for (j = 1 ; j ≤ f(i); j++){
        sum=sum+1
    }
}
```

Bestimmen Sie für jede der folgenden Funktionen f den Wert der Variable *sum* am Ende der Berechnung exakt.

a) $f(i) = n + i$

b) $f(i) = a \cdot n$ für ein $a \in \mathbb{N}$.

c) $f(i) = (a - 1) \cdot a^i$ für ein $a \in \mathbb{N}_{>1}$.

d) $f(i) = n/(2^i)$

Hinweis: Die Formel für Partialsummen der geometrischen Reihe (Satz 2.7 im Skript) kann hilfreich sein. Für Aufgabenteil *d*) kann angenommen werden, dass $n = 2^k$ für ein $k \in \mathbb{N}$ eine Zweierpotenz ist.