

Algorithmen und Datenstrukturen, Übung 2

Marouane Soussi, Lars Happel

April 2022

Aufgabe 1

siehe .java Datei

Aufgabe 2 - Vielfache von 3

a) Pseudocode

```
sum := 0
for i=1 to n do:
    if (a[i] % 3 == 0):
        sum += a[i]
return sum // Wenn Sum=0 ist hatte a keine Vielfachen von 3
```

b) Korrektheitsbeweis

Schleifeninvariante: Zu Beginn jeder Iteration ist in sum die Summe aller Vielfachen von 3 in $A[1, i - 1]$ gespeichert.

Initialisierung: $A[1, n] = A[1, 1]$

Nach der Schleife steht in der Summe:
$$\begin{cases} A[1], & \text{falls } A[1] \text{ ein Vielfaches von 3 ist.} \\ 0 & \text{sonst} \end{cases}$$

Fortsetzung: $(n \rightarrow n + 1)$

Nach Schleifeninvariante ist in der Summe die Summe der Vielfachen von 3 im Teilarray $A[1, i - 1]$ abgelegt. In der nächsten Iteration diese Summe um $A[i]$ inkrementiert, falls $A[i]$ ein Vielfaches von 3 ist, oder die Summe bleibt gleich.

Terminierung: Die Schleife terminiert wenn $n = n + 1$. (Wenn das gesamte Array einmal durchlaufen wurde.)

Sei t_i die Anzahl an positiven Auswertungen des if, bzw. die Anzahl an Vielfachen von 3 im Array.

Kosten	Anzahl
c_1	1
c_2	$n + 1$
c_3	n
c_4	$\sum_{i=1}^n t_i$
c_5	1

Worst Case: Das gesamte Array enthält nur Vielfache von 3. In diesem Fall ist $t_i = 1 \rightarrow \sum_{i=1}^n t_i = n$.

$$T(A) = c_1 + n * c_2 + c_2 + n * c_3 + n * c_4 + c_5 = n(c_2 + c_3 + c_4) + c_1 + c_5 + c_2 \in \Theta(n)$$

In diesem Fall ist die Laufzeit linear, also $O(n)$.

Aufgabe 3 - Pseudocode

a)

1. Was macht der Code?

Die Methode berechnet das Produkt der beiden größten Werte des Array

2. Laufzeit

Die Laufzeit ist $O(n^2)$ da für jede Iteration der äusseren Schleife die innere Schleife einmal komplett n durchläuft.

Kosten	Anzahl
c_1	1
c_2	$n + 1$
c_3	$(\sum_{i=1}^n n) + 1 = n^2 + 1$
c_4	$(\sum_{i=1}^n n) = n^2$
c_5	$(\sum_{i=1}^n n) - 1 = n^2 - 1$
c_6	$(\sum_{i=1}^n n) - 1 = n^2 - 1$
c_7	1

$$T(n) = c_1 + c + 2(n + 1) + (n^2 + 1)c_3 + n^2c_4 + (n^2 - 1)(c_5 + c_6) + c_7$$

$$= n^2(c_3 + c_4 + c_5 + c_6) + n(c_2 + c_1 + c_2 + c_3 - c_5 - c_6 + c_7) \in \Theta(n^2)$$

b)

```

x = 0
y = 0
for i=1 to n do:
    if (a[i] > x):
        y = x
        x = a[i]
```

```
    else if (a[i] > y):  
        y = a[i]  
    return x * y
```

Aufgabe 4 - Knobelaufgabe

```
results[] = Array von 0 bis 2n  
a = 1  
b = 2n  
for i = 0 to 2n do:  
    if (a > b):  
        results[i] = input[a]  
        a += 1  
    else:  
        results[i] = input[b]  
        b -= 1  
return results[]
```