

## DAP2 – Präsenzübung 3

Besprechung: 02.05.2018 — 04.05.2018

### Abgabe:

Präsenzübungen müssen nicht zu Hause bearbeitet werden, sondern werden unter Anleitung während der Übung erarbeitet.

### Präsenzaufgabe 3.1: (Rekursionsgleichung)

Gegeben sei die Rekursionsgleichung:

$$T(n) = \begin{cases} 9 \cdot T(n/3) + n^3 & \text{wenn } n > 1 \\ 1 & \text{sonst} \end{cases}$$

- Bestimmen Sie eine asymptotische obere Schranke für  $T(n)$  und beweisen Sie sie mittels Induktion. Sie dürfen annehmen, dass  $n$  von der Form  $3^k$  für ein  $k \in \mathbb{N}$  ist.
- Begründen Sie, warum die Annahme über  $n$  keine Beschränkung der Allgemeinheit darstellt.

### Präsenzaufgabe 3.2: (Teile und Herrsche)

Gegeben sei der folgende rekursive Sortieralgorithmus:

Sort(Array  $A$ , Integer  $l$ , Integer  $r$ ):

```
1 if  $l < r - 1$  then
2    $k \leftarrow (r - l + 1) \text{ div } 3$ 
3   Sort( $A, l, r - k$ )
4   Sort( $A, l + k, r$ )
5   Sort( $A, l, r - k$ )
6 else
7   if  $A[l] > A[r]$  then // Vertauschen von  $A[l]$  und  $A[r]$ 
8      $temp \leftarrow A[l]$ 
9      $A[l] \leftarrow A[r]$ 
10     $A[r] \leftarrow temp$ 
```

Die Sortierung des Arrays  $A$  erfolgt durch den Aufruf Sort( $A, 1, \text{length } [A]$ ).

- Bestimmen Sie asymptotische Worst-Case Komplexität des angegebenen Sortieralgorithmus. Nehmen Sie für die Eingabelänge  $n$  des Arrays eine Dreierpotenz  $n = 3^k$  an.
- Beweisen Sie die Korrektheit des Algorithmus.