



# Grundzüge der Informatik 1

Vorlesung 9 - flipped classroom

# Teile und Herrsche

## Aufgabe 1

- Eine Inversion in einem Feld ist ein Indexpaar  $(i,j)$ ,  $i < j$ , mit  $A[i] > A[j]$
- Entwerfen Sie einen Teile & Herrsche Algorithmus zur Bestimmung der Anzahl von Inversionen
- Hinweis: Wie können Sie die Anzahl Inversionen zwischen  $A[p..m-1]$  und  $A[m..r]$  bestimmen, wenn beide Teilbereiche sortiert sind?
- Was ist die Laufzeit Ihres Algorithmus?

# Teile und Herrsche

## Inversionen2(A,p,r)

1. **if**  $p=r$  **then return** 0
2.  $m = \lfloor (p + r) / 2 \rfloor$
3.  $C = \text{Inversionen2}(A,p,m) + \text{Inversionen2}(A,m+1,r)$
4.  $\text{Mergesort}(A,p,m)$
5.  $\text{Mergesort}(A,m+1,r)$
6.  $j = m$
7. **for**  $i=p$  **to**  $m$  **do**
8.     **while**  $j < r$  and  $A[j] < A[i]$  **do**  $j = j + 1$
9.      $C = C + j - (m+1)$
10. **return**  $C$

# Teile und Herrsche

## Laufzeit

- $T(n) = 2 T(n/2) + c n \log n$
- $T(1) = c$

## Behauptung:

- $T(n) \leq 2c n \log^2 n$

## Beweis:

- Ind. Anfang:  $T(2) = 4c \leq 2c n \log^2 n$  für  $n=2$
- Ind. Annahme: Sei  $n > 2$  eine Zweierpotenz. Für  $m=n/2$  gilt:  $T(m) \leq 2c m \log^2 m$
- Ind. Schluss:  $T(n) = 2 T(n/2) + c n \log n \leq 2 \cdot 2c n/2 \log^2 (n/2) + c n \log n$
- $\leq 2 (c n \log n) (\log n - 1) + c n \log n \leq 2 c n \log^2 n$