

5.5 Quicksort

150

- Benutzt wie Mergesort eine divide-and-conquer-Strategie.
- Rekursion
- Komplexität / Laufzeit
 - im Worst-Case: $O(n^2)$
 - im Mittel (Average-Case): $O(n \log n)$
- Überblick
 - 5.5.1 Beschreibung
 - 5.5.2 Worst-Case
 - 5.5.3 Best-Case
 - 5.5.4 Average-Case

1. 0 ist klein. → bleibt vorne. P_2++ . P_1++ .

2. 6 ist groß. → gehe weiter P_2++ .

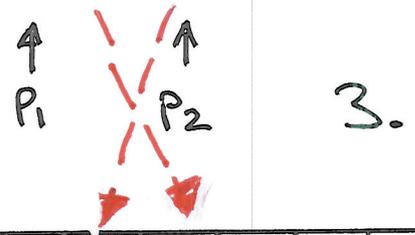
3. 4 ist klein. → tausche an das Ende der kleinen Elemente.

$$A[P_1+1] \leftrightarrow A[P_2]$$

gehe weiter.

P_2++ . P_1++ .

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| / | 0 | 6 | 4 | 7 | 2 | 1 | 3 | 9 | 8 | 5 |



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| / | 0 | 4 | 6 | 7 | 2 | 1 | 3 | 9 | 8 | 5 |



4. 7 ist groß. → gehe weiter. P_2++

5. 2 ist klein → Tausch:

$$A[P_1+1] \leftrightarrow A[P_2]$$

gehe weiter. P_2++ . P_1++ .

Am Ende: $p_2 = 1$, $p_1 = 5$

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| / | 0 | 4 | 2 | 1 | 3 | 7 | 6 | 9 | 8 | 5 |

↑
 p_1

↑
 p_2

Tausche: $A[p_1 + 1] \leftrightarrow A[p_2]$

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| / | 0 | 4 | 2 | 1 | 3 | 5 | 6 | 9 | 8 | 7 |

Sortiere mit
Quicksort

Sortiere mit
Quicksort

In Pseudocode:

Quicksort (A, p, r)

- 1 IF $p < r$
- 2 $q \leftarrow \text{Partition}(A, p, r)$
- 3 Quicksort(A, p, $q-1$)
- 4 Quicksort(A, $q+1$, r)

Partition (A, p, r)

- 1 $x \leftarrow A[r]$ "Referenzelement"
- 2 $i \leftarrow p-1$
- 3 FOR $j \leftarrow p$ TO $r-1$ DO
- 4 IF $A[j] \leq x$
- 5 $i \leftarrow i+1$
- 6 tausche $A[i] \leftrightarrow A[j]$
- 7 tausche $A[i+1] \leftrightarrow A[r]$
- 8 return $i+1$.