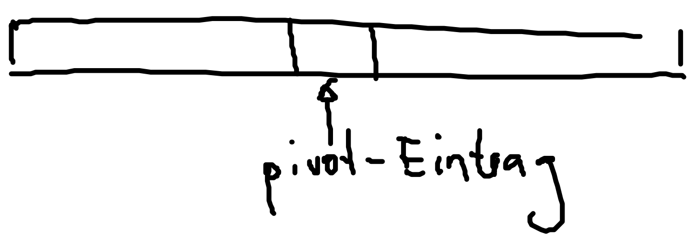
# Quicksort

- empirisch das schnellste Sortierverfahren
- worst-case-Laufzeit: $\Omega(n^2)$
- average-case-Laufzeit: $\Theta(n \log n)$

Der Algorithmus

Gegeben: Array vec mit $n \in \mathbb{N}$ Einträgen

Wähle einen beliebigen Pivoteintrag vec[pivot]




Zerlege das Array in zwei Teilbereiche

$$\rightarrow \underbrace{vec[0], \ldots, vec[k-1]}_{\text{Einträge } i \text{ mit } vec[i] \leq vec[pivot]}$$

$\rightarrow$ vec[k] = vec[pivot]

$\rightarrow$ $\underbrace{\text{vec}[k+1], \ldots, \text{vec}[n]}$

Einträge i mit vec[i] $\geq$ vec[pivot]

Sofern einer dieser Bereiche mehr als ein Element enthält, wende Quicksort rekursiv an.

Beispiel

63 24 12 $\boxed{53}$ 72 18 44 35

pivot

Tausche erstes Element mit Pivot-Element

$\boxed{53}$ 24 12 63 72 18 44 35

lowSwap highSwap

Tausche Einträge von vec[lowSwap] und vec[highSwap]

$\boxed{53}$ 24 12 35 72 18 44 63

Tausche ...

$\boxed{53}$ 24 12 35 44 18 72 63

$\boxed{53}$ 24 12 35 44 18 72 63

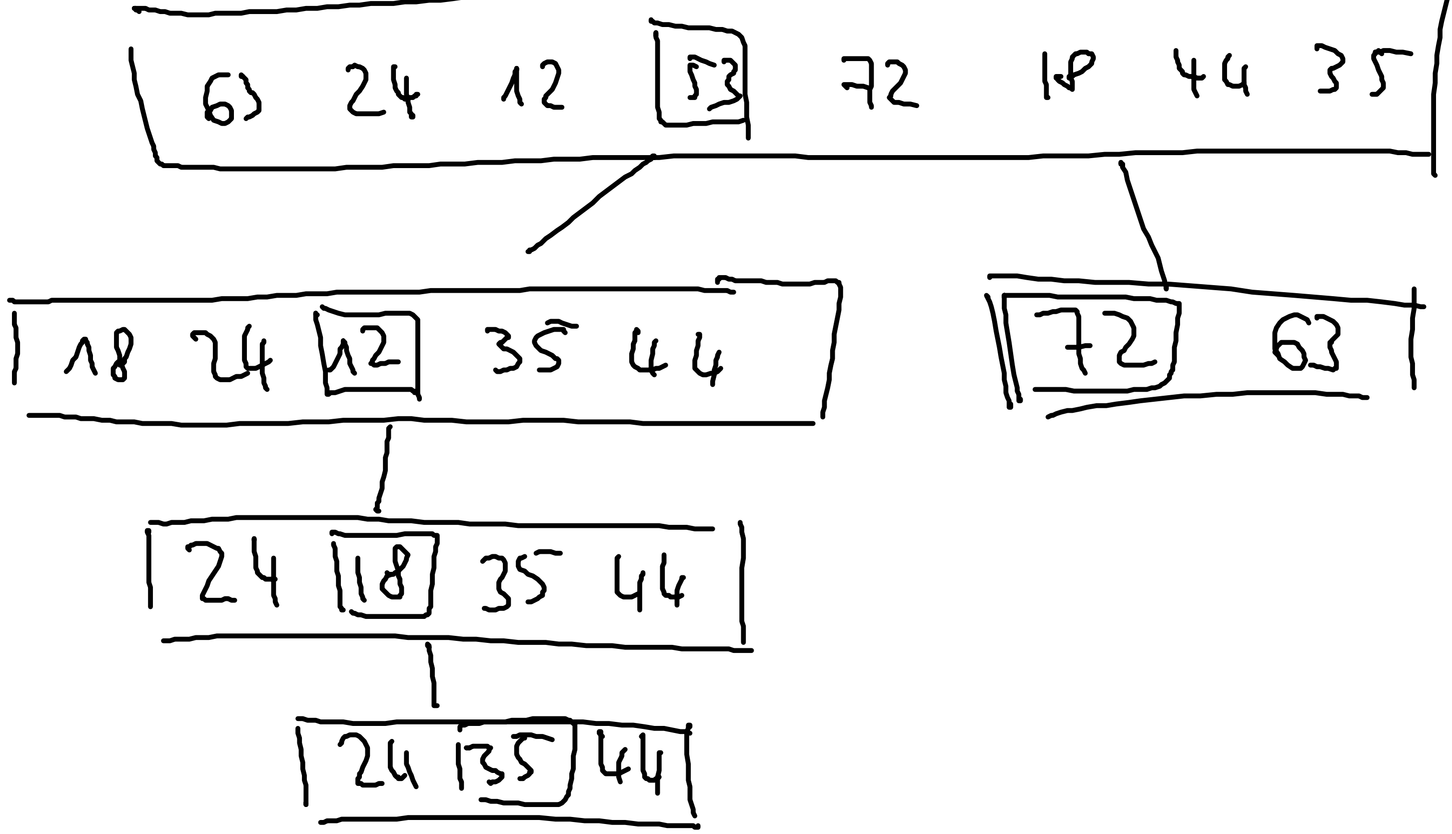
highSwap lowSwap

Tausche Einträge von vec[0] und vec[highSwap]

$$\underbrace{18 \quad 24 \quad 12 \quad 35 \quad 44}_{\leq 53} \quad \underbrace{\boxed{53}}_{53} \quad \underbrace{72 \quad 63}_{\geq 53}$$

<u>Lemma</u>: Enthält der aufzuteilende Bereich $n \in \mathbb{N}$ Einträge, so benötigen wir zum Aufteilen $n-1$ Vergleiche.

Rekursionsbaum




Überlegung: Für ein Array der Länge $n \in \mathbb{N}$, ist die Rekursionstiefe von Quicksort

$$T(n) \leq n-1$$

Für die Anzahl Rekursiver Aufrufe gilt ebenfalls

$$R(n) \leq n-1.$$

Der Worst-Case-Aufwand von Quicksort ist $\Theta(n^2)$

$$C(n) \leq (n-1) + (n-2) + \dots + 2 + 1 = \frac{n(n-1)}{2} \in \Theta(n^2).$$

Wollen zeigen, dass dieser Worst-case-Fall sehr selten auftritt, und dass amortisiert über alle Permutationen eine mittlere Laufzeit von

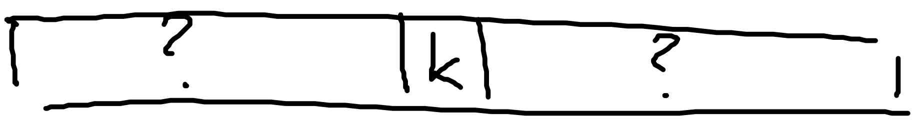
$$\mathcal{O}(n \log n) = \overline{C}(n)$$

gilt.

Sei $\Pi$ die Menge aller Permutationen von $\{1, \dots, n\}$, und sei $C(\pi)$ die Anzahl Vergleiche, die Quicksort zum Sortieren der Permutation $\pi$ benötigt, dann gilt

$$\boxed{\overline{C}(n) = \frac{1}{n!} \sum_{\pi \in \Pi} C(\pi)} \qquad \boxed{|\Pi| = n!}$$

Für $k \in \{1,\dots,n\}$ sei $\Pi_k \subseteq \Pi$ die Menge der Permutationen, in denen das Pivotelement $k$ ist.



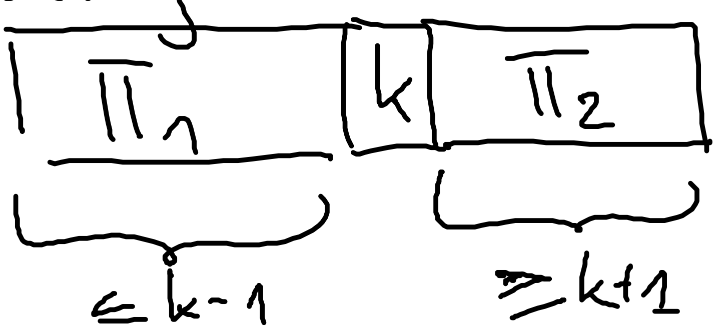

$\pi \in \Pi_k$

$$|\Pi_k| = (n-1)!$$

$$\sum_{\pi \in \Pi_k} C(\pi) = \sum_{\pi \in \Pi_k} \left[ Z(\pi) + C(\Pi_1) + C(\Pi_2) \right]$$

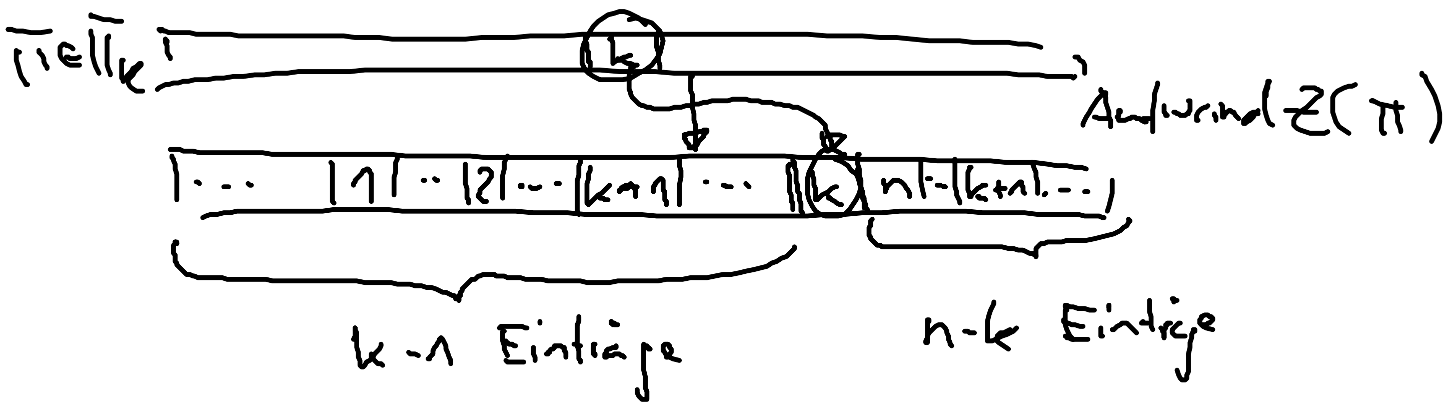
Aufwand zum Teilen des Arrays $\pi$ in die Teilarrays

(rekursive) Aufwand zum Sortieren von $\Pi_1$ bzw. $\Pi_2$



$$\sum_{\pi\in\Pi_k}{}' C(\pi) = \underbrace{\sum_{\pi\in\Pi_k}{}' Z(\pi)}_{\substack{< (n-1)!\cdot n \\ = n!}} + \underbrace{\sum_{\pi\in\Pi_k}{}' C(\pi_1)}_{= \frac{(n-1)!}{(k-1)!}\sum'_{\pi_1\in\Pi(k-1)} C(\pi_1)} + \underbrace{\sum_{\pi\in\Pi_k}{}' C(\pi_2)}_{=(n-1)!\,\overline{C}(n-k)}$$

$$= (n-1)!\,\overline{C}(k-1)$$




Wir suchen eine rekursive Formel für $\overline{C}(n)$.

$$\overline{C}(n) = \frac{1}{n!}\sum_{k=1}^{n}{}'\sum_{\pi\in\Pi_k}{}' C(\pi)$$

$$< \frac{1}{n!}\sum_{k=1}^{n}{}'\left(n! + (n-1)!\,\overline{C}(k-1) + (n-1)!\,\overline{C}(n-k)\right)$$

$$= \frac{n!}{n!}\sum_{k=1}^{n}{}' 1 + \frac{1}{n}\sum_{k=1}^{n}{}'\overline{C}(k-1) + \frac{1}{n}\sum_{k=1}^{n}{}'\overline{C}(n-k)$$

$$= n + \frac{1}{n}\sum_{k=0}^{n-1} \overline{C}(k) + \frac{1}{n}\sum_{k=0}^{n-1} \overline{C}(k)$$

$$= n + \frac{2}{n}\sum_{k=0}^{n-1} \overline{C}(k)$$

Wir überlegen uns: $\overline{C}(0) = 0$

$\overline{C}(1) = 0$

$\overline{C}(2) = 1$

$$\overline{C}(n) \leq n + \frac{2}{n}\sum_{k=2}^{n-1} \overline{C}(k) \quad \text{für } n \geq 2.$$

<u>Theorem</u>: Die Lösung $r(n)$ der Rekursionsgleichung $r(n) \leq n + \frac{2}{n}\sum_{k=2}^{n-1} r(k)$ mit den Anfangswerten $r(0) = r(1) = 0$, $r(2) = 1$ erfüllt die Ungleichung

$$r(n) \leq 2n\ln(n).$$

Beweis: Durch vollständige Induktion.

Wir zeigen $r(n) \leq c \cdot n \ln(n)$ mit $c=2$.

Induktionsanfang: $r(2) = 1 \leq 2 \cdot 2 \cdot \ln(2) \approx 2{,}7\ldots$ ✓

Induktionsvoraussetzung: Die Behauptung gelte für alle $k \in \{0, \ldots, n-1\}$

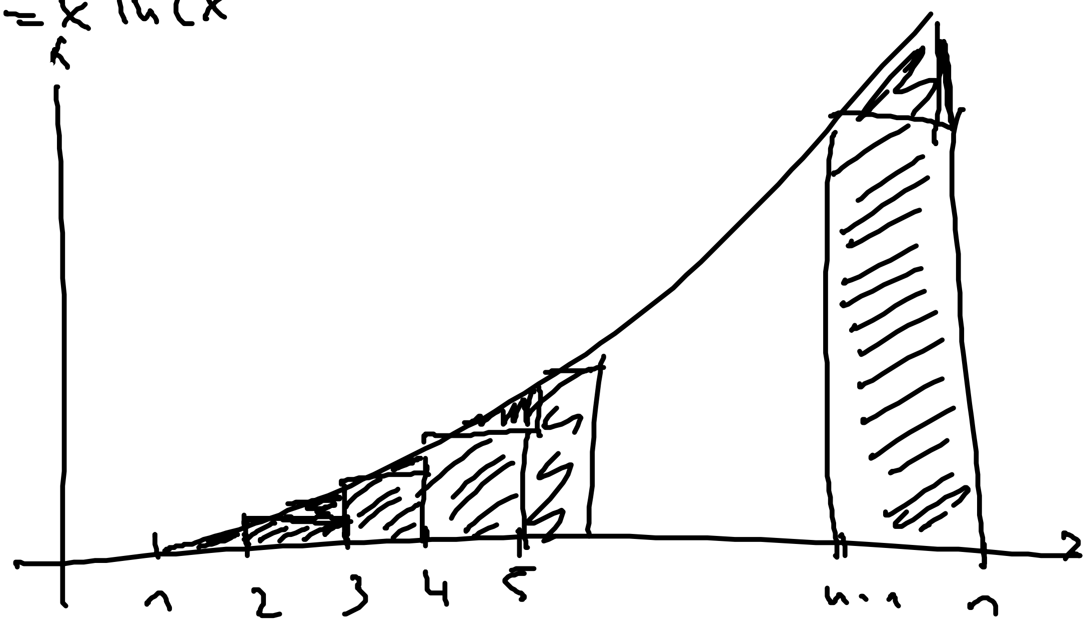
Induktionsschluss:

$$r(n) = n + \frac{2}{n} \sum_{k=2}^{n-1} r(k)$$

$$\leq n + \frac{2}{n} \sum_{k=2}^{n-1} c \cdot k \ln(k) \quad \text{(Induktionsvor.)}$$

$$= n + \frac{2c}{n} \boxed{\sum_{k=2}^{n-1} k \ln(k)}$$ wollen wir abschätzen.

$f(x) = x \ln(x)$




$$\blacksquare = \sum_{k=2}^{n-1} k \ln k$$
$$\leq \int_2^n x \ln(x)\, dx$$

Wir erhalten

$$\sum_{k=2}^{n-1} k \ln(k) \leq \int_2^n x \ln(x)\, dx$$

---

Partielle Integration:

$$\int_a^b f'(x) \cdot g(x)\, dx = \left[ f(x)\, g(x) \right]_a^b - \int_a^b f(x)\, g'(x)\, dx$$

---

$n$

$$\sum_{1}' \cdots \le \int_2^n x \ln(x)\, dx$$

$$= \left[\frac{x^2}{2} \ln x\right]_2^n - \int_2^n \frac{x^2}{2} \cdot \frac{1}{x}\, dx$$

$$= \left[\frac{x^2}{2} \ln x\right]_2^n - \left[\frac{x^2}{4}\right]_2^n$$

$$= \frac{n^2}{2} \ln(n) - 2\ln(2) - \frac{n^2}{4} + 1$$

$$\le \boxed{\frac{n^2}{2} \ln(n) - \frac{n^2}{4}}$$

$$\vee\!\!|$$

Wir erhalten

$$r(n) \le n + \frac{2c}{n} \boxed{\sum_{k=2}^{n-1}{}' k \ln k}$$

$$\le n + \frac{2c}{n}\left(\frac{n^2}{2} \ln(n) - \frac{n^2}{4}\right)$$

$$= n + c \cdot n \cdot \ln(n) - \frac{c}{2} n$$

$$= c \cdot n \cdot \ln(n) + n - \frac{c}{2} n$$

$$\boxed{c = 2}$$

$$= 2n\ln(n) \in \mathcal{O}(n \cdot \log n),$$