

Алгоритмы и структуры данных

Быстрая сортировка

Александр Куликов

Псевдокод

процедура QUICKSORT(A, l, r)

если $l \geq r$:

 ВЫЙТИ

$m \leftarrow \text{PARTITION}(A, l, r)$

QUICKSORT($A, l, m - 1$)

QUICKSORT($A, m + 1, r$)

Разделение за линейное время на месте

функция $\text{PARTITION}(A, l, r)$

$x \leftarrow A[l]$

$j \leftarrow i$

для i от $l + 1$ до r :

если $A[i] \leq x$:

$j \leftarrow j + 1$

обменять местами $A[j]$ и $A[i]$

обменять местами $A[l]$ и $A[j]$

вернуть j

Плохие и хорошие разделители

Случайный разделитель

Оценка времени работы

Теорема

Пусть все ключи массива $A[1 \dots n]$ различны. При случайном выборе разделителя среднее время работы $\text{QUICKSORT}(A)$ есть $O(n \log n)$. Время работы в худшем случае есть $O(n^2)$.

Доказательство: основные идеи

Доказательство: завершение

Одинаковые ключи

Элиминация хвостовой рекурсии: глубина рекурсии $O(\log n)$ в худшем случае

процедура $\text{QUICKSORT}(A, l, r)$

пока $l < r$:

$m \leftarrow \text{PARTITION}(A, l, r)$

$\text{QUICKSORT}(A, l, m - 1)$

$l \leftarrow m + 1$

IntroSort: время работы $O(n \log n)$ в худшем случае