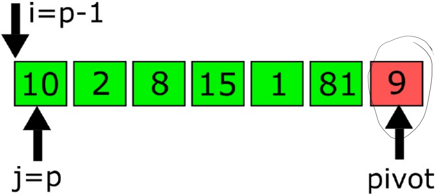
Быстрая сортировка

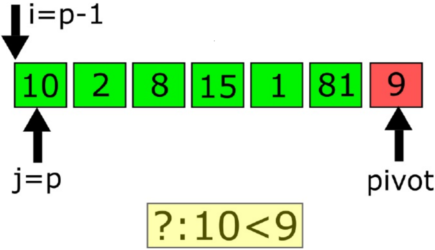
Суть:

- разбить на два подмассива с помощью "разделительного" элемента
- рекурсивно отсортировать оба подмассива
- объединять не надо, так как сортировка производится напрямую в исходном массиве

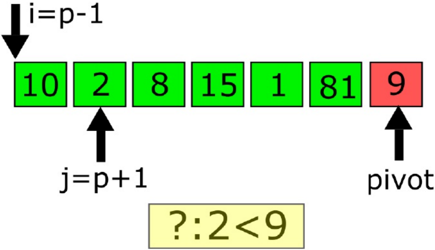
Быстрая сортировка



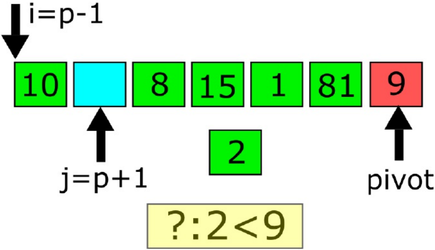
Быстрая сортировка



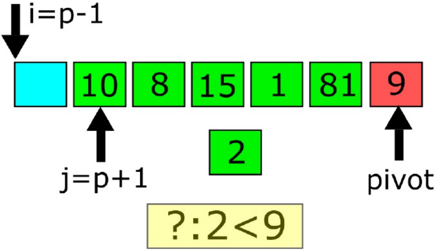
Быстрая сортировка



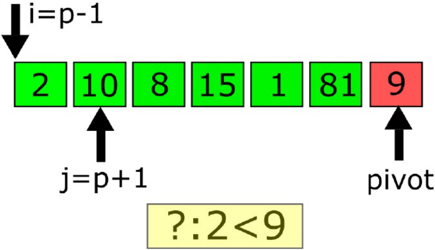
Быстрая сортировка



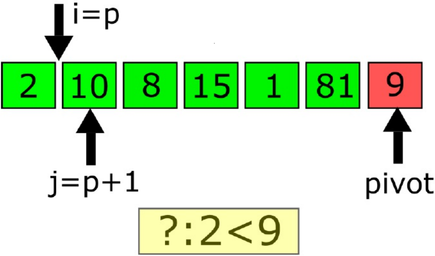
Быстрая сортировка



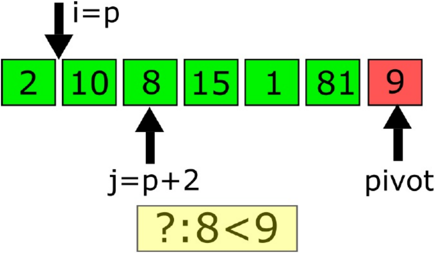
Быстрая сортировка



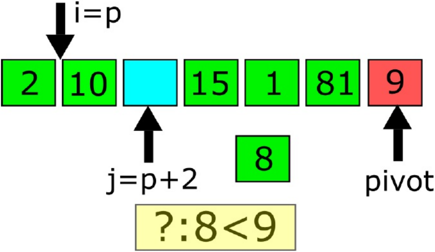
Быстрая сортировка



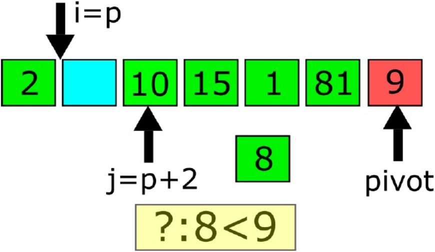
Быстрая сортировка



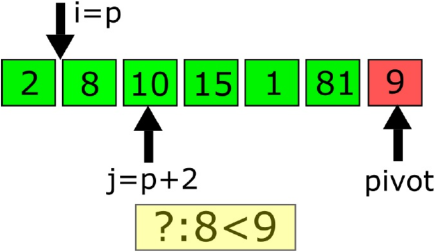
Быстрая сортировка



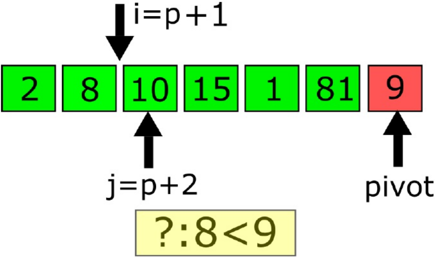
Быстрая сортировка



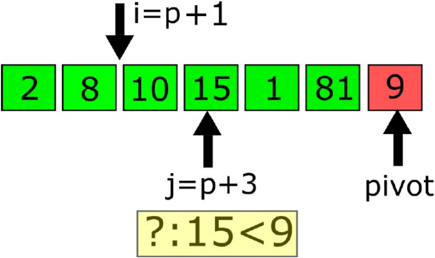
Быстрая сортировка



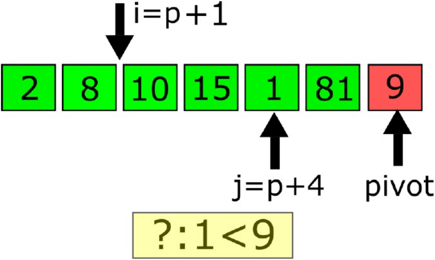
Быстрая сортировка



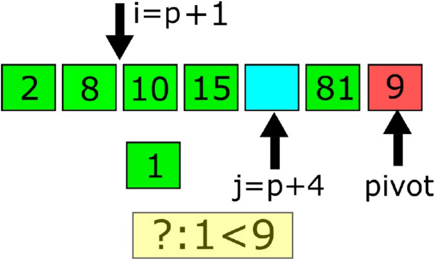
Быстрая сортировка



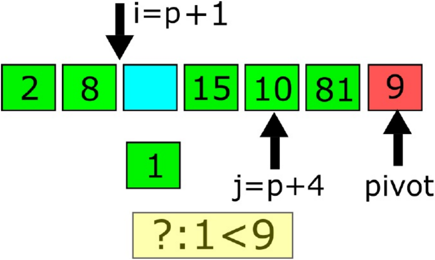
Быстрая сортировка



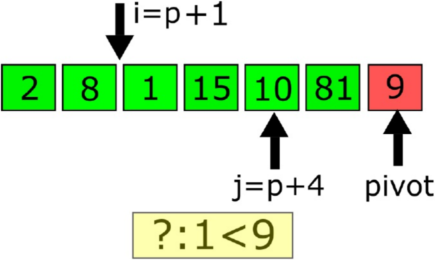
Быстрая сортировка



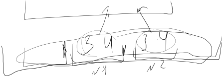
Быстрая сортировка

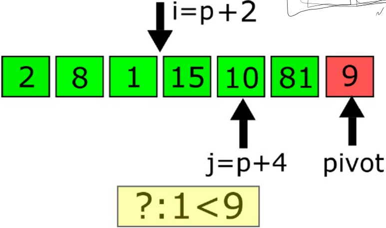


Быстрая сортировка

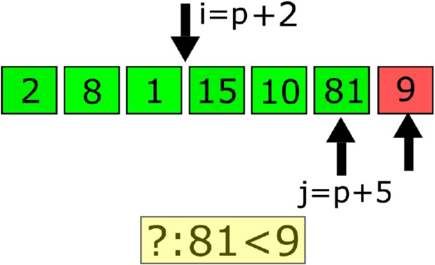


Быстрая сортировка

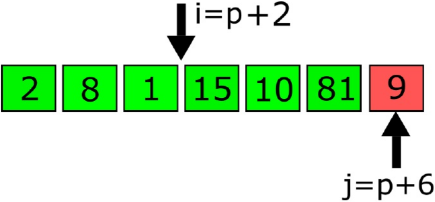




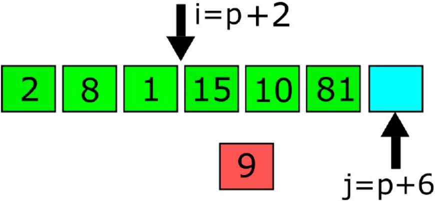
Быстрая сортировка



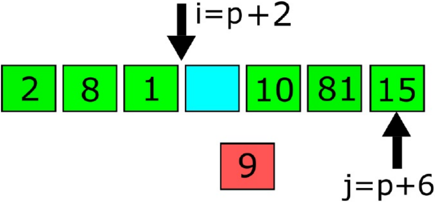
Быстрая сортировка

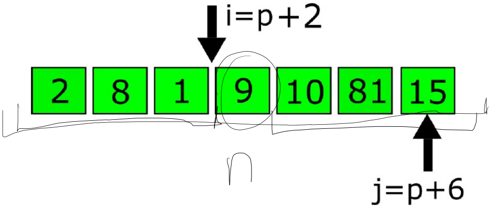


Быстрая сортировка



Быстрая сортировка




i=p+2
2
8
1
9
10
81
15
j=p+6

Быстрая сортировка

1 ... n

n ... 1

$\frac{n}{2}$ $\frac{n}{2}$

$\log n$

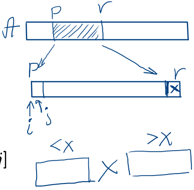
2	8	1	9	10	81	15

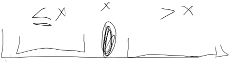
$n \log n$

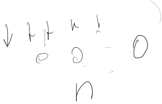
n

Операция разделения

```
PARTITION(A, p, r)
x = A[r]
i = p − 1
for j = p to r − 1
    if A[j] ≤ x
        i = i + 1
        Обменять A[i] и A[j]
Обменять A[i + 1] и A[r]
return i + 1
```







Анализ алгоритма быстрой сортировки

```
QUICKSORT(A, p, r)
if p < r
    q = PARTITION(A, p, r)
    QUICKSORT(A, p, q − 1)
    QUICKSORT(A, q + 1, r)
```

$T(n)$

$T\left(\frac{n}{2}\right)$

$T\left(\frac{n}{2}\right)$

$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$

Наихудшее поведение алгоритма: разбиение делает одну задачу размера n-1, а вторую пустой

$T(n) = \Theta(n) + T(n-1) + T(1) = \Theta(n^2)$

Наилучшее поведение алгоритма: разбиение делает две подзадачи половинного размера

Балансировка разбиения

RANDOMIZED-PARTITION(A, p, r)

1 i = RANDOM(p, r)
2 Обменять $A[r]$ и $A[i]$
3 **return** PARTITION(A, p, r)

В среднем время сортировки: $O(n \lg n)$

свойства: O

$T_1(n) = O(g_1(n))$, $T_2(n) = O(g_2(n))$

1) правило суммы

$T_1(n) + T_2(n) = O(\max(g_1(n), g_2(n)))$

$T_1(n) = O(g_1(n)) \Rightarrow \exists k_1\ \forall n \geq k_1\ \exists c_1$
$0 \leq T_1(n) \leq c_1 g_1(n)$

$T_2(n) = O(g_2(n)) \Rightarrow \exists k_2\ \forall n \geq k_2\ \exists c_1$
$0 \leq T_2(n) \leq c_2 g_2(n)$

$T_1(n) + T_2(n) \leq c_1 g_1(n) + c_2 g_2(n) \leq$

$\exists k = \max(k_1, k_2)$, $\forall n \geq k$ $\exists c = \max(c_1, c_2)$

$\max(g_1(n), g_2(n))$ $\leq c \max(g_1(n), g_2(n))$

$a \leq b \leq c$ $= O(\max(g_1(n), g_2(n)))$

$a \leq c$

2) правило произведения

$T_1(n) \cdot T_2(n) = O(g_1(n) \cdot g_2(n))$

3) констант

$c_1 T_1(n) = O(g_1(n))$

$T_1(n) + c_2 = O(g_1(n))$