

3. Сортировка Шелла

Определение задачи

Сортировка Шелла — это улучшенный вариант сортировки вставками, который позволяет эффективно сортировать массивы за счёт сравнения элементов, расположенных на определённом расстоянии друг от друга. Цель алгоритма — уменьшить количество необходимых сдвигов элементов, постепенно сокращая интервал сравнения.

Алгоритм сортировки Шелла

Алгоритм сортировки Шелла состоит из следующих шагов:

1. Выбирается начальный интервал h , обычно равный половине длины массива.
2. Выполняется сортировка вставками для элементов, отстоящих друг от друга на h позиций.
3. Интервал h сокращается, например, делится на 2.
4. Процесс повторяется до тех пор, пока интервал не станет равен 1.

Анализ сложности

Худший случай

В худшем случае время выполнения сортировки Шелла составляет $O(n^2)$, однако на практике она работает значительно быстрее благодаря уменьшению числа необходимых операций обмена. Сложность зависит от выбора последовательности интервалов h . При использовании оптимальных последовательностей сложность может приближаться к $O(n \log n)$.

Средний случай

В среднем случае сортировка Шелла демонстрирует производительность лучше, чем $O(n^2)$, но хуже, чем $O(n \log n)$. Конкретная временная сложность зависит от выбранной последовательности интервалов. Рассмотрим пример для последовательности, где интервалы образуют геометрическую прогрессию:

$$h_k = \frac{h_{k-1}}{2}$$

Таким образом, интервалы могут быть такими: $h = n/2, n/4, n/8, \dots, 1$. Сумма затрат на сортировку на каждом этапе может быть представлена как:

$$T(n) = n \left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots \right)$$

Эта сумма является сходящейся геометрической прогрессией с суммой:

$$\sum_{k=0}^{\infty} \frac{1}{2^k} = 2$$

Следовательно:

$$T(n) = O(n) \times 2 = O(n)$$

Однако, на практике из-за неравномерного распределения элементов и необходимости выполнения операций вставки, реальная временная сложность оказывается выше и приближается к $O(n^{1.5})$.

Пояснение:

Каждый шаг с интервалом h_k выполняет сортировку вставками на подмассивах с размером примерно $\frac{n}{h_k}$. Для последовательности интервалов $h_k = \frac{h_{k-1}}{2}$, сумма затрат по всем шагам будет:

$$T(n) = n \left(1 + \frac{1}{2} + \frac{1}{4} + \dots \right) = O(n)$$

Однако из-за фактических операций вставки и распределения элементов временная сложность оказывается ближе к $O(n^{1.5})$.

Более оптимальные последовательности, такие как последовательность Седжвика или Штрассена, могут улучшить среднюю производительность, приближая её к $O(n \log n)$. Основной вклад в эффективность среднего случая вносит уменьшение числа сравнений и сдвигов за счёт предварительной сортировки элементов на больших интервалах, что позволяет быстрее двигаться к окончательной упорядоченной последовательности при $h = 1$.