

$O(N \log N)$ сортування

Кшиванський Олександр
27.04.2021

Приклади $O()$

Алгоритм	Складність
Повний перебір	$O(n!)$
Бульбашкове сортування	$O(n^2)$
Сортування злиттям	$O(n \log n)$
Пірамідне сортування	$O(n \log n)$
Швидке сортування	$O(n \log n)$
Сортування деревом	$O(n \log n)$
Алгоритм Каданеса	$O(n)$
Двійковий пошук	$O(\log n)$
Визначення парності числа	$O(1)$

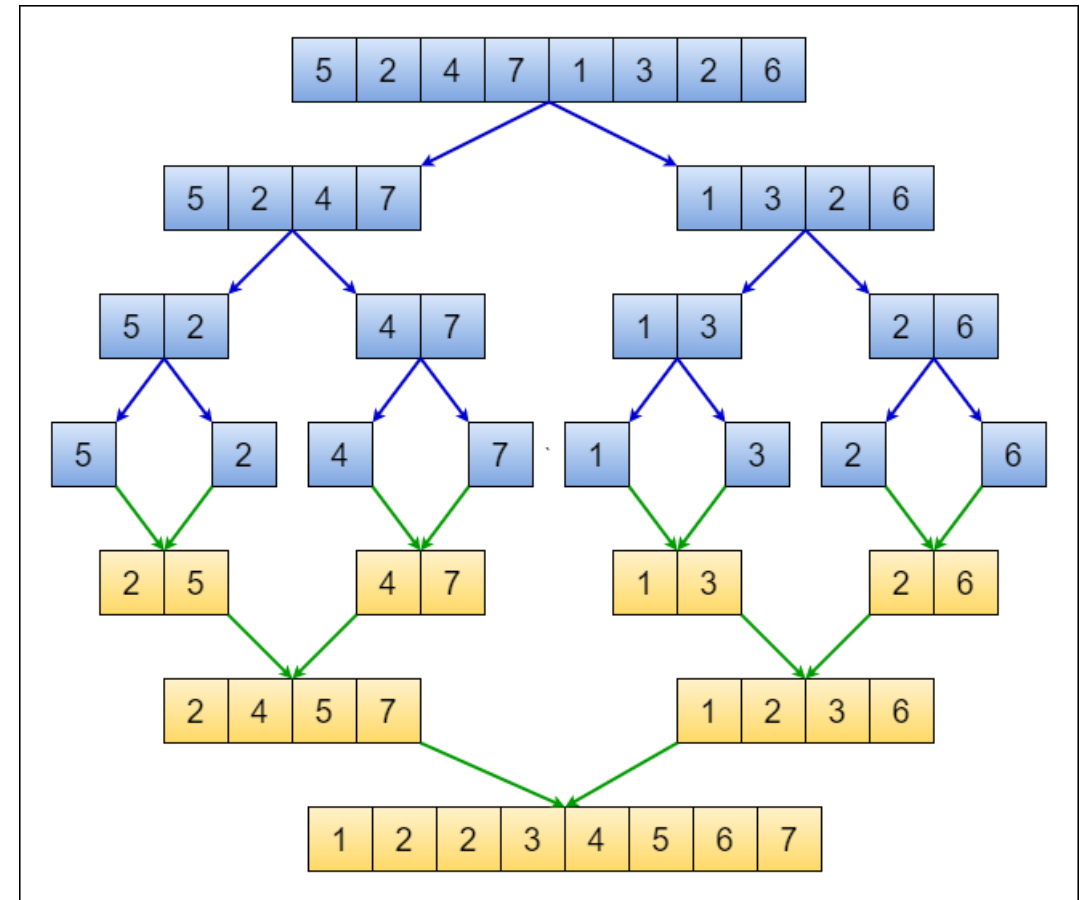
$O(n \log n)$ алгоритми

- **Сортування злиттям** (merge sort) – розділяє вхідний масив на гілки, окремо їх сортує, після чого «зливає» у відсортований масив.
- **Пірамідальне сортування** (heap sort) – використовує структуру даних heap, сортування перебором елементів.
- **Швидке сортування** (quick sort) – обирає один елемент і відбирає ті що менше та більше за нього.
- **Сортування деревом** (tree sort) – утворює бінарне дерево з вхідного масиву, після чого проходить по дереву сортуючи його.

Merge sort

Це алгоритм сортування, в основі якого лежить принцип «Розділяй та володарюй» (Divide and conquer), тобто злиття двох упорядкованих ділянок масиву в одну впорядковану ділянку іншого масиву.

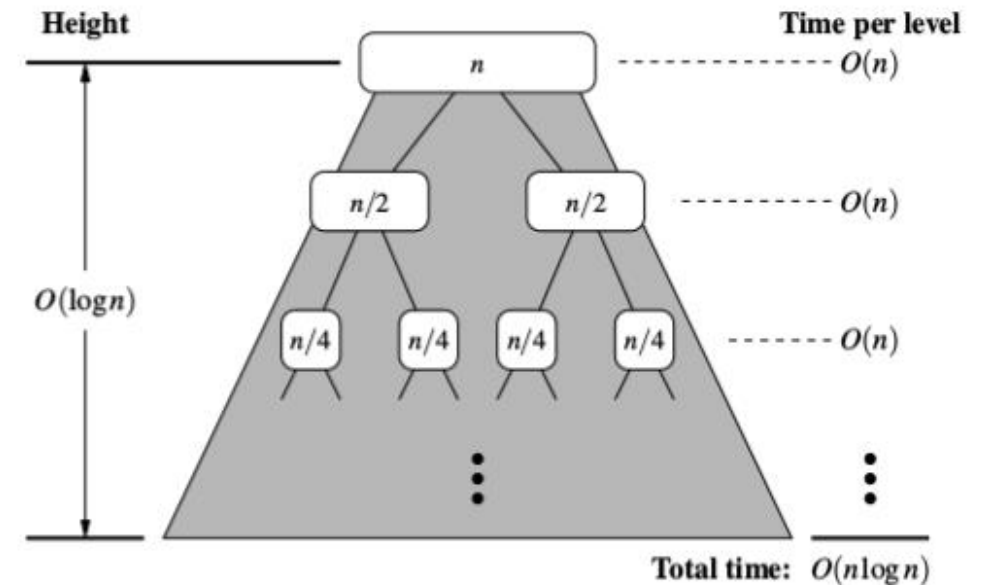
Вхідний масив розділяється на окремі елементи, що зливаються у відсортовану пару, яка в свою чергу зливається з іншою відсортованою парою утворюючи новий відсортований масив.



Merge sort

$(n \log n)$ – в початковий масив входить n елементів. При розбитті початкового масиву маємо $(\log n + 1)$ «рівнів», не враховуючи самого масиву.

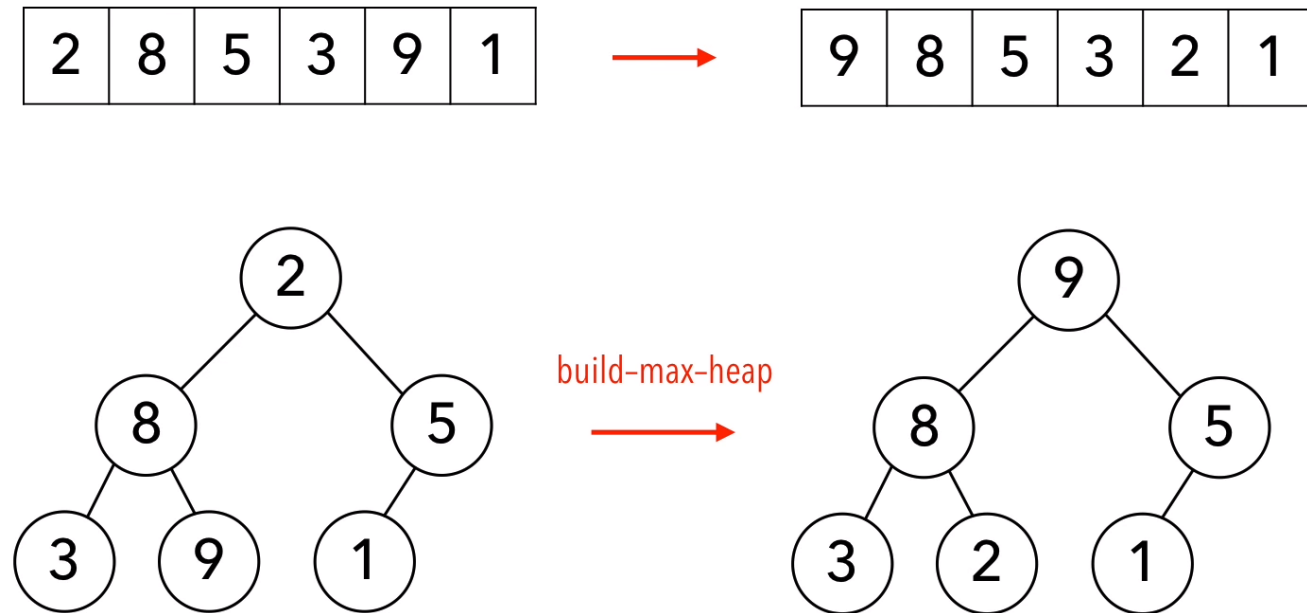
Кількість операцій на кожному рівні $\sim n$. Тому загальна кількість операцій $\sim n \log n$.



Heap sort

Це алгоритм сортування, в основі якого лежить бінарне сортувальне дерево.

Спочатку будується сортувальне дерево. Після чого утворюють max – heap (parent > child).

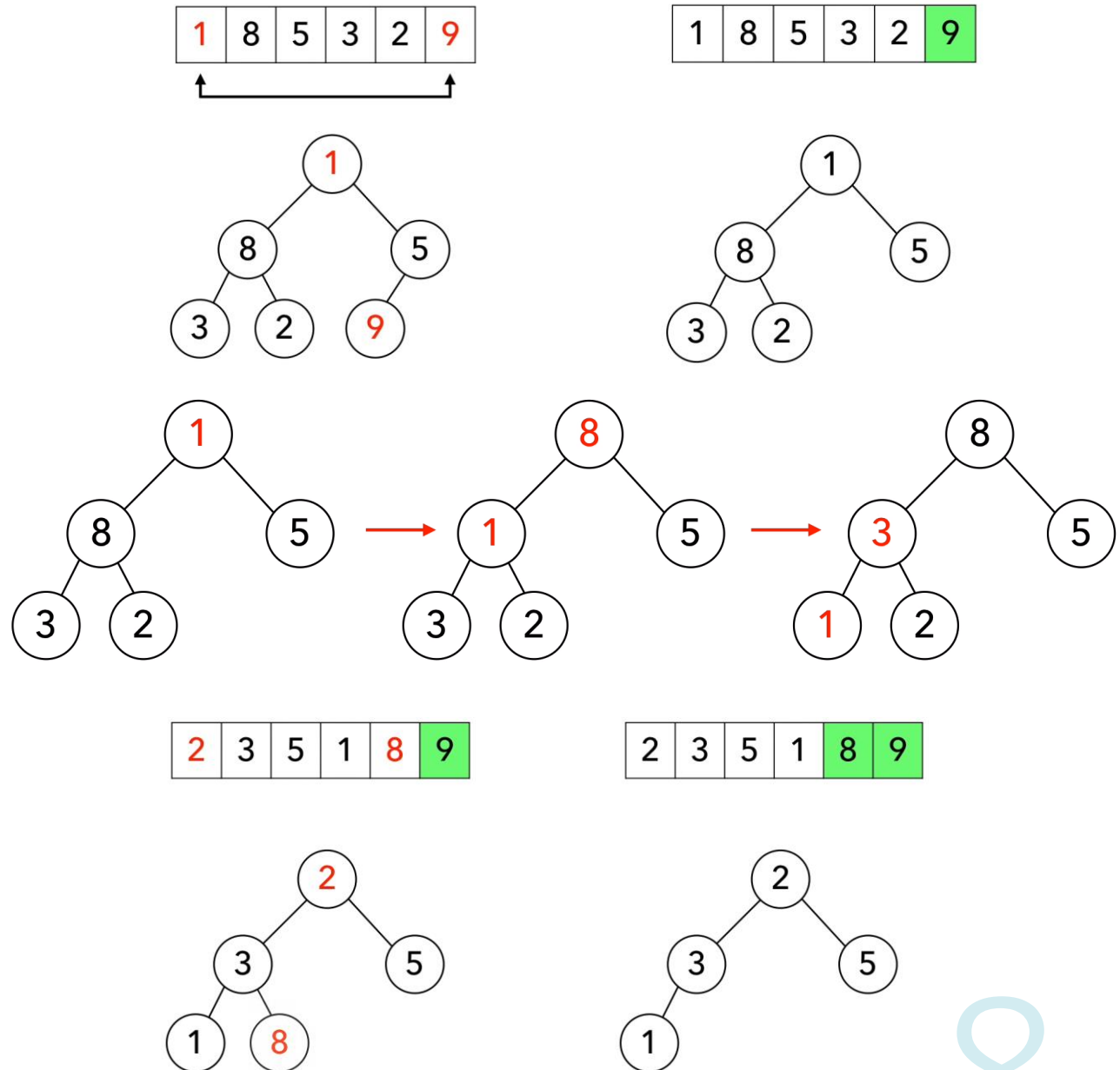


Heap sort

Після цього перший та останній елемент міняють місцями і новий перший елемент видаляється (вважається відсортованим).

Масив знову приводять до купи (heapify) після чого процес повторюється.

Він продовжується поки не залишиться лише один елемент.

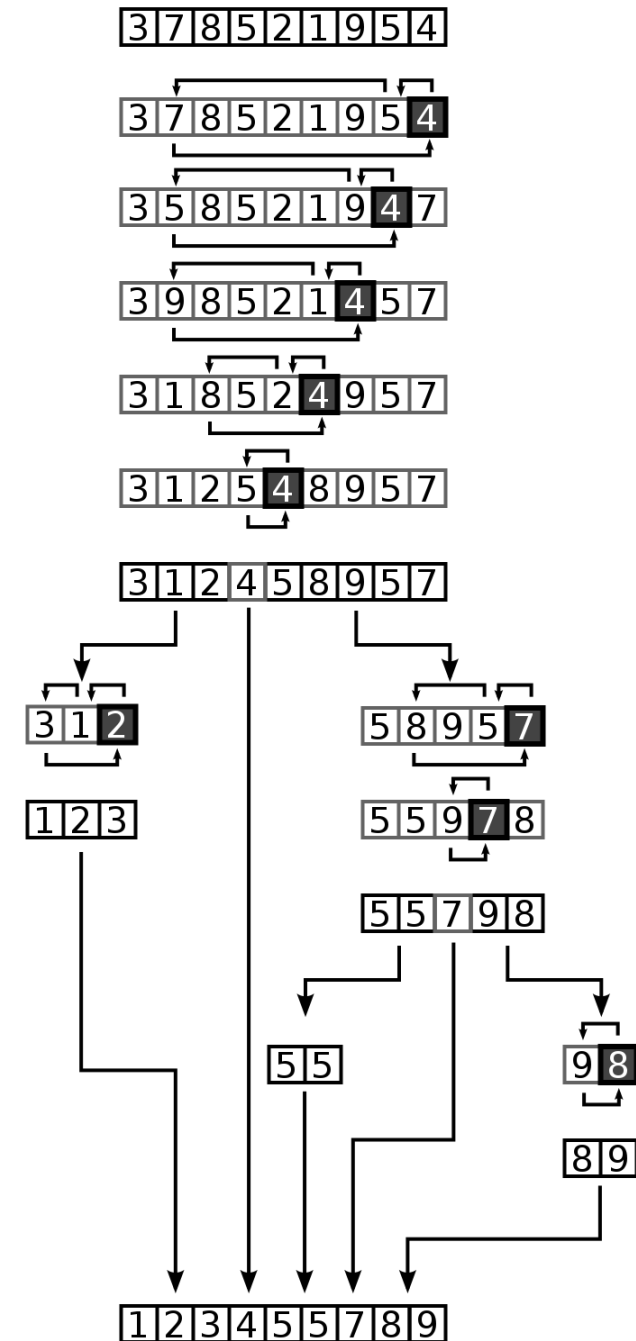


Quick sort

Це алгоритм сортування, що обирає опорний елемент та переносить більші елементи – праворуч, а менші – ліворуч від опорного.

Загальний алгоритм:

- Обирається опорний елемент і переноситься у кінець масиву.
- Елементи переносяться праворуч та ліворуч утворюючи дві купи – «більше» та «менше» за опорний елемент.
- Якщо довжина цих областей більше за 1, то для них рекурсивно відбувається та ж операція.



Tree sort

Це алгоритм сортування, в основі якого лежить побудова двійкового дерева пошуку.

Загальний алгоритм:

- Отримати елементи вхідного масиву.
- Побудувати двійкове дерево вставляючи елементи вхідного масиву в двійкове дерево пошуку (parent > left child, parent < right child).
- Виконати обхід дерева, щоб отримати результуючий масив з впорядкованими елементами.

