

Heap sort 與 Merge sort 的比較

參考老師 moodle 上的表，如下

Sorting Algorithm	Average Time	Best Time	Worst Time	Extra Space (in-place)	Stability
Bubble Sort	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	Stable
Insertion Sort	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	Stable
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	Unstable
Quick Sort	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(n)$	Unstable
Merge Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	Stable
Heap Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(1)$	Unstable
Radix Sort	$O(n)$	$O(n)$	$O(n)$	$O(n)$	Stable
Counting Sort	$O(n)$	$O(n)$	$O(n)$	$O(n+k)$	Stable

以下為來自 <https://blog.csdn.net/microsues/article/details/6107584> 的表格

	500随机整数	5000随机整数	20000随机整数
合并排序	0.3125	1.5625	7.03125
Shell排序	0.3125	1.25	6.875
堆排序	0.46875	2.1875	6.71875
快速排序	0.15625	0.625	2.8125

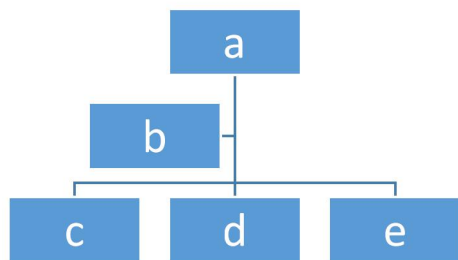
可見兩種 sort 的速度雖然是同一個階層的，但是 merge sort 佔用的空間更大，由於 merge sort 需要分塊後比較的緣故，所以需要佔用更多空間，而 heap sort 則是一個 root 跟各個子節點比較，並不需要佔用多餘空間，故佔用空間較小；

但穩定性上看，則 merge sort 更優，或許是由於分法比較簡單，由兩數之間比較慢慢發展為兩個大 list 比較，比較相對簡單；而 heap sort 的 root 則需時常變更的緣故吧。

另一方面看，merge sort 像組團後比較，而 heap sort 比較像單點比較，直覺上 merge sort 也比 heap sort 來的穩定

從他們的長相上看，merge sort 為{a,b,c,d,e}

Heap sort 為



看上去，merge sort 也給人比較穩定的感覺

總的來說，merge sort 比 heap sort 的優勢是穩定，而 heap sort 比 merge sort 的優勢在於節省空間

所以如果資料量過大的情況下，比較適用 heap sort；
而在資料量較小且追求穩定性的情況下，比較適用 merge sort。

*參考資料

- 1、<https://blog.csdn.net/microsues/article/details/6107584>
- 2、https://docs.google.com/presentation/d/e/2PACX-1vRAGwnUvg6BcXoML5u9f4gO6YKcz0vXf7bDnPho_S7mG5D0SBR78djt91RKUPMxqNfkVlcu3l5WCXPh/pub?start=false&loop=false&delayms=3000&slide=id.g6504c48e6e_0_6
- 3、<https://tingtseng.pixnet.net/blog/post/39924871-algorithm-time-complexity-演算法時間複雜度整理>