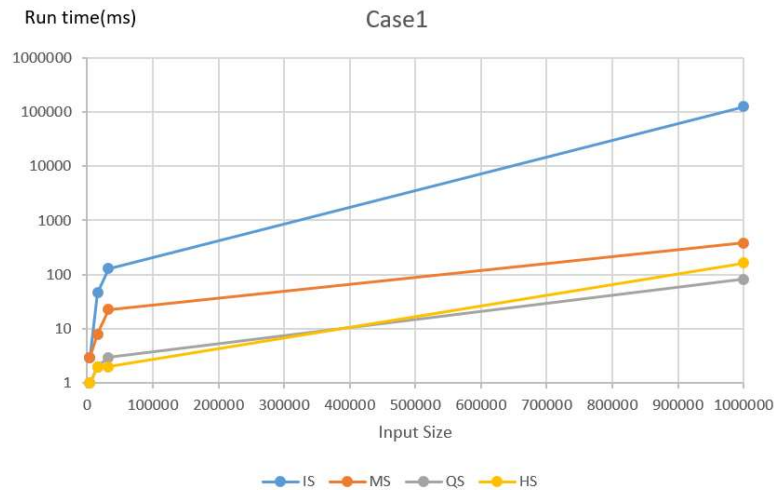


Alg20 PA1 b07901187 宋家齊

Input Size	Is		Ms		Qs		Hs	
	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)
4000.case2	0.001	12500	0.002	12500	0.017	12616	0.001	12500
4000.case3	0.006	12500	0.002	12500	0.013	12524	0.001	12500
4000.case1	0.003	12500	0.003	12500	0.001	12500	0.001	12500
16000.case2	0.001	12648	0.007	12648	0.158	13324	0.002	12648
16000.case3	0.066	12648	0.007	12648	0.148	12944	0.002	12648
16000.case1	0.046	12648	0.008	12648	0.002	12648	0.002	12648
32000.case2	0.001	12648	0.013	12836	0.459	14076	0.003	12648
32000.case3	0.285	12648	0.013	12836	0.397	13316	0.003	12648
32000.case1	0.129	12648	0.023	12836	0.003	12648	0.002	12648
1000000.case2	0.002	18668	0.301	22760	446.93	63416	0.093	18668
1000000.case3	251.645	18668	0.323	22760	270.545	33832	0.093	18668
1000000.case1	125.872	18668	0.382	22760	0.082	18668	0.165	18668

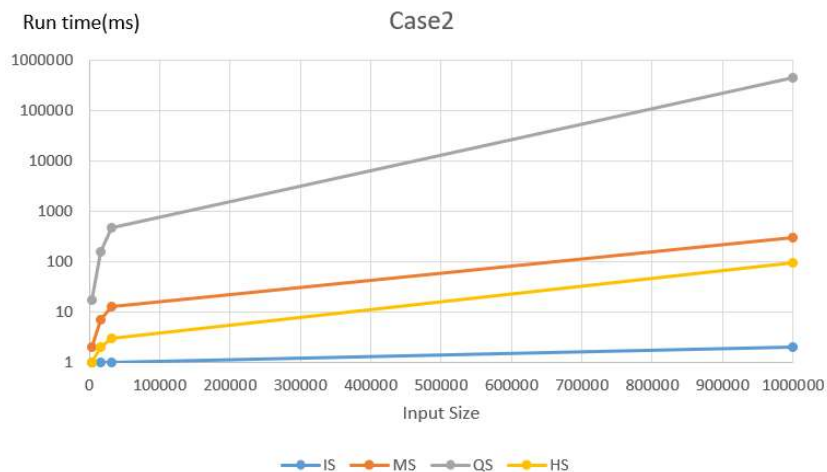
Case 1 (random order):

可看出 Quick sort 在隨機分布的數列跑的最快，而 Insertion sort 為 $O(n^2)$ ，所以跑的最慢。(取對數觀察)



Case2(Sorted array):

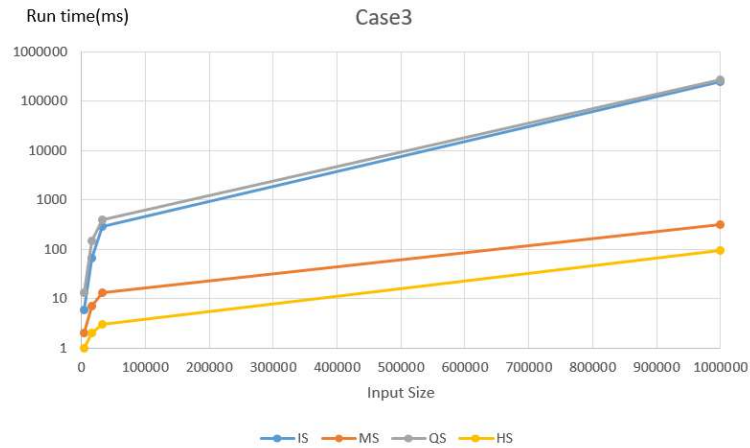
可看出 Insertion sort 跑得最快，因為 IS 只需要跑過一次數列，不需要做任何交換數字動作，而 QS 取 pivot 的方式是選 subarray 中最後一個數，因此跑完整個 partition 會花很多時間。



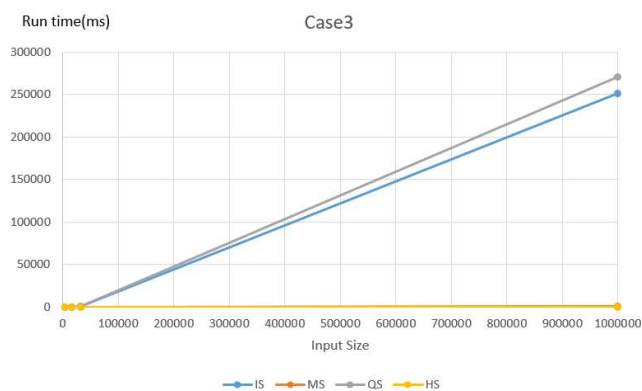
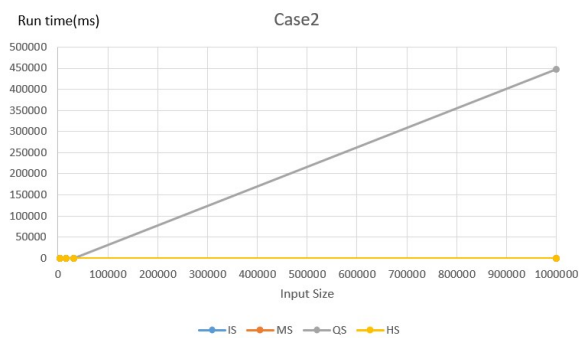
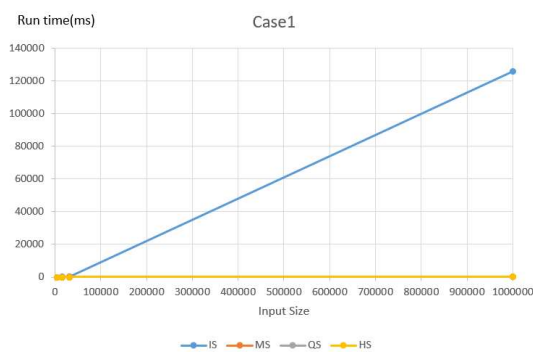
Alg20 PA1 b07901187 宋家齊

Case3 (reverse order):

可看出 IS 和 QS 花了最多時間，因為 IS 的時間複雜度是 $O(n^2)$ ，在 Case3 下是 worst case. 同 case2，QS 選 pivot 的位置是最後一個數字，因此不管是 increasing order 和 reverse order 都會花較久時間。

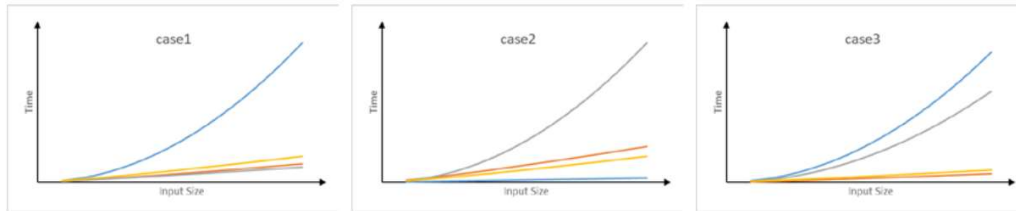


不取 log:



In the document,

4. In your report, draw figures to show the growth of running time as a function of input size and try to analyze the curve (as the following example, where each curve represents an algorithm.)



The corresponding cases are:

Case1: random order

Case2: increasing order

Case3: reverse order

Insertion sort: Blue

Merge sort: red

Quick sort: gray

Heap sort: orange