Algorithms:

Programming Assignment 1 Report

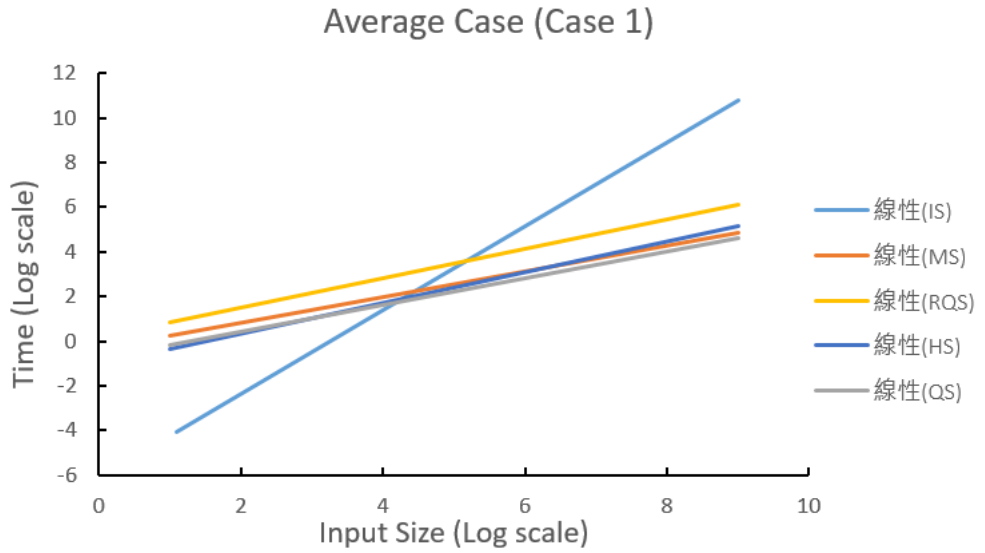
B10901006 電機三 鄒昀樺

1. Performance of Algorithms with Different Input Sizes

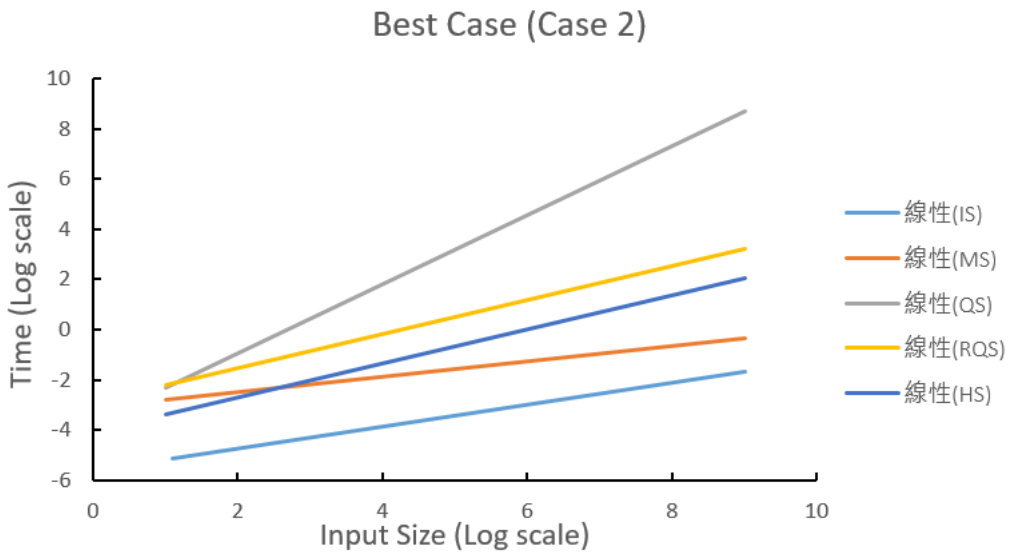
- Data is run on the EDA union lab machines

IS		MS		QS		RQS		HS	
CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)	CPU time (s)	Memory (KB)
0.14m	5904	1.679m	6040	11.391m	5904	9.322m	5904	0.481m	5904
10.184m	5904	1.614m	6040	10.028m	5904	11.413m	5904	0.699m	5904
6.179m	5904	2.902m	6040	0.903m	5904	11.47m	5904	0.951m	5904
0.132m	6056	3.19m	6056	86.798m	6056	25.611m	6056	2.57m	6056
82.753m	6056	4.031m	6056	64.333m	6304	24.445m	6056	1.486m	6056
43.837m	6056	5.585m	6056	2.81m	6056	25.723m	6056	1.361m	6056
0.155m	6188	5.527m	6316	325.608m	6188	47.092m	6188	3.381m	6188
329.714m	6188	4.418m	6316	242.283m	6736	44.12m	6188	3.032m	6188
162.012m	6188	6.95m	6316	2.881m	6188	47.276m	6188	2.938m	6188
1.254m	12144	146.615m	16236	294894m	12144	1400.9m	12144	78.923m	12144
318976m	12144	157.877m	16236	198141m	23996	1383.79m	12144	76.683m	12144
158817m	12144	232.16m	16236	92.533m	12144	1438.83m	12144	141.302m	12144

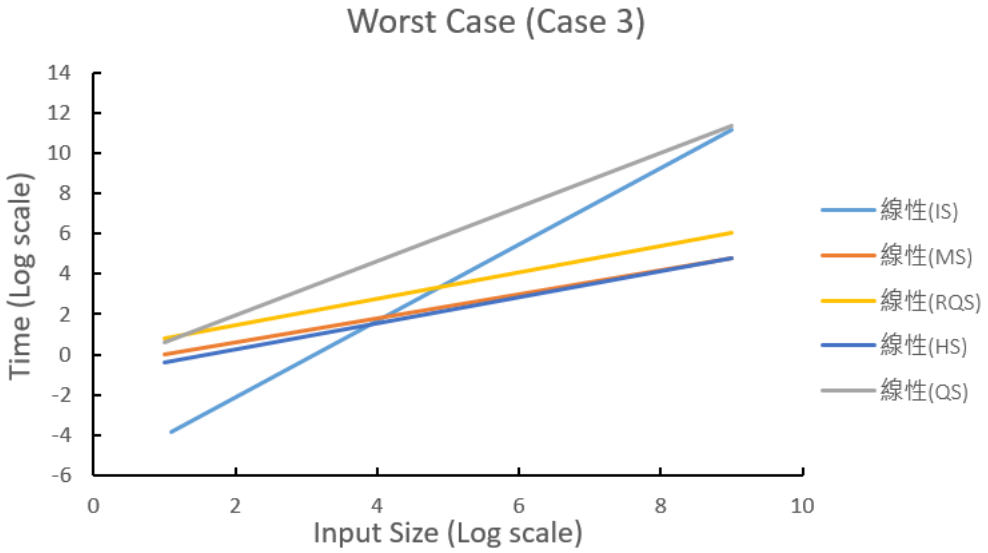
2. Growth of Run Time



In the average case, insertion sort has the least performance while the other four algorithms have similar performance. The slope of the insertion sort is about 1.88 ≈ 2. This corresponds to the run time conclusion of insertion sort made in class, which is $O(n^2)$. Other lines have slope about 0.6 to 0.7, corresponding to $O(n \cdot \lg n)$ after applying log.



As for the best case, Quicksort has shown the worst performance as the input size grows. Its slope is about 1.4, while others' are between 0.3-0.6. The possible reason behind this could be that Quicksort always uses the first element as its pivot. For the best case which the elements have already been arranged, the array would always be cut into 1 vs. (n-1). Therefore, the run time would be $O(n^2)$, which is 2 (1.4 in this case) of the slope in the graph. Randomized Quick Sort has solved the problem by choosing the pivot randomly, resulting run time of $O(n \cdot \lg n)$. Insertion sort has the best performance because the array has already been sorted, so no insertion is needed in the process. As for slopes of Mergesort and Heapsort (which is under 1), they are simply the outcome of $O(n \cdot \lg n)$.



Finally in the worst case, both insertion sort and Quicksort have the worst performance. The reason for Quicksort is same as the above case 2, while the reason for insertion sort is same as the above case 1. Both are from the run time $O(n^2)$. Others have the same performance as the outcome of $O(n \cdot \lg n)$.