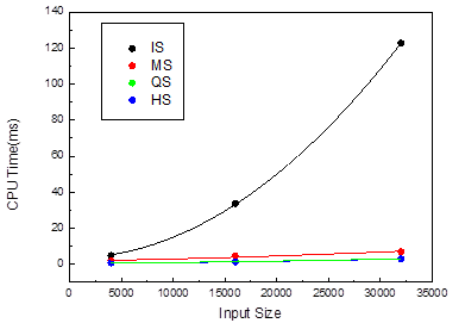
PA1 Homework 陳韋辰 r08222017

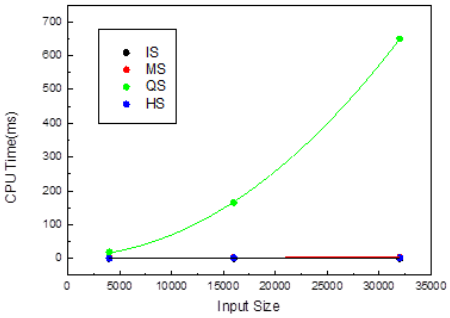
1. Complexity Analysis

Input size	IS		MS		QS		HS	
	CPU time (ms)	Memory (KB)	CPU time (ms)	Memory (KB)	CPU time (ms)	Memory (KB)	CPU time (ms)	Memory (KB)
4000.case2	0.103	5892	1.826	6028	19.448	5960	0.713	5880
4000.case3	10.133	5892	1.663	6028	14.367	5892	0.709	5880
4000.case1	4.969	5892	1.914	6028	0.756	5892	0.821	5880
16000.case2	0.089	6044	2.898	6200	165.079	6672	2.545	6032
16000.case3	64.352	6044	2.94	6200	131.583	6292	1.845	6032
16000.case1	33.711	6044	4.638	6200	1.593	6044	1.357	6032
32000.case2	0.098	6176	3.68	6368	649.894	7488	2.038	6164
32000.case3	247.033	6176	4.134	6368	483.428	6728	1.757	6164
32000.case1	122.643	6176	7.059	6368	3.079	6176	3.053	6164
1000000.case2	1.165	12132	115.151	21808	629106	56828	75.386	12120
1000000.case3	251537	12132	121.619	21808	331189	27236	72.921	12120
1000000.case1	123847	12132	191.463	21808	80.568	12132	132.814	12120

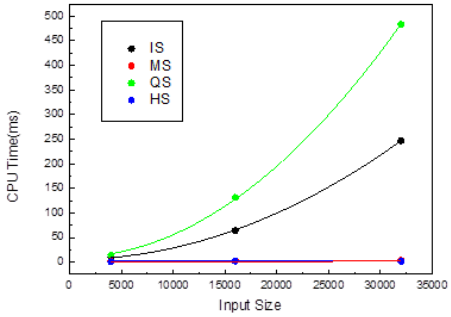
Case1



Case2



Case3



對 IS 來說，反序(case3)是 worst case，因為每個值都要比完，可用n^2擬合，而相

反正序(case2)是 best case，可用 n 擬合，對於 random 的數據(case1)也符合 average case n^2的擬合。

對 MS 來說，反序(case3)和正序(case2)較快，推測因為相比於 random 的數據(case1)，會快將單一邊 array 的數字排掉，加快運算速度，不過三者都比 upper bound nlgn 小。

對 QS 來說反序(case3)和正序(case2)是 worst case，因為每一次都用最大或最小去比，無法切成兩部分比較，失去 QS 的優勢，可用n^2擬合，而 random 的數據(case1)可用 average time 的 nlgn 擬合

對 HS 來說沒有出現特別的 best 或 worst case，不過時間都比 upper bound nlgn 小。

2. Data Structure

IS: array 組成，將還沒排列好的 element 從大到小跟排列好的 element 比較，放入正確位置，因此排列好的部分會 incremental 的增加，最後完成 sorting。

MS: array 組成，利用 divide and conquer，把 elements 分成兩部分，利用 recurrence 分到最小，兩兩放到另外的 array 並 merge 回原來的 array，完成 sorting。

QS: array 組成，利用 divide and conquer，把 elements 分成大小於隨機數的兩邊，再利用 recurrence 完成 sorting。

HS: heap data structure 是由 binary tree 構成的 array，每一個節點都是一個 element，除了最下層其他都要填滿。任一個 element 都會比底下的分支 element 大，重複取 root 的值，並將下層的 element 重新 max heapify，即可完成 sorting。

3. Conclusion

IS 跟 QS 在運算上比較不穩定，遇到 worst case 會出現n^2的 Complexity，相較之下 MS 跟 HS 較穩定，都可以維持在 nlgn 的 Complexity，而 HS 又更快一些，推測是 in-place 的設計加上 binary tree 的 data structure，不用每次都將 tree 跑完，節省很多時間。在記憶體方面 IS 跟 HS 發揮 in-place 的優勢，使用最少記憶體，而 MS 則因為需要使用額外的 array 而增加了使用量，QS 的使用量是變動的，推測是因為 return 的中間值的數量每個 case 是不一樣的，造成記憶體的使用量不一樣。

4. References

https://www.upgrad.com/blog/sorting-in-data-structure-with-examples/