PA1 Report

B09901066 謝承修

1. The run time and memory of four versions of different input sizes.

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.case1	7.022	5904	1.708	6040	0.867	5904	0.696	5904
4000.case2	0.122	5904	0.574	6040	0.774	5904	0.699	5904
4000.case3	10.522	5904	0.838	6040	0.842	5904	0.808	5904
16000.case1	37.808	6056	4.074	6056	2.254	6056	1.868	6056
16000.case2	0.097	6056	1.242	6056	1.974	6056	1.309	6056
16000.case3	72.168	6056	1.862	6056	1.918	6056	1.927	6056
32000.case1	147.185	6188	7.749	6188	2.929	6188	3.422	6188
32000.case2	0.11	6188	2.503	6188	1.854	6188	3.282	6188
32000.case3	274.917	6188	2.732	6188	2.874	6188	3.18	6188
1000000.case1	142997	12144	147.357	15956	110.202	12144	152.088	12144
1000000.case2	1.011	12144	53.109	15956	59.416	12144	81.175	12144
1000000.case3	289491	12144	65.267	15956	62.39	12144	76.078	12144

We can see that for the same input size, every sorter takes the same memory, no matter which case is.

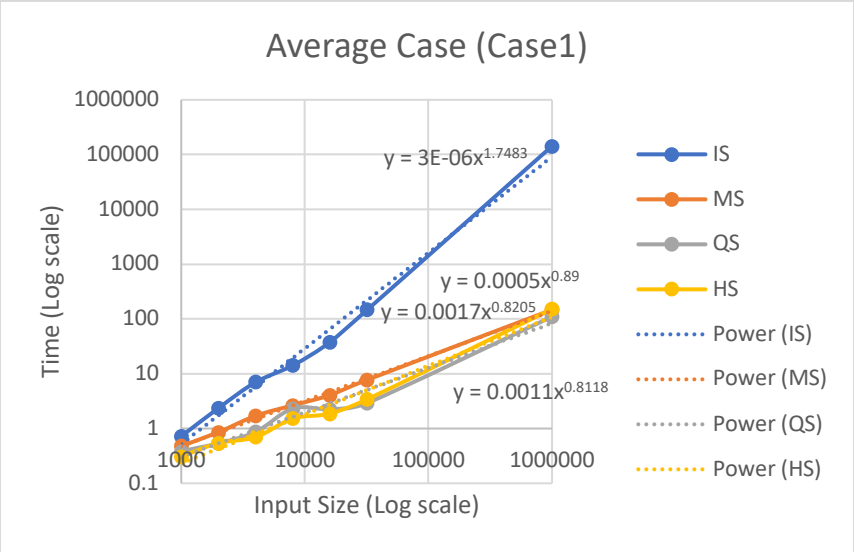
And four sorters take almost the same memory for the same input size, so we can conclude that the performance of them nearly depend on how much time they take.

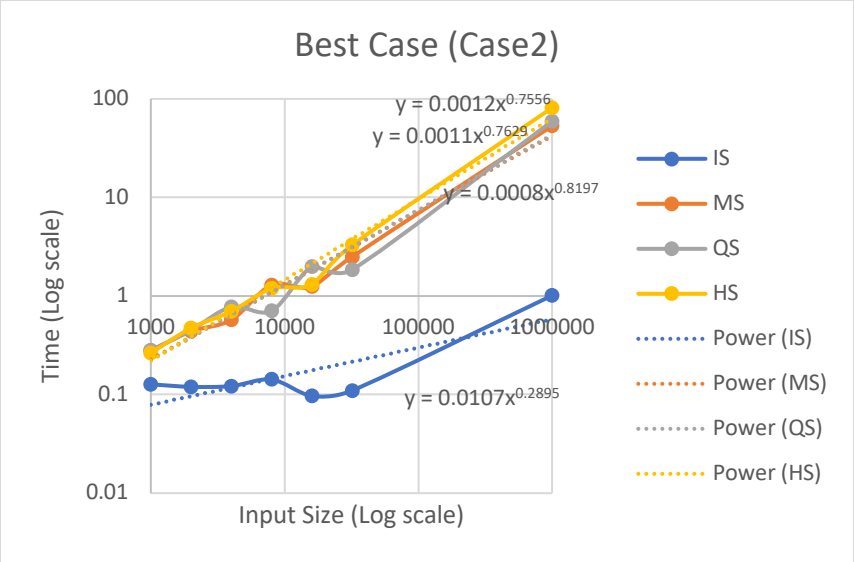
To compare the run time they take, I make the following table, using seven different input sizes.

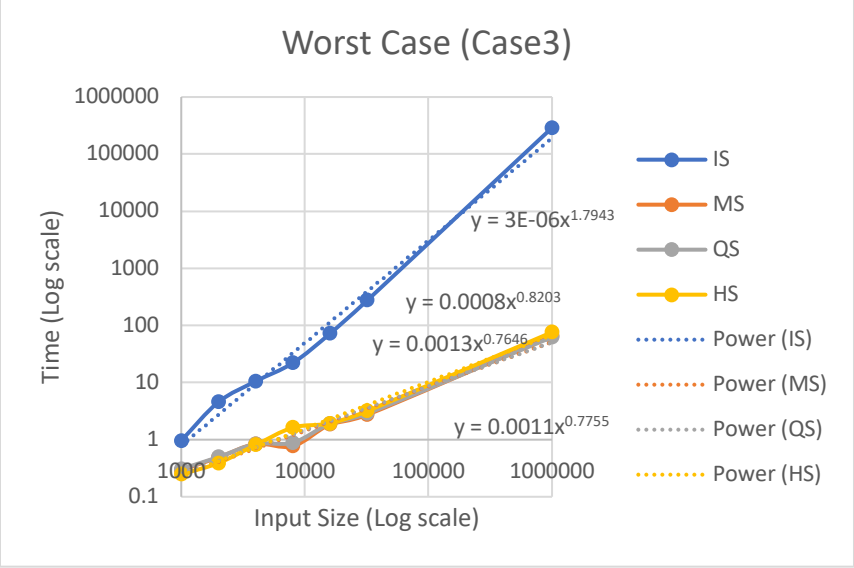
Also, for each case, I draw figures to show the growth of run time as a function of input size.

input size	case1	IS	MS	QS	HS	case2	IS	MS	QS	HS	case3	IS	MS	QS	HS
1000		0.72	0.478	0.384	0.309		0.127	0.282	0.279	0.264		0.963	0.293	0.307	0.248
2000		2.361	0.859	0.539	0.544		0.12	0.438	0.45	0.473		4.56	0.489	0.496	0.389
4000		7.022	1.708	0.867	0.696		0.122	0.574	0.774	0.699		10.522	0.838	0.842	0.808
8000		14.201	2.652	2.326	1.52		0.142	1.29	0.709	1.193		22.008	0.769	0.886	1.63
16000		37.808	4.074	2.254	1.868		0.097	1.242	1.974	1.309		72.168	1.862	1.918	1.927
32000		147.185	7.749	2.929	3.422		0.11	2.503	1.854	3.282		274.917	2.732	2.874	3.18
1000000		142997	147.357	110.202	152.088		1.011	53.109	59.416	81.175		289491	65.267	62.39	76.078

2. The growth of run time







Analysis:

The trendline is of the form $y = ax^b$, take logarithm for both sizes, we get $\log y = \log a + b \log x$, which tells us that the slope of the trendline ($\log y - \log x$) is b, and a means the intercept.

We can see that in the average case and worst case, the MS, QS, HS, have the slope of about 0.8, and the slope of the IS is about 1.75, which is much larger than that of other three sorters. As for the best case, however, IS gives the best performance, which only has the slope of about 0.3, and other three sorters still have the slope about 0.8.

There, however, is an important issue that the slope doesn't follow the formula in the textbook, where MS, QS, HS do not grow with the input size with the rate of $n\log n$, and IS is O(n). To explain that, I asked TA and search for this problem, and I concluded two reasons. First, we cannot ignore the operations of the machine, since there are more complicated about low-level programming language. Second, the input size is not big enough. I think the less of input size, the more deviation there exists. To sum up, we cannot conclude the time complexity of this PA, but surely it shows us that the different performance of the four sorters under the different input sizes and cases.