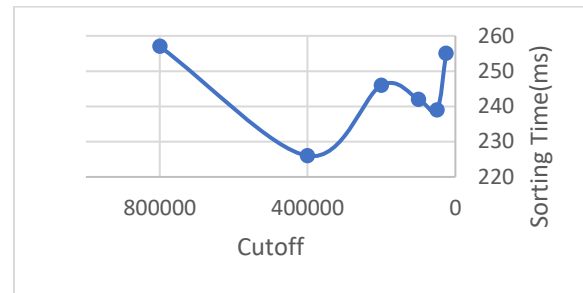
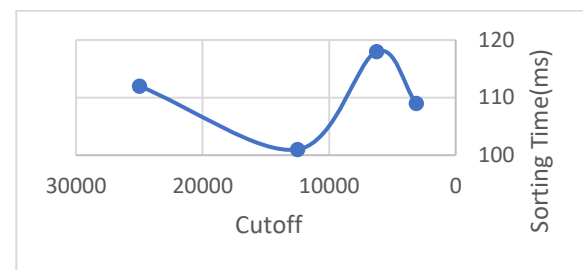


REPORT

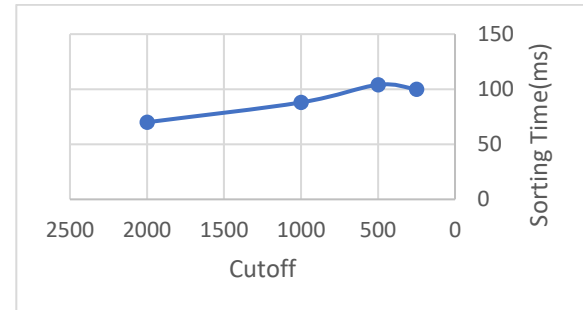
Array Size	Cutoff	Sorting time
800000	800000	257
	400000	226
	200000	246
	100000	242
	50000	239
	25000	255



Array Size	Cutoff	Sorting time
25000	25000	112
	12500	101
	6250	118
	3125	109



Array Size	Cutoff	Sorting time
2000	2000	70
	1000	88
	500	104
	250	100



Observations

- Threads running in parallel are dependent on the core architecture of the operating system. Since the number of cores on my machine are equal to 4, the number of threads running in parallel can be 3.
- The number of cores of the system can be found using the following command-
Runtime.getRuntime().availableProcessors()

For cases when the array size is equal to the cutoff there is only 1 thread running. There will be 3 threads running when array size is 800000 and cutoff is 400000. Similarly, for all the cases I have checked for instances

1) when array size is equal to cutoff

2) when we have 3 threads in parallel

3) When we have more than 3 threads and not running in parallel.

- For smaller arrays, we observe that when there is only one thread running the processing time is less compared to the scenarios when there is more than one thread. This happens because spawning of threads also takes extra time and in such cases the process is completed quickly with one thread only.
- For Large arrays, we observe that since the number of cores of the machine is 4, only 3 threads can run in parallel and in such cases the processing time is relatively low compared to other cases when there is only one thread or more than 3.