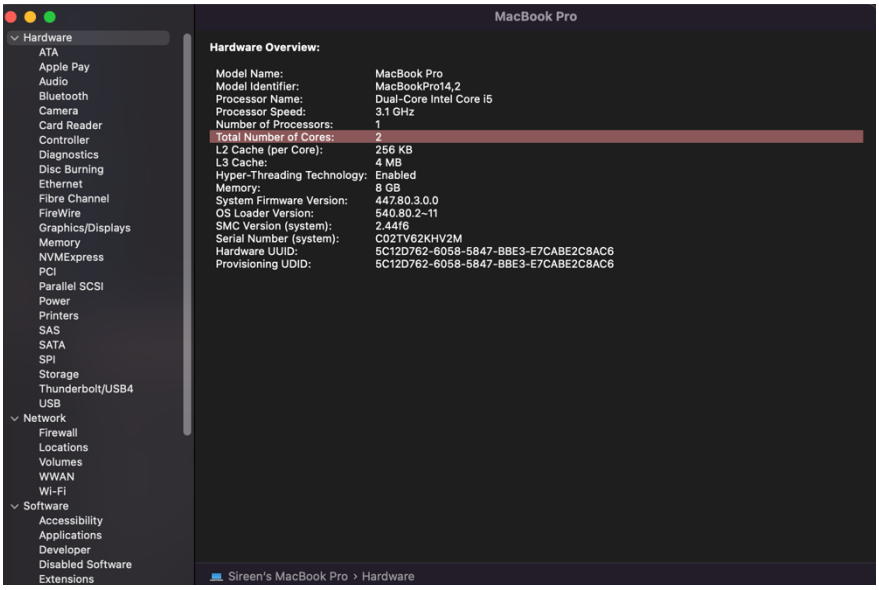
EE463 HW4

Operating Systems

Sereen Omar Bahdad
ID-1806392
Section A

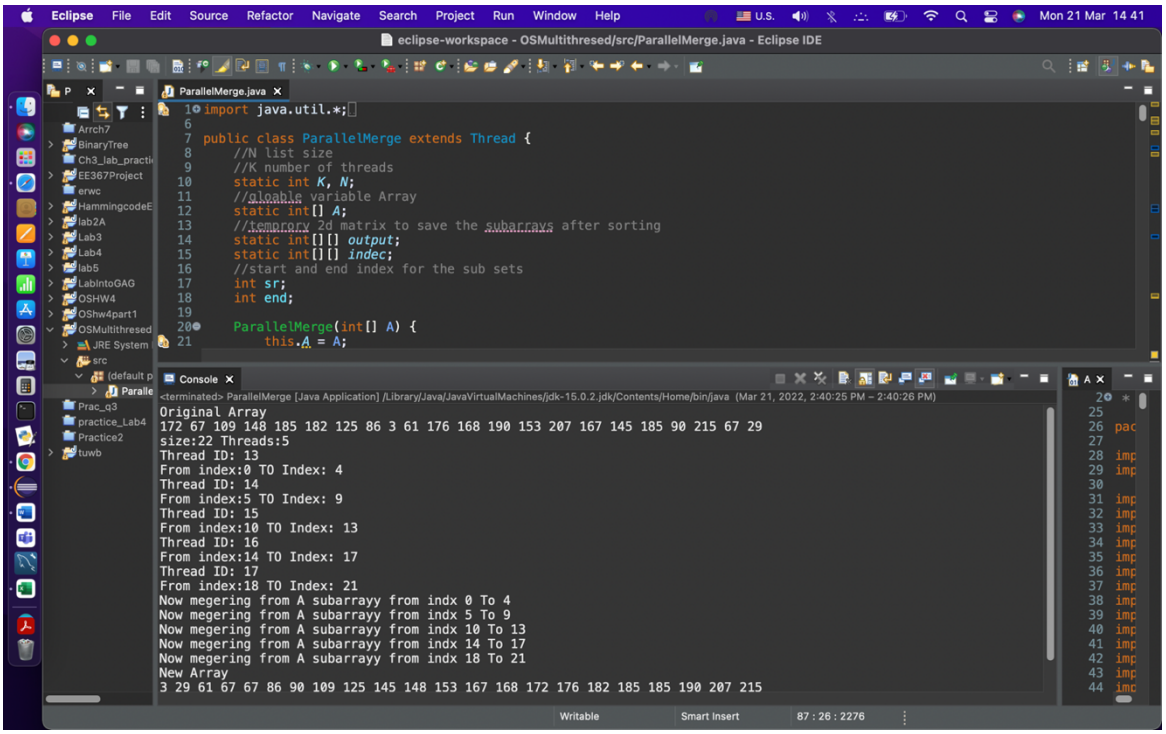
In this HW a multithreaded program was developed to sort an 1D array using the theory of merge sorting. Throughout this report the HW questions were answered, and the results obtained in each case was recoded and analyzed.

1. Available device hardware information



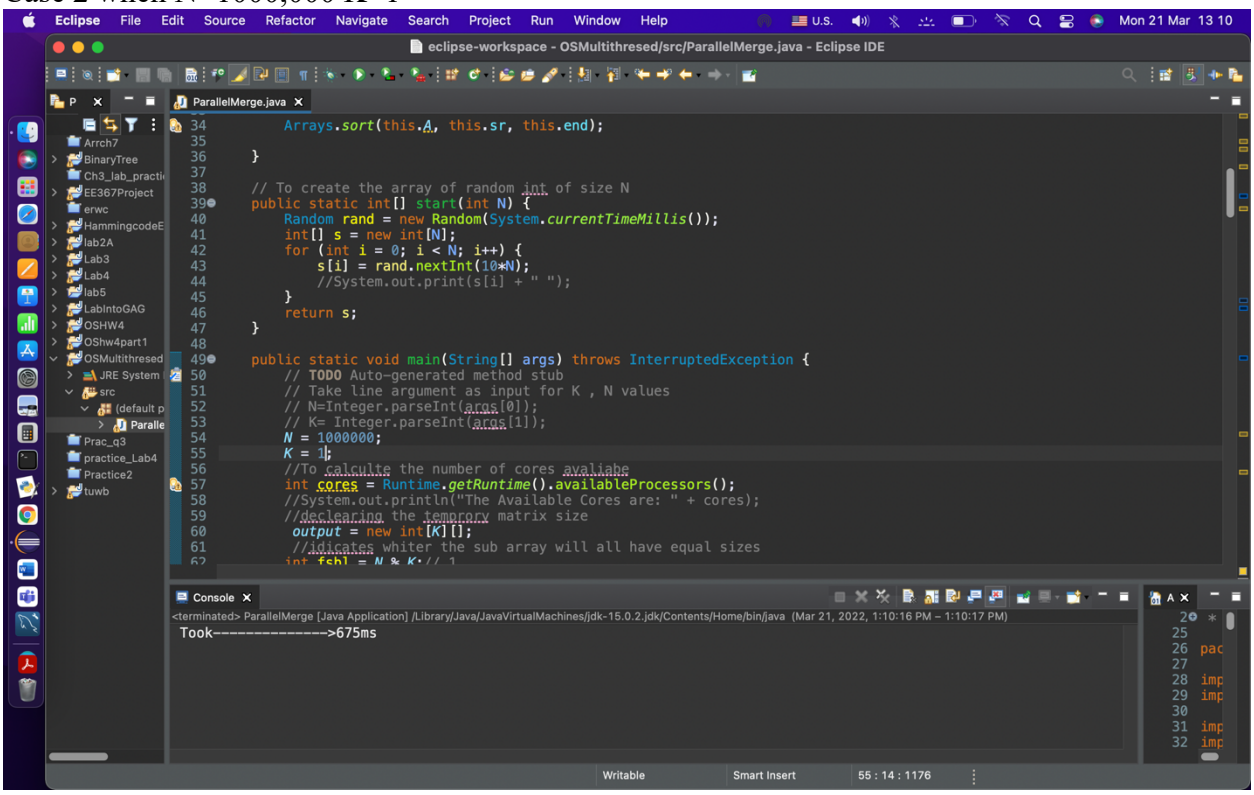
Dual processors and each have two cores this means total of 4 cores, and it was calculated using the java API functions

Case 1 when N=24, K=5

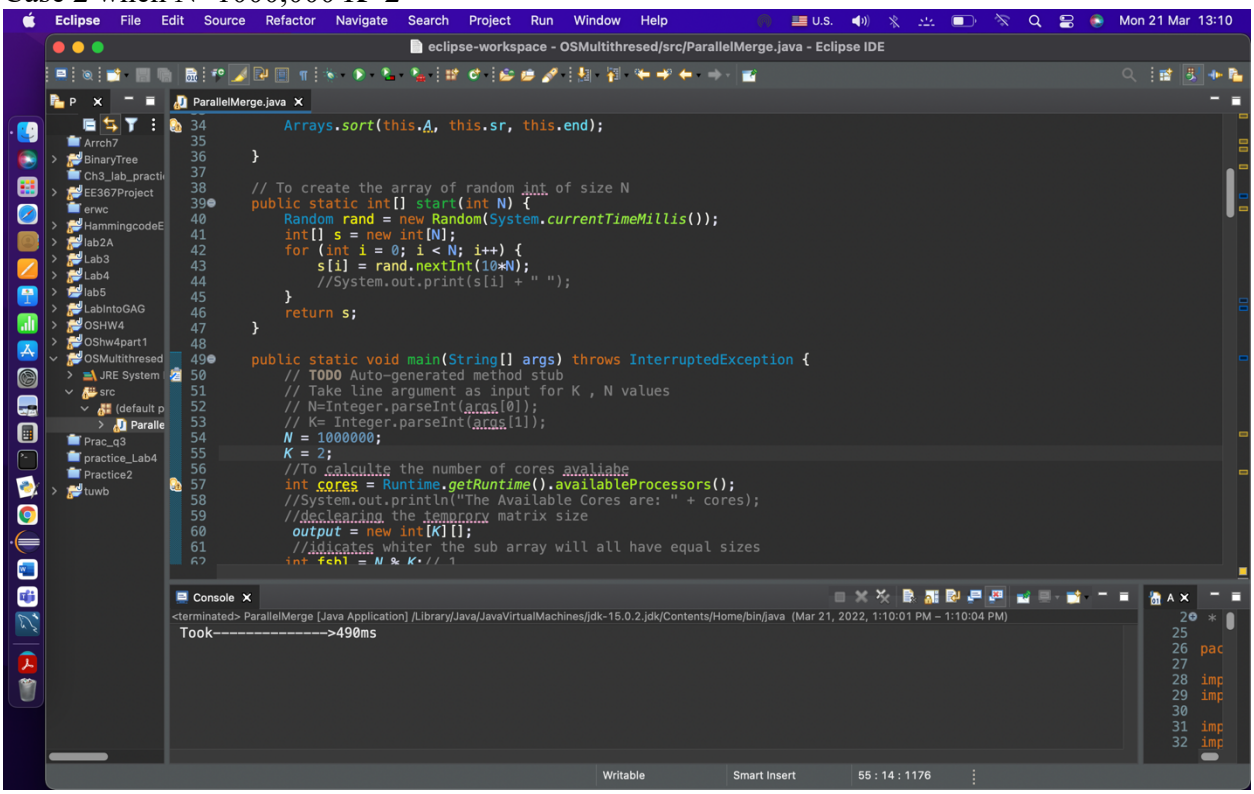


To Record the time taken please comment all the output messages in the code

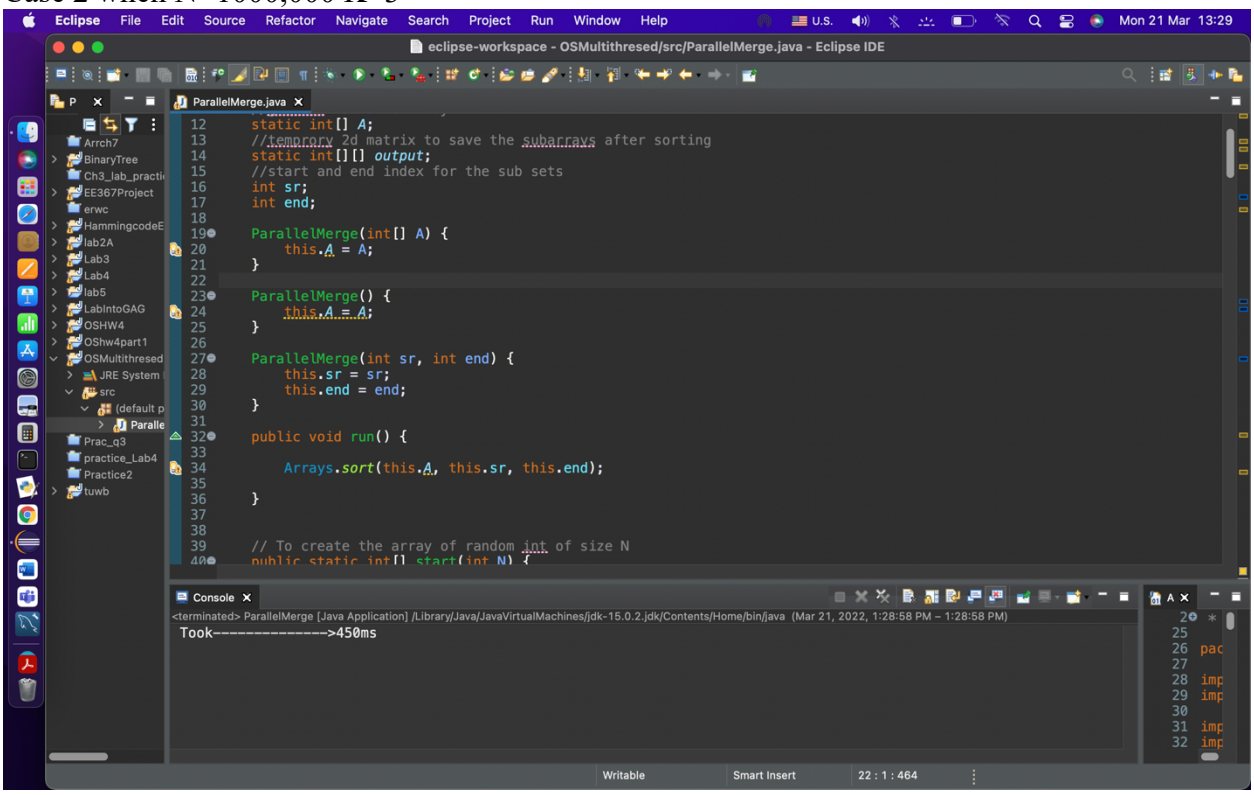
Case 2 when N=1000,000 K=1



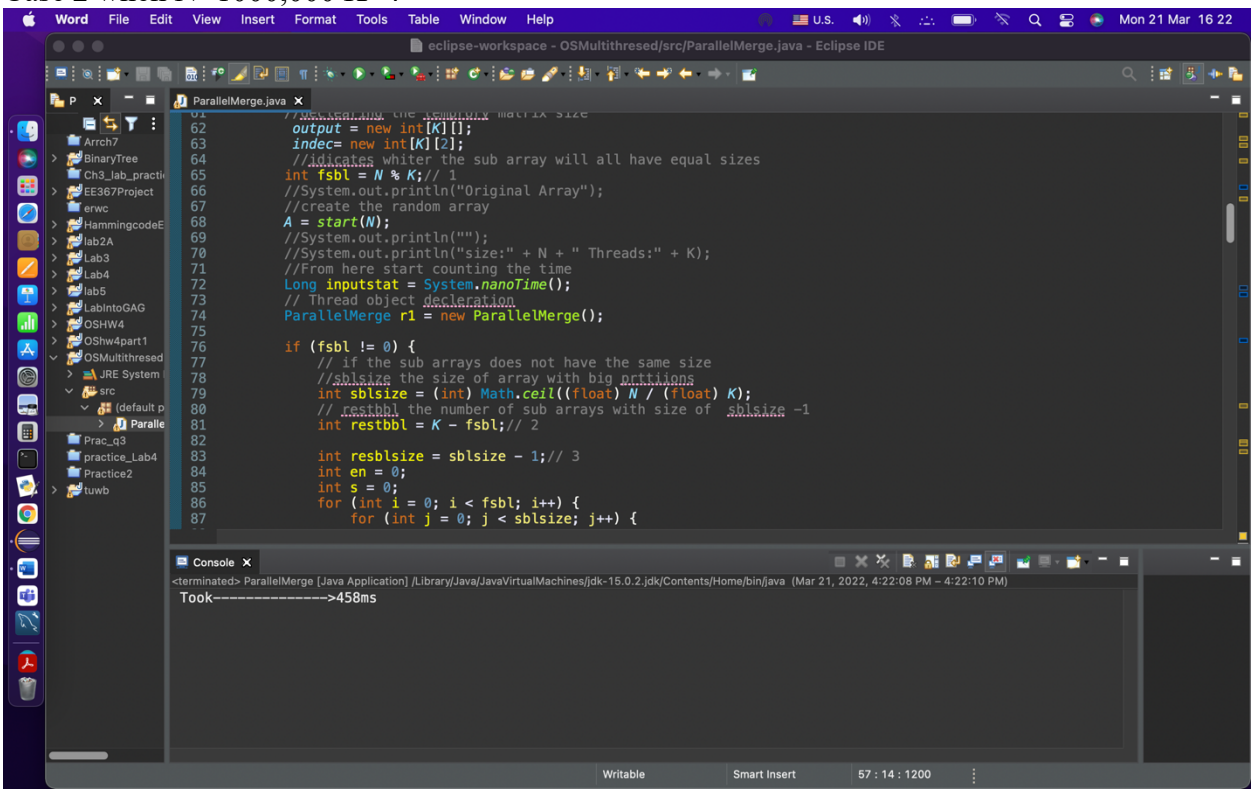
Case 2 when N=1000,000 K=2



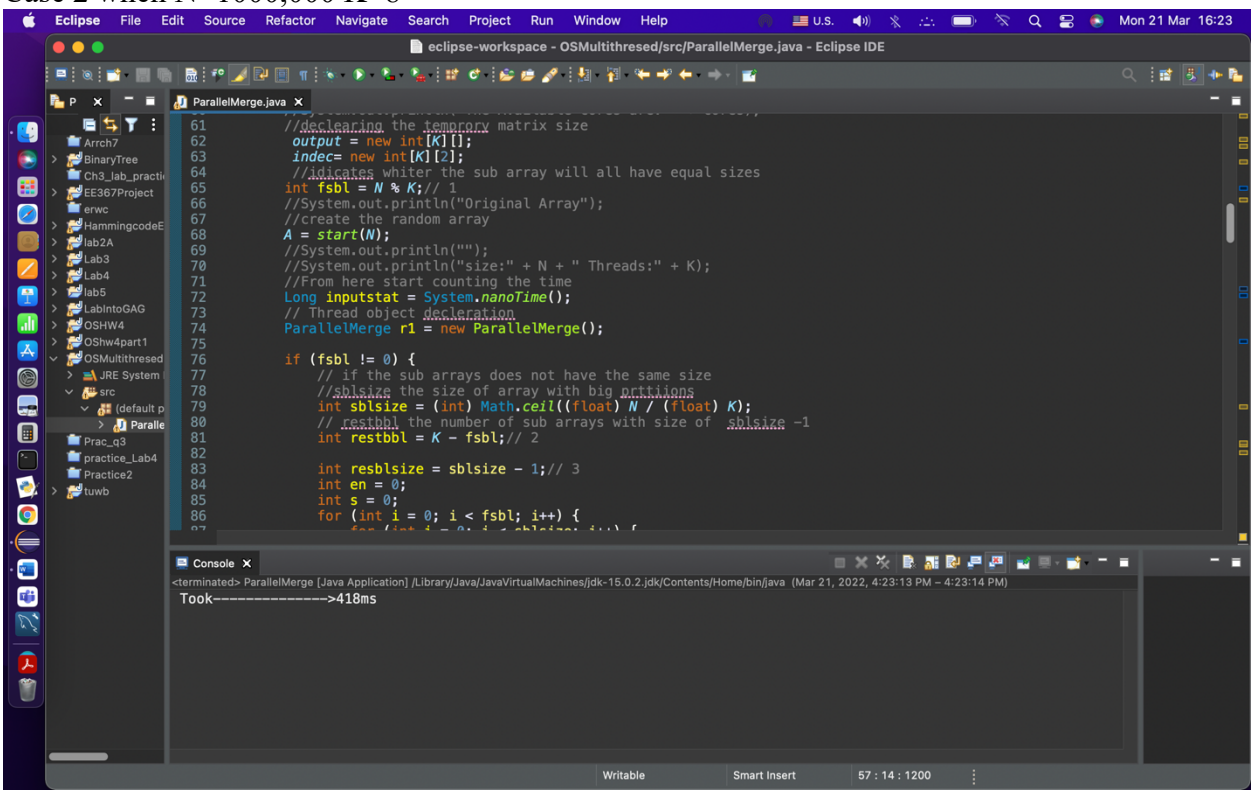
Case 2 when N=1000,000 K=3



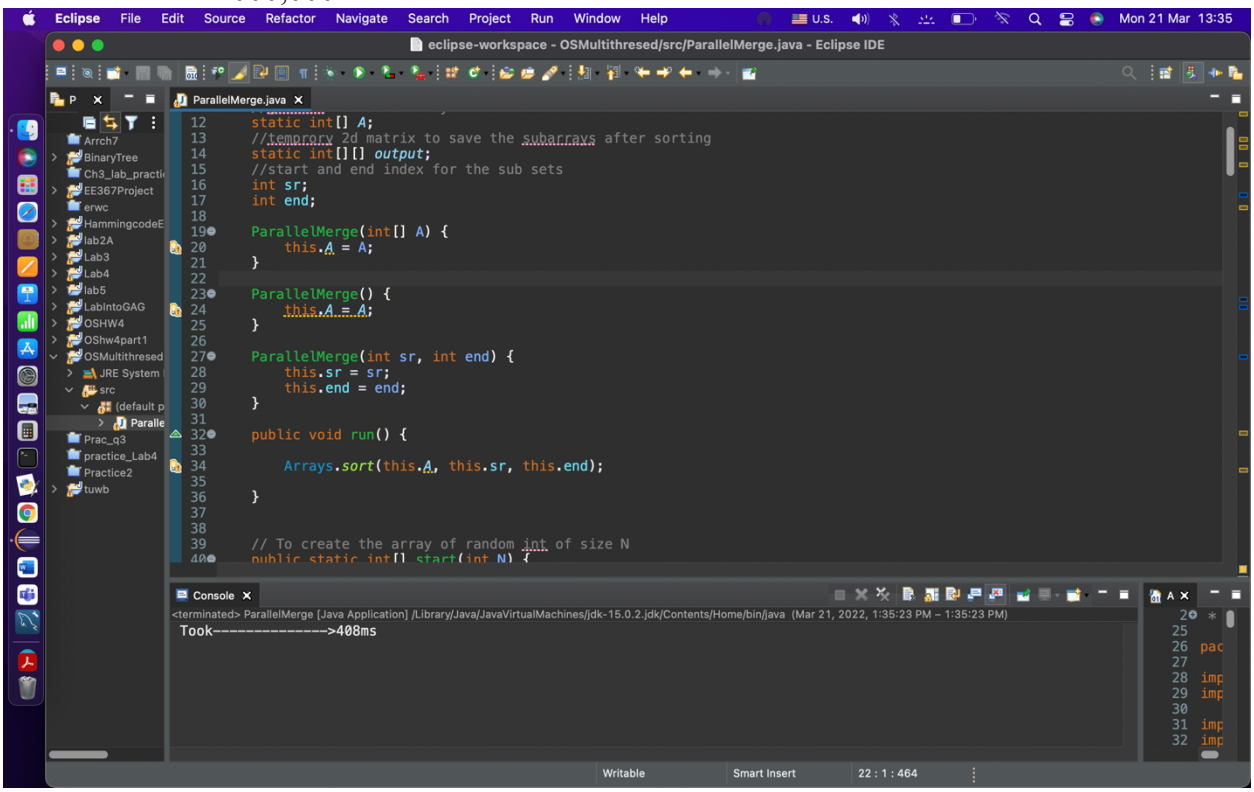
Case 2 when N=1000,000 K=4



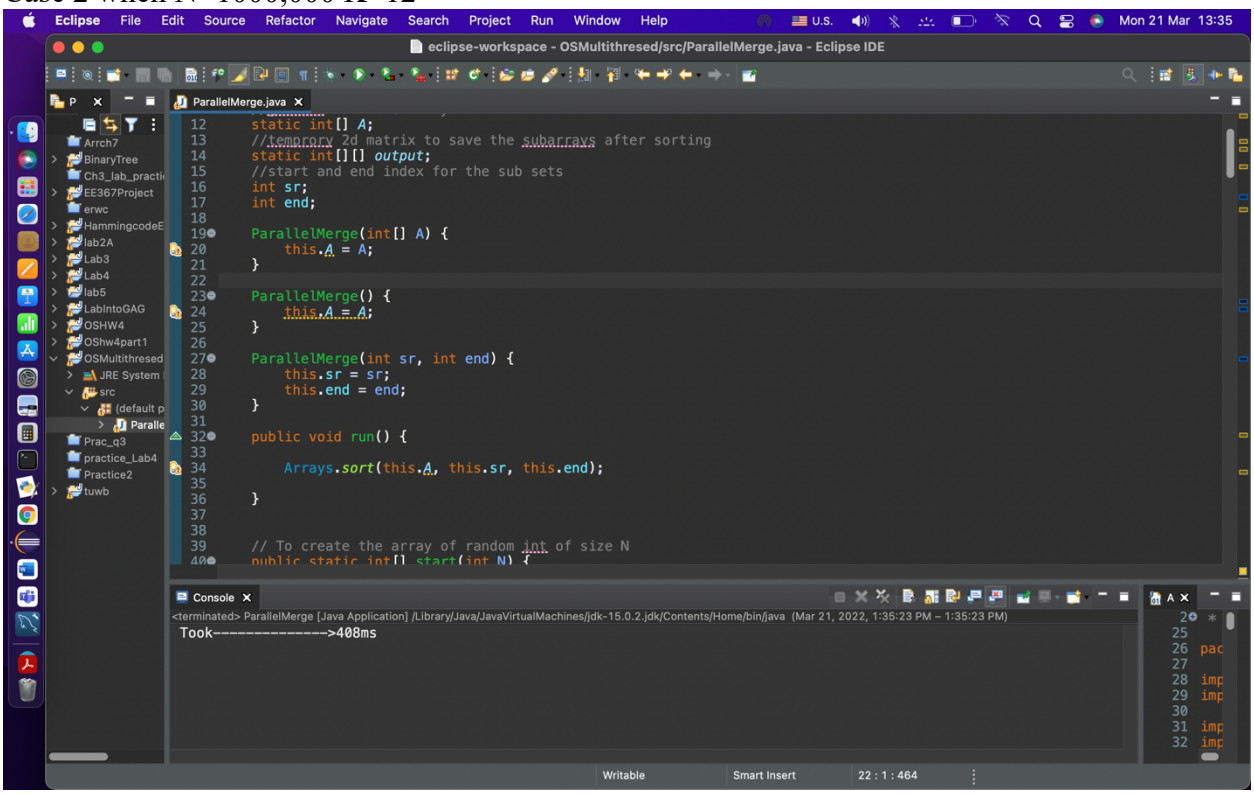
Case 2 when N=1000,000 K=8



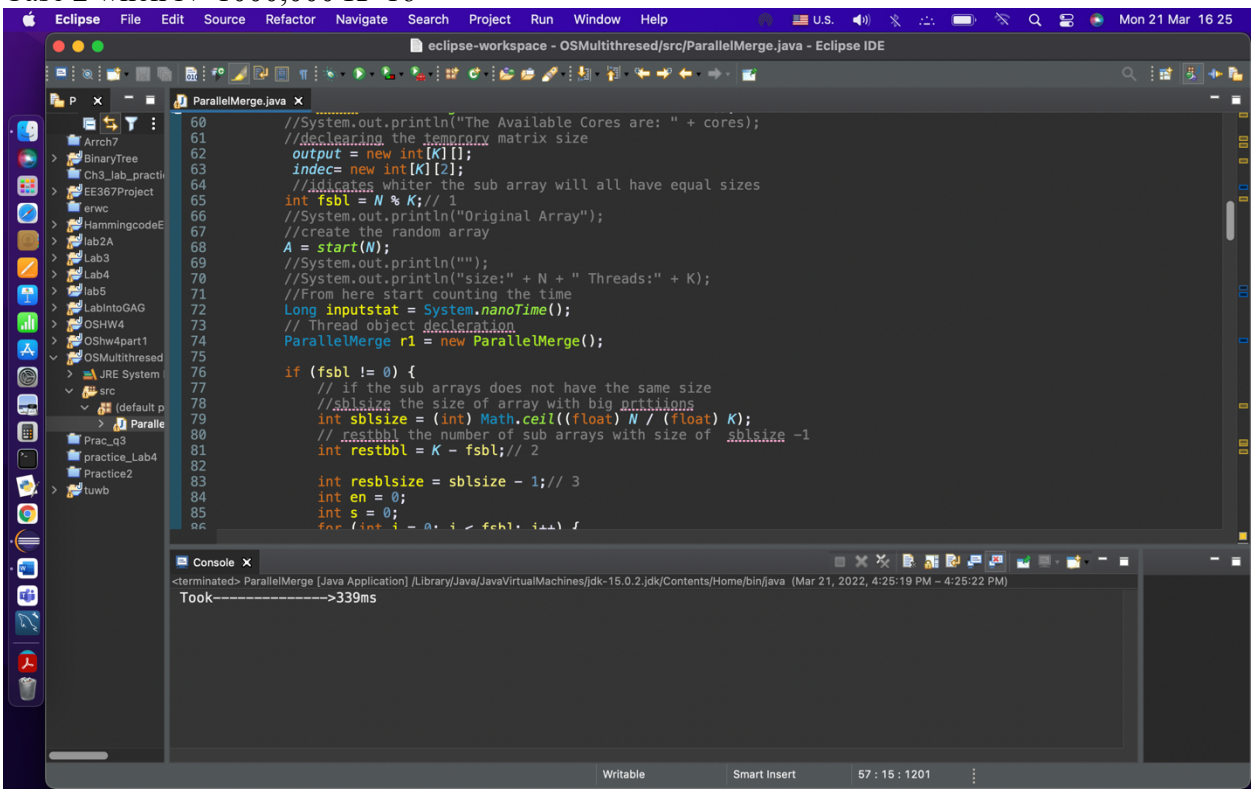
Case 2 when N=1000,000 K=12



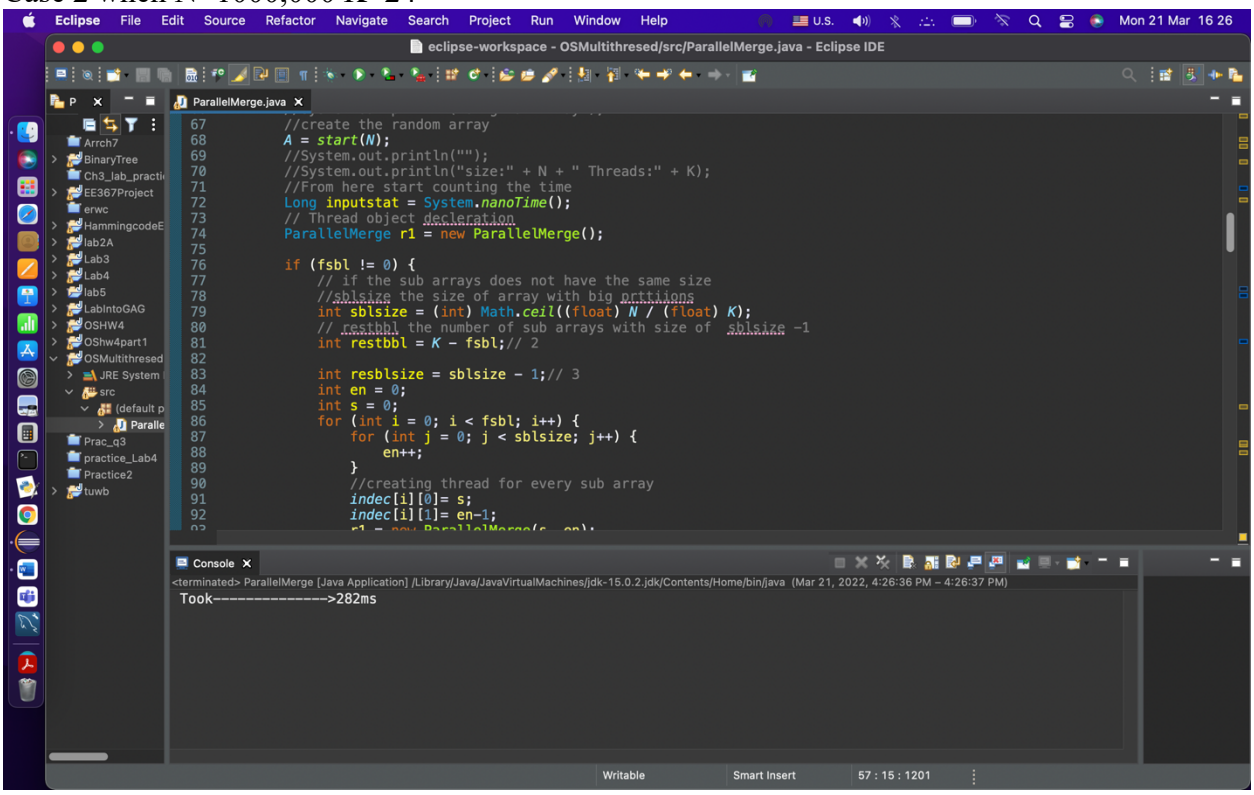
Case 2 when N=1000,000 K=12



Case 2 when N=1000,000 K=16



Case 2 when N=1000,000 K=24



Case 2 when N=1000,000 K=32

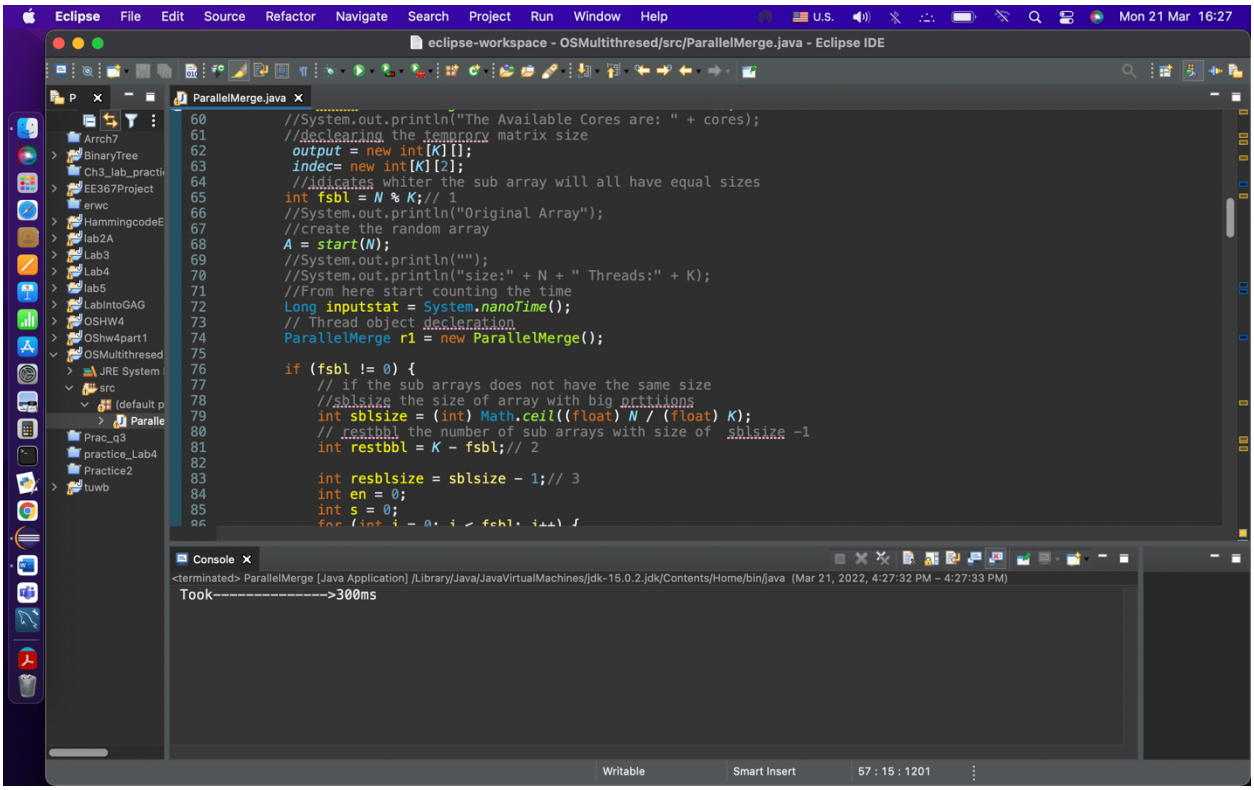
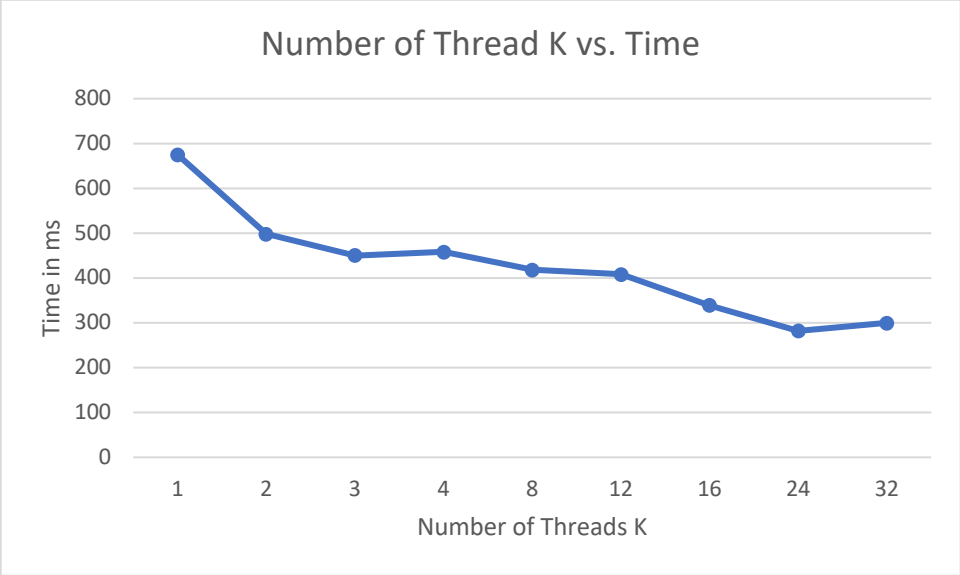
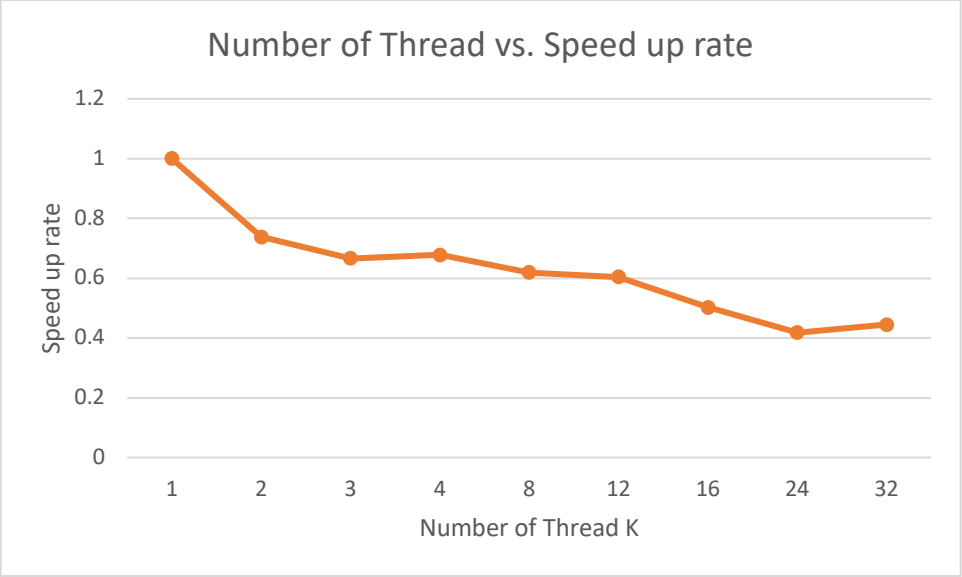


Table 1- Where N = 1,000,000 K Varies

T	675	498	450	458	418	408	339	282	300
N	1,000,000								
K	1	2	3	4	8	12	16	24	32
Speed Up Rate	1	0.737	0.667	0.678	0.6192	0.6044	0.5022	0.4178	0.444



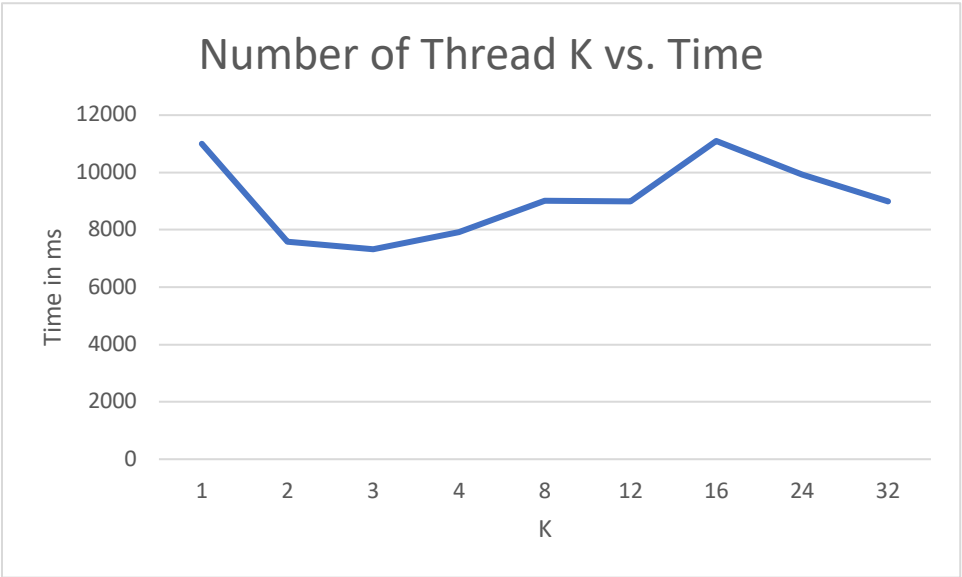
It is presented from the graph of K vs. T that the total time started to decrease with the increasing of the number of threads until K=4, and this highly illustrated the theory of CPU time sharing and parallelism of process and threads. However, when K>4 which represents the available cores for my device (dual processor, 2 cores) the program total time will increase with the increasing of the number of threads as in case K=4 the total number of threads will be 5 along with the main thread that will not be occupy with this device performance but still the results will be better than the serial single process when k=1.

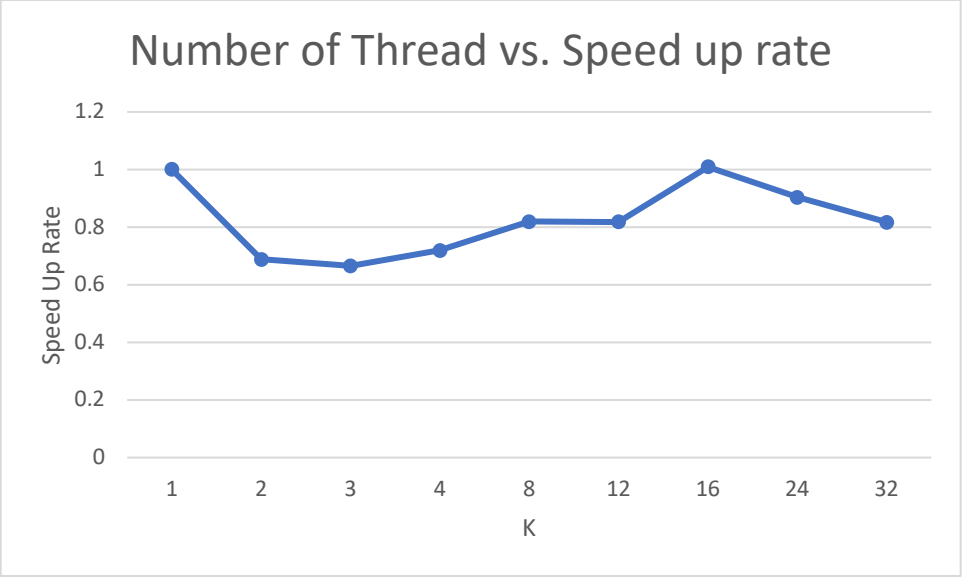


Speed up rate was calculated by dividing the time recoded for each K by the time recorded for K=1 and the result obtained were plotted as well. From the plot its clear that the time taken by the K>1 was less than the serial thread when K=1 and it also indicates the time will decrease with the increment of the thread however the hardware of the device used to run this algorithm needs to be taken into consideration. As in this case the hardware used was had 4 physical cores so from K=4 the response of changing the K and time will be changing.

Table 2- Extra Case Testing When N=100,000,000

T	11003	7578	7323	7911	9011	9000	11100	9939	8993
N	100,000,000								
K	1	2	3	4	8	12	16	24	32
Speed up rate	1	0.737	0.667	0.678	0.6192	0.6044	0.5022	0.4178	0.444





In Table2, this case the effect of hardware cores is highly illustrated as after K=4 the program will take more time executing the algorithm because the number of threads are more the available cores.

Testing the program on macOS cmd using line argument as inputs