Pranoti Dhabu (002124373)

Program Structures & Algorithms

Fall 2021

Assignment No. 5 (Parallel Sort)

⊙ Tasks performed

i. Modified Main.java and ParSort.java to use multithreading (thread pool).

ii. Modified main method to ignore the command line arguments and iterate over a loop for

   a. Varying thread counts (2 to 2048)

   b. Varying cutoff values (5000 to N or N/2)

   c. Varying array size (1M to 32M)

   Doubling method was implemented to benchmark values.

iii. Ran the Java code – Main for various thread count and array sizes.

iv. Plotted a graph to understand and analyze results.

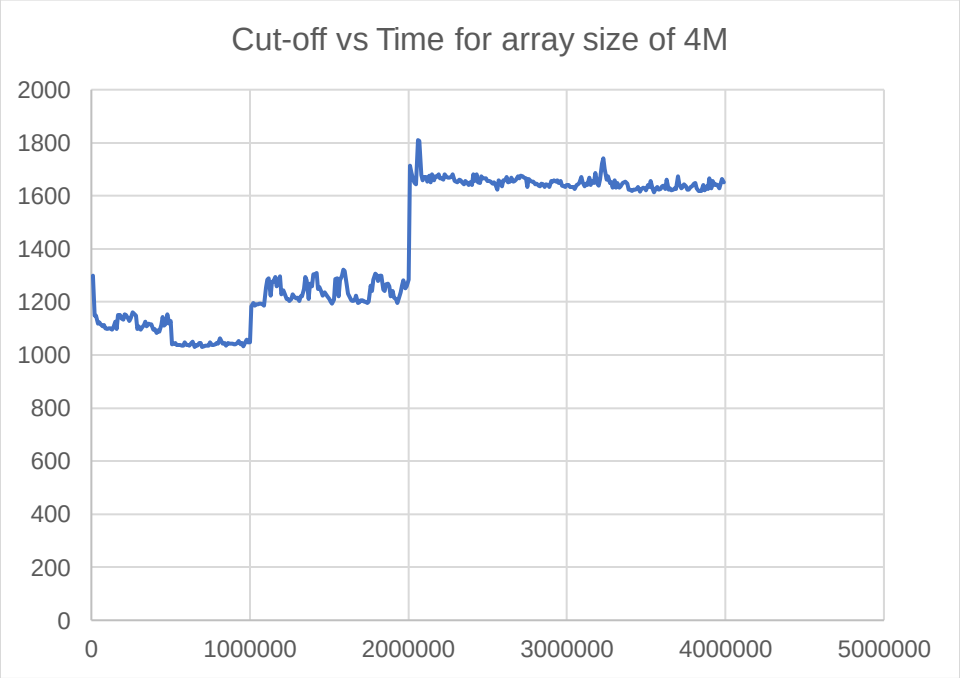
⊙ Relationship Conclusion:

A standard plot of cutoff vs time to sort was plotted for varying number of threads. From the graph (attached below) it is evident that high number of threads is not much of a contributor to improving the efficiency of sorting as much as the cut-off value.

1. **Increasing the number of threads beyond 8 had no significant improvement in efficiency** of sorting. The graph is a linear plot of increasing cut-off irrespective of number of threads and took same time for sorting an array of size 2M.

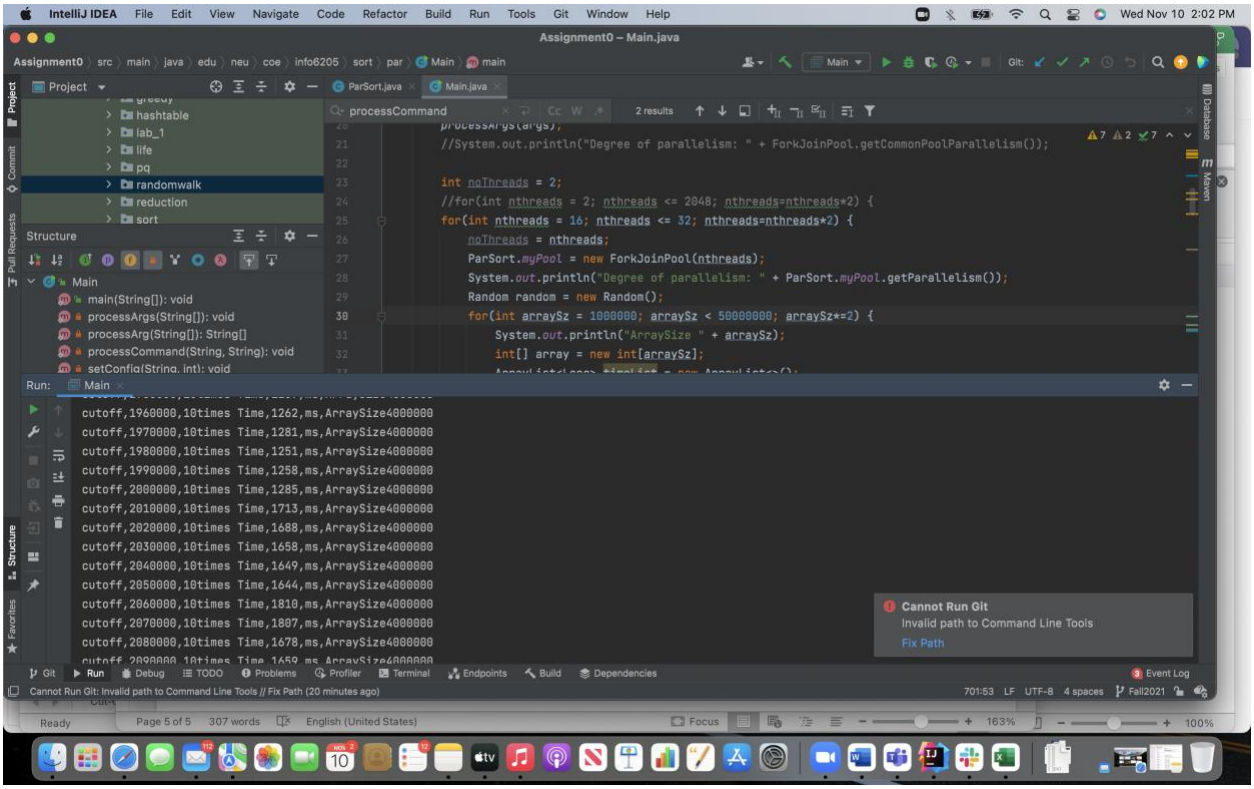
2. **Cut-off should be less than or equal to N/4 and greater than N/8.** If cutoff is between N/4 and N/2 or between N/8 and N/4, then performance degrades slightly but it is still acceptable. However performance degrades significantly when cutoff is greater than N/2.

With this experiment I can conclude that **cut-off value should be between N/8 and N/4** and number of threads approx around 8 or 16 to improve the efficiency of sorting.

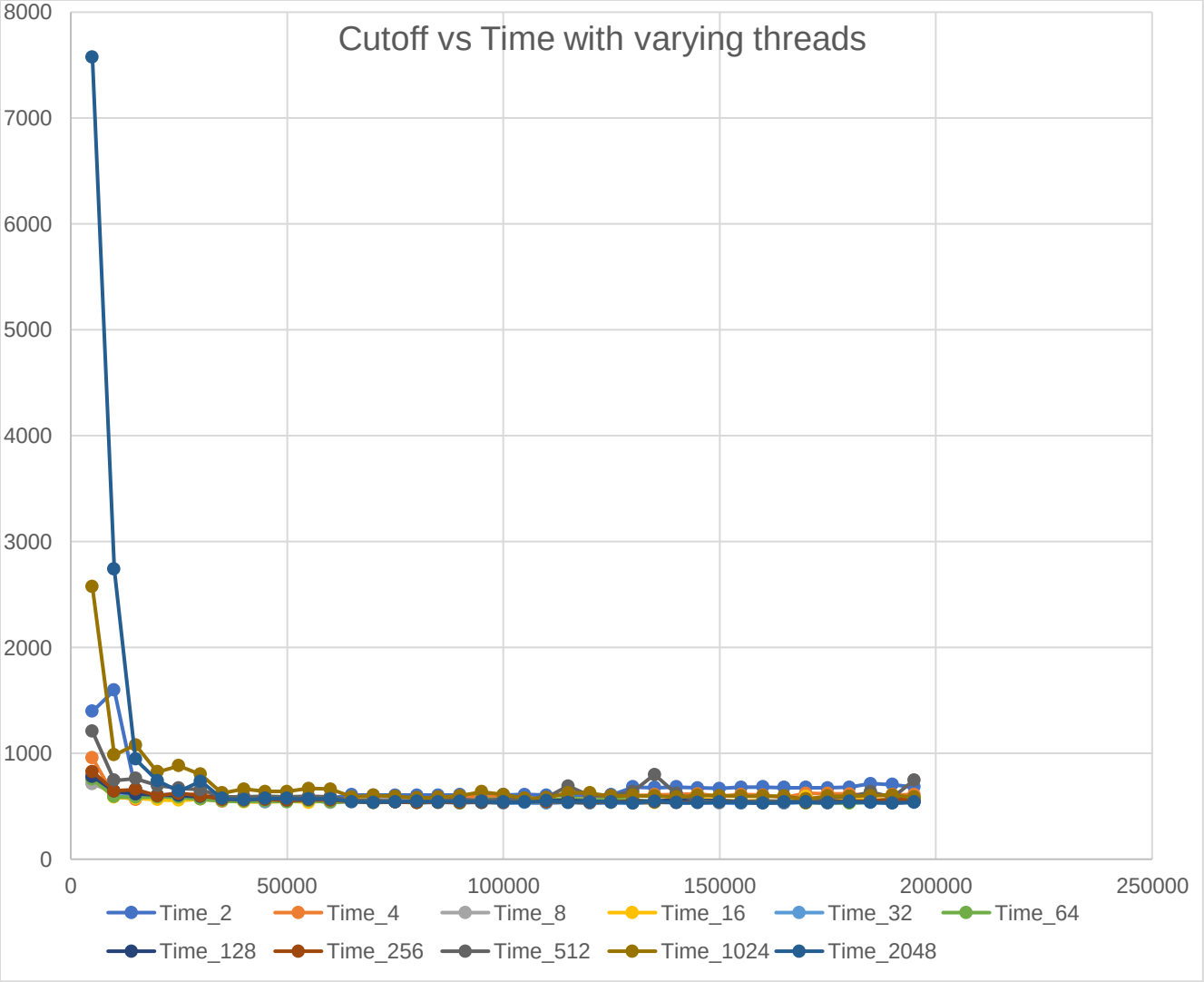


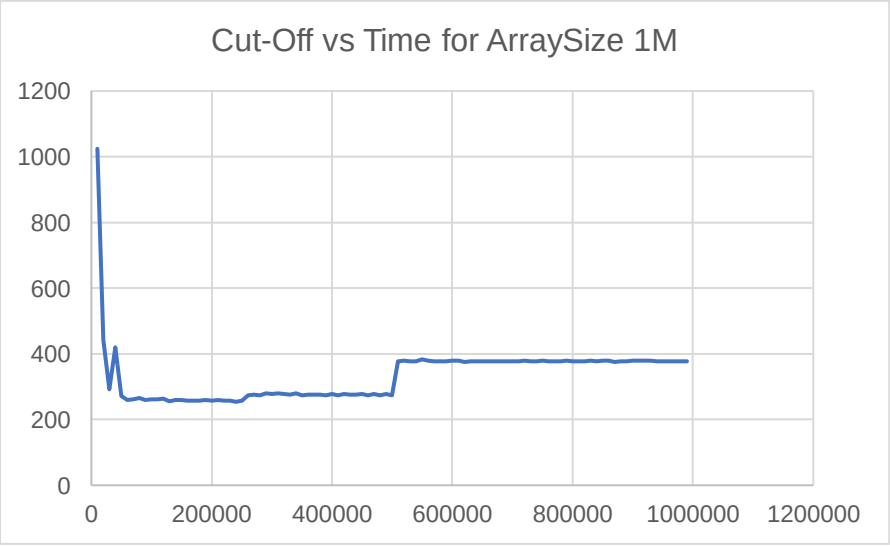
- **Evidence to support the conclusion:**

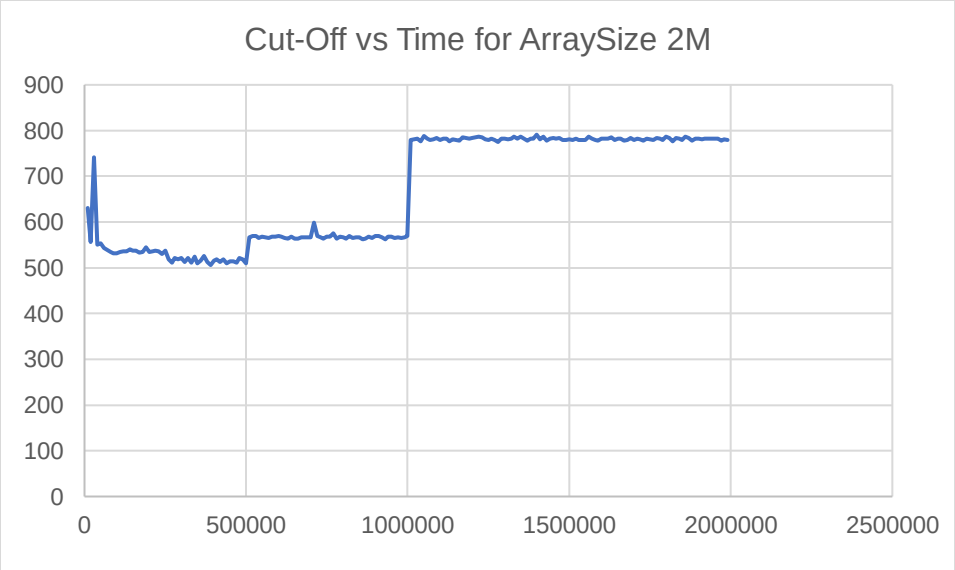
1. **Output (Snapshot of Code output in the terminal)**

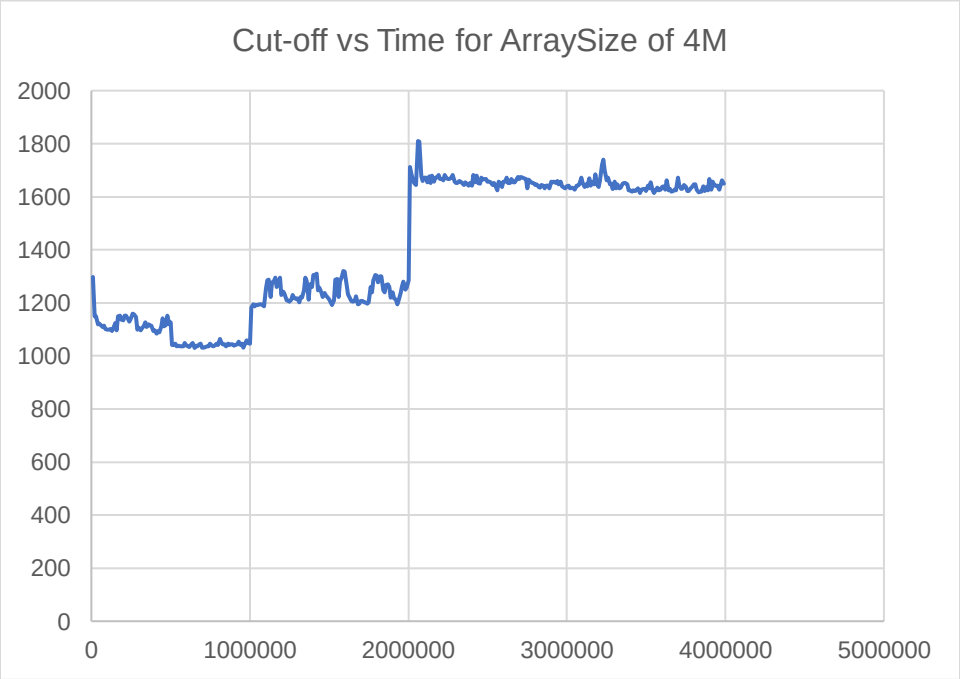


2. **Graphical Representation(Observations from experiments should be tabulated and analyzed by plotting graphs(usually in excel) to arrive on the relationship conclusion)**


Cutoff vs Time with varying threads
8000
7000
6000
5000
4000
3000
2000
1000
0
0
50000
100000
150000
200000
250000
Time_2
Time_4
Time_8
Time_16
Time_32
Time_64
Time_128
Time_256
Time_512
Time_1024
Time_2048


Cut-Off vs Time for ArraySize 1M
1200
1000
800
600
400
200
0
0
200000
400000
600000
800000
1000000
1200000





- **Unit tests result:(Snapshot of successful unit test run)**

N/A