Cheng Yan (001090569)

Program Structures & Algorithms

Fall 2021

Assignment No. 5

⊙ Task (List down the tasks performed in the Assignment)

1. A cutoff (defaults to, say, 1000) which you will update according to the first argument in the command line when running. It's your job to experiment and come up with a good value for this cutoff. If there are fewer elements to sort than the cutoff, then you should use the system sort instead.
2. Recursion depth or the number of available threads. Using this determination, you might decide on an ideal number (t) of separate threads (stick to powers of 2) and arrange for that number of partitions to be parallelized (by preventing recursion after the depth of lg t is reached).
3. An appropriate combination of these.

⊙ Relationship Conclusion:

*All conclusions are concluded based on my personal 2017 MacBook Pro, the results may vary by the performance of core and number of core.

1. **It's most effective when cutoff is Set to around 25% (1/4) of the array size.**

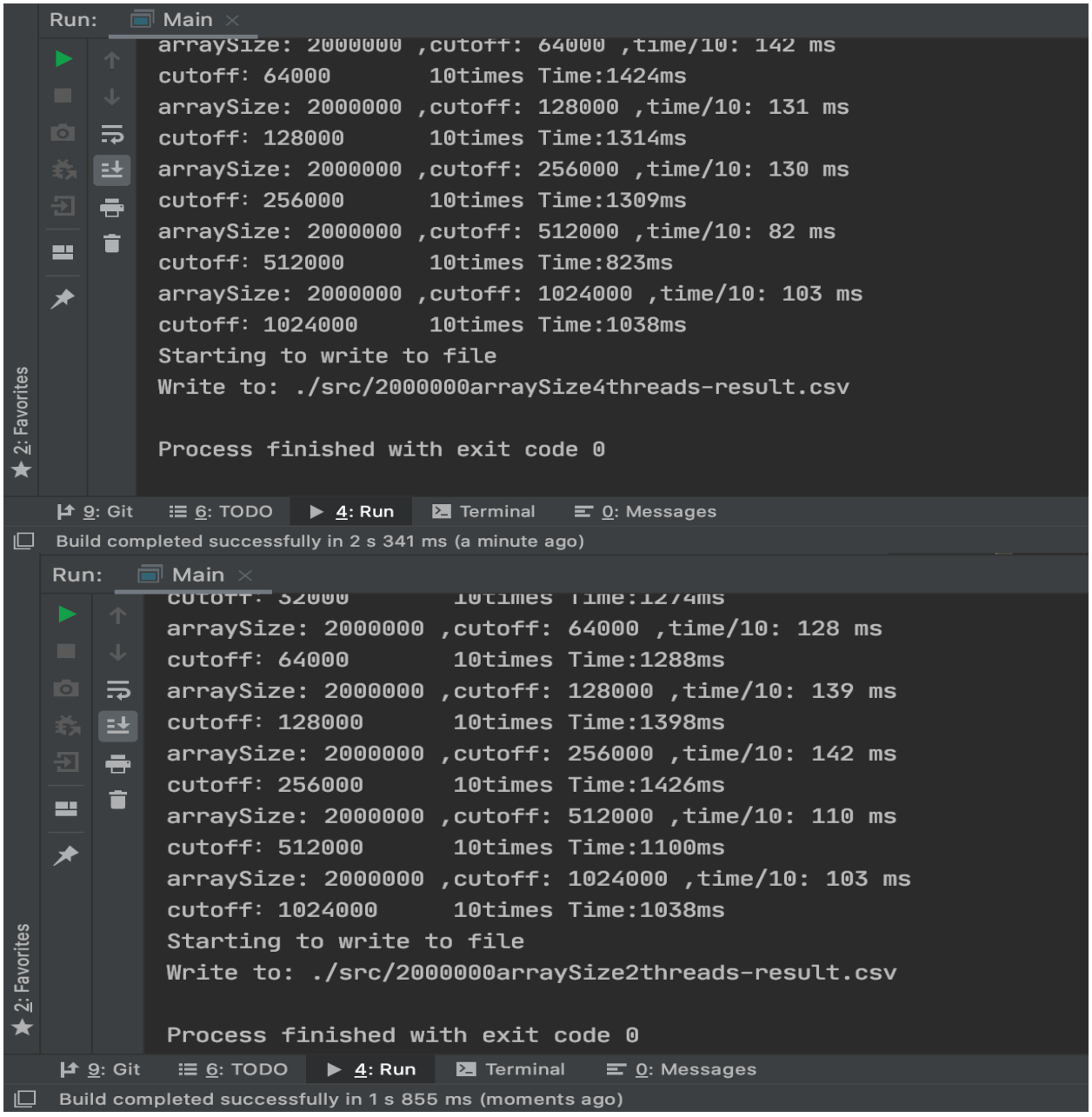
$$\boldsymbol{Cutoff = \frac{1}{4}\, Array\ size}$$

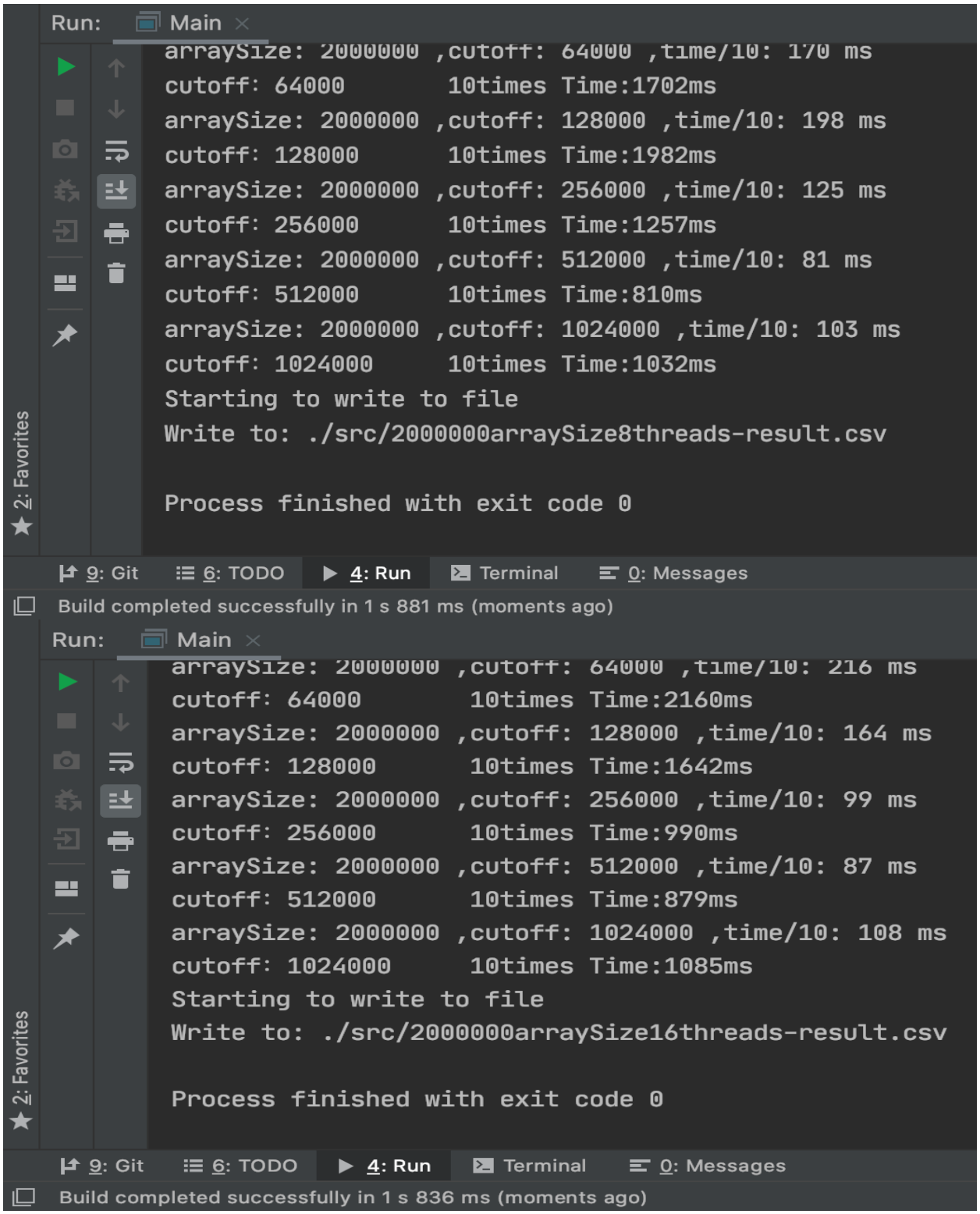
2. **After 8 threads the time it cost is almost same.**

   So it's best to set it to 8 threads

- **Evidence to support the conclusion:**

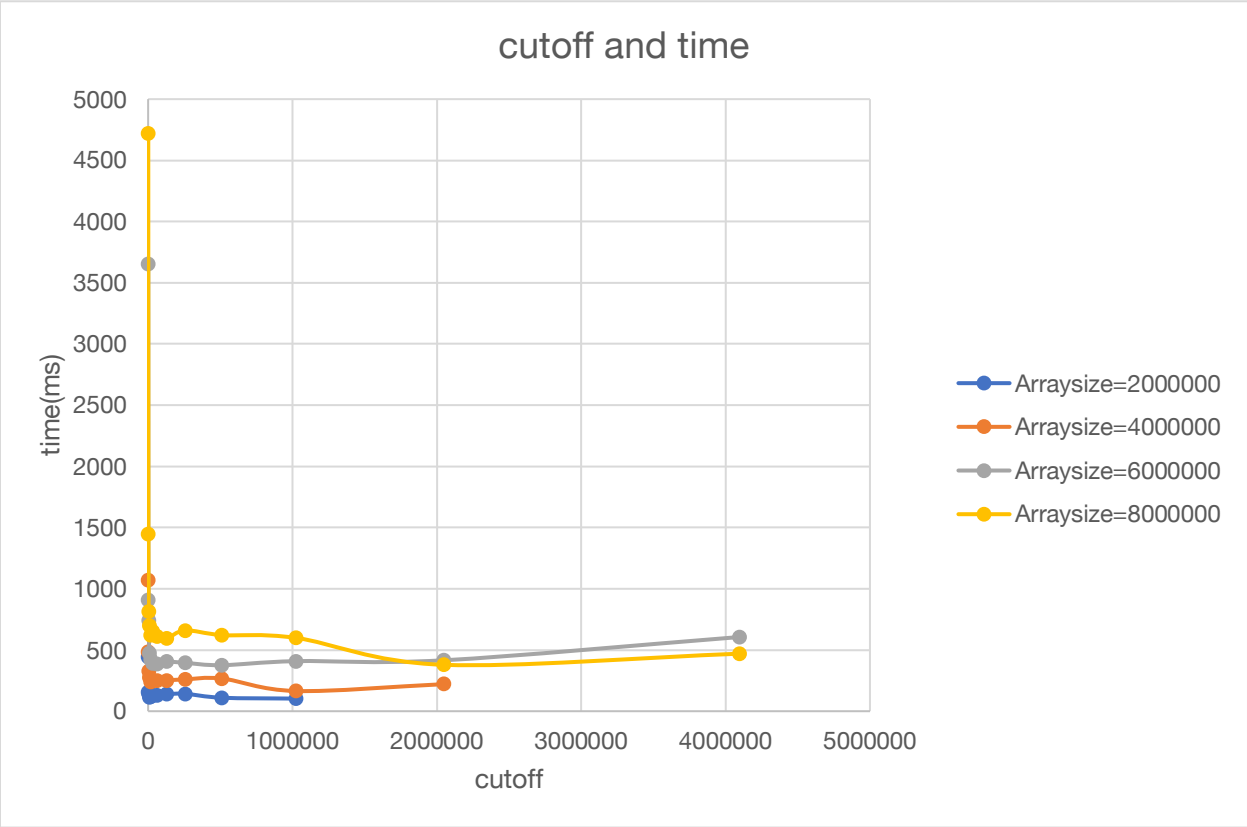
1. **Output (Snapshot of Code output in the terminal)**



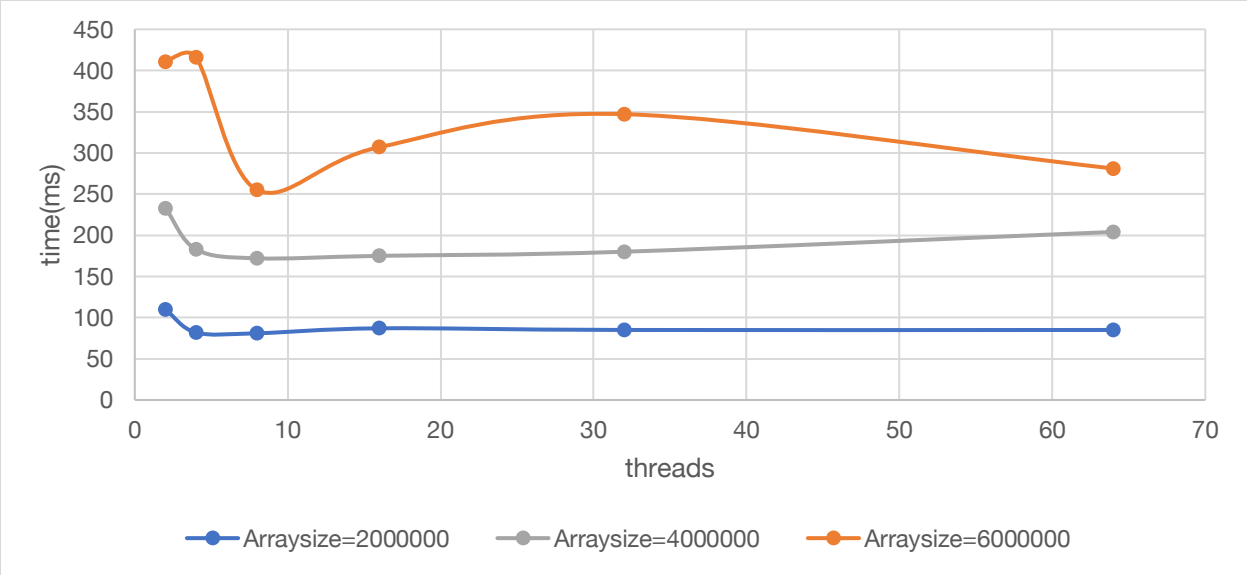


Here is part output in the console, the entire output in the under this directory (*.csv). And the filename is based on the array size and thread number.

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



Graph 1 cutoff and time relation and different array size



Graph 2 threads and time relation and different array size