

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.

$$Cutoff = \frac{1}{4} \text{ Array size}$$

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)

The image displays two screenshots of a terminal window, likely from an IDE, showing the output of a program. The terminal is titled 'Run: Main'.

**Top Screenshot:**

```
arraySize: 20000000 ,cutoff: 64000 ,time/10: 142 ms
cutoff: 64000      10times Time:1424ms
arraySize: 20000000 ,cutoff: 128000 ,time/10: 131 ms
cutoff: 128000     10times Time:1314ms
arraySize: 20000000 ,cutoff: 256000 ,time/10: 130 ms
cutoff: 256000     10times Time:1309ms
arraySize: 20000000 ,cutoff: 512000 ,time/10: 82 ms
cutoff: 512000     10times Time:823ms
arraySize: 20000000 ,cutoff: 1024000 ,time/10: 103 ms
cutoff: 1024000    10times Time:1038ms
Starting to write to file
Write to: ./src/20000000arraySize4threads-result.csv

Process finished with exit code 0
```

**Bottom Screenshot:**

```
CUTOFF: 32000      10times Time:1274ms
arraySize: 20000000 ,cutoff: 64000 ,time/10: 128 ms
cutoff: 64000      10times Time:1288ms
arraySize: 20000000 ,cutoff: 128000 ,time/10: 139 ms
cutoff: 128000     10times Time:1398ms
arraySize: 20000000 ,cutoff: 256000 ,time/10: 142 ms
cutoff: 256000     10times Time:1426ms
arraySize: 20000000 ,cutoff: 512000 ,time/10: 110 ms
cutoff: 512000     10times Time:1100ms
arraySize: 20000000 ,cutoff: 1024000 ,time/10: 103 ms
cutoff: 1024000    10times Time:1038ms
Starting to write to file
Write to: ./src/20000000arraySize2threads-result.csv

Process finished with exit code 0
```

Both screenshots show a series of array sizes (20,000,000) and cutoff values (64,000, 128,000, 256,000, 512,000, 1,024,000) being tested. The output includes the time taken for each test (e.g., 142 ms, 131 ms, etc.) and the total time for 10 iterations (e.g., 1424ms, 1314ms, etc.). The program then writes the results to a CSV file and finishes with exit code 0.

```
Run: Main x
arraySize: 20000000 ,cutoff: 64000 ,time/10: 170 ms
cutoff: 64000      10times Time:1702ms
arraySize: 20000000 ,cutoff: 128000 ,time/10: 198 ms
cutoff: 128000     10times Time:1982ms
arraySize: 20000000 ,cutoff: 256000 ,time/10: 125 ms
cutoff: 256000     10times Time:1257ms
arraySize: 20000000 ,cutoff: 512000 ,time/10: 81 ms
cutoff: 512000     10times Time:810ms
arraySize: 20000000 ,cutoff: 1024000 ,time/10: 103 ms
cutoff: 1024000    10times Time:1032ms
Starting to write to file
Write to: ./src/20000000arraySize8threads-result.csv

Process finished with exit code 0

9: Git 6: TODO 4: Run Terminal 0: Messages
Build completed successfully in 1 s 881 ms (moments ago)

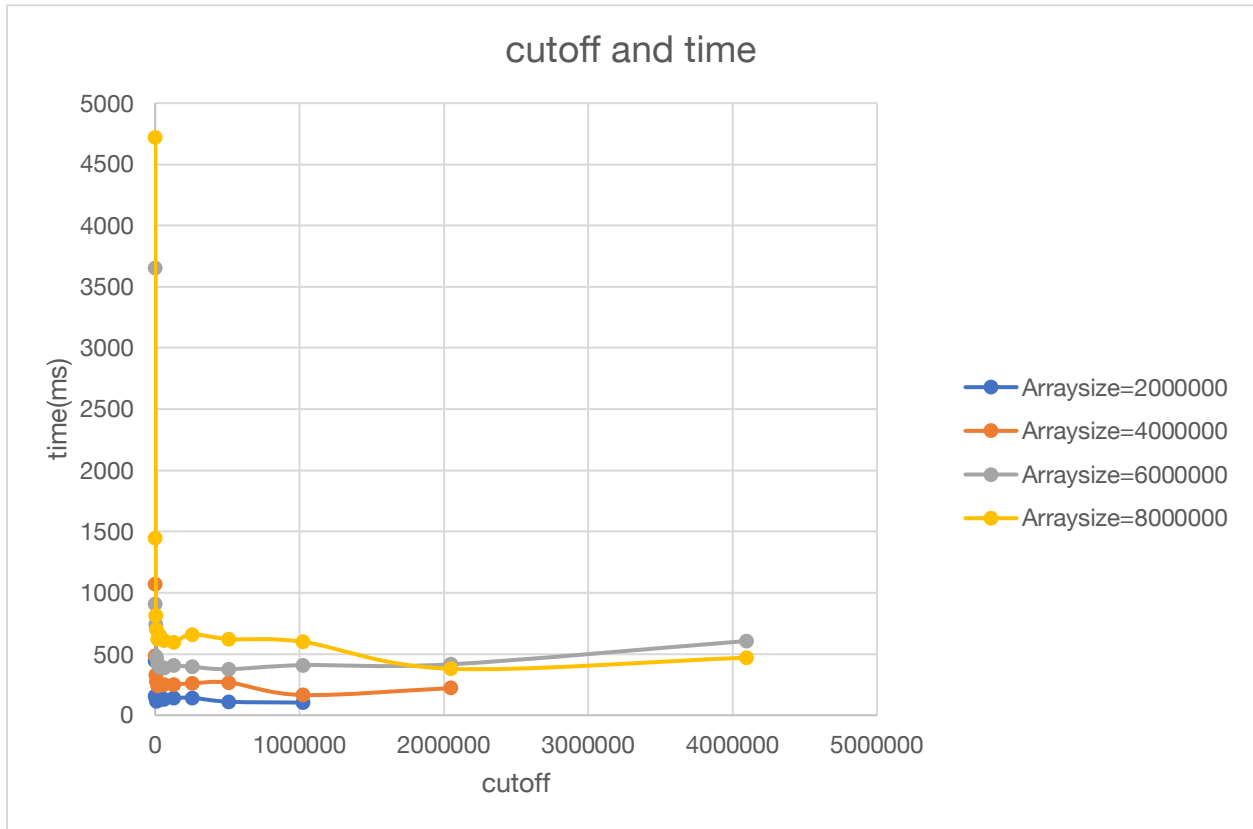
Run: Main x
arraySize: 20000000 ,cutoff: 64000 ,time/10: 216 ms
cutoff: 64000      10times Time:2160ms
arraySize: 20000000 ,cutoff: 128000 ,time/10: 164 ms
cutoff: 128000     10times Time:1642ms
arraySize: 20000000 ,cutoff: 256000 ,time/10: 99 ms
cutoff: 256000     10times Time:990ms
arraySize: 20000000 ,cutoff: 512000 ,time/10: 87 ms
cutoff: 512000     10times Time:879ms
arraySize: 20000000 ,cutoff: 1024000 ,time/10: 108 ms
cutoff: 1024000    10times Time:1085ms
Starting to write to file
Write to: ./src/20000000arraySize16threads-result.csv

Process finished with exit code 0

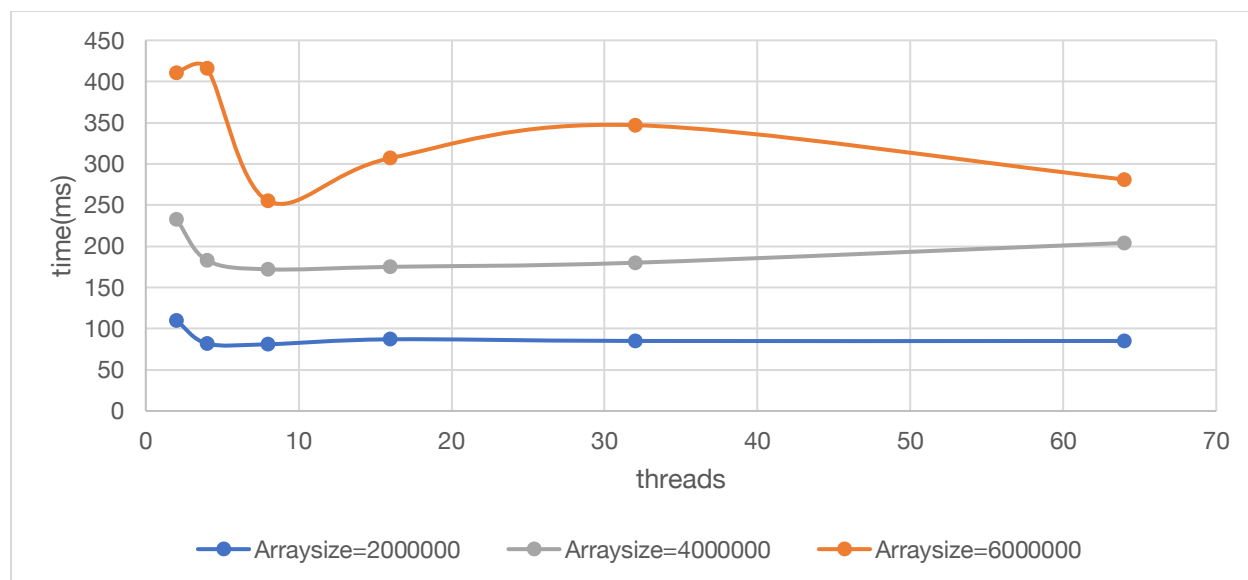
9: Git 6: TODO 4: Run Terminal 0: Messages
Build completed successfully in 1 s 836 ms (moments ago)
```

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