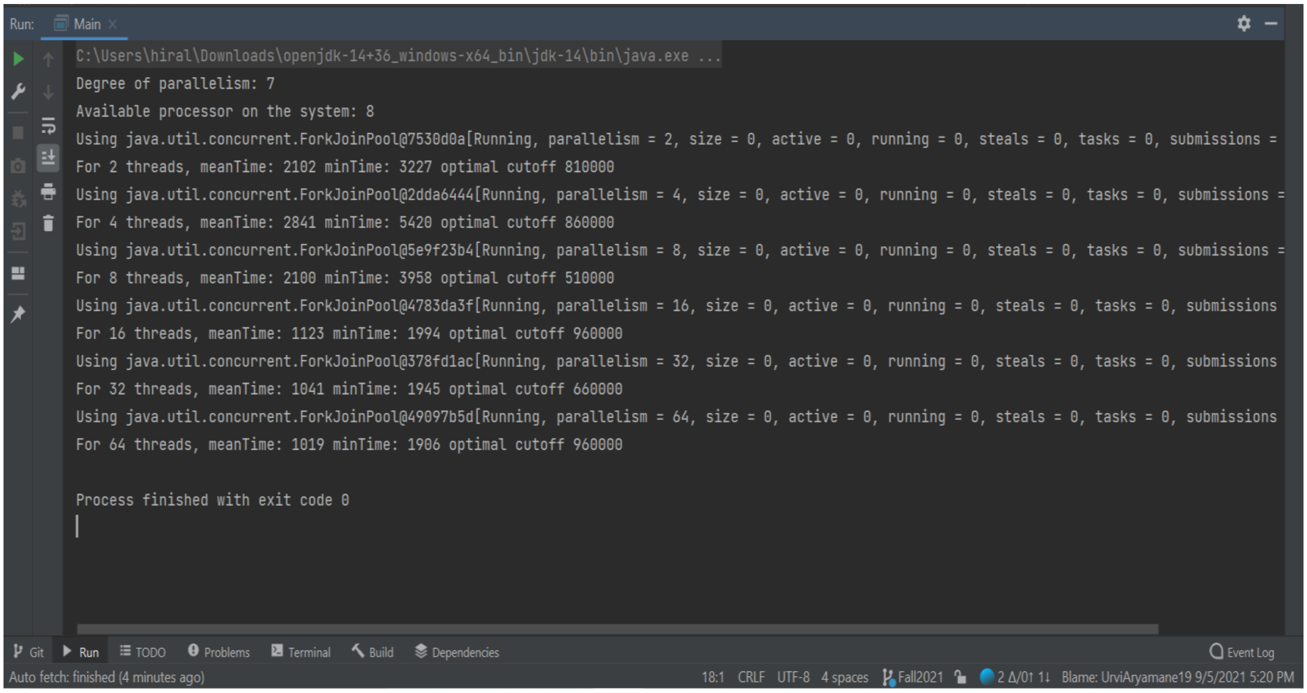
MSSES INFO6205 13718 PROGRAM STRUCTURES & ALGORITHMS SEC 01- FALL2021
ASSIGNMENT 5: HIRAL RAJESH NAGDA (001560027)

The snapshot below is for different threads vs cutoff values for a constant array size of 2000000


Run:
Main
Git
Run
TODO
Problems
Terminal
Build
Dependencies
Event Log
Auto fetch: finished (4 minutes ago)
18:1 CRLF UTF-8 4 spaces Fall2021 2 Δ/01 1↓ Blame: UrviAryamane19 9/5/2021 5:20 PM

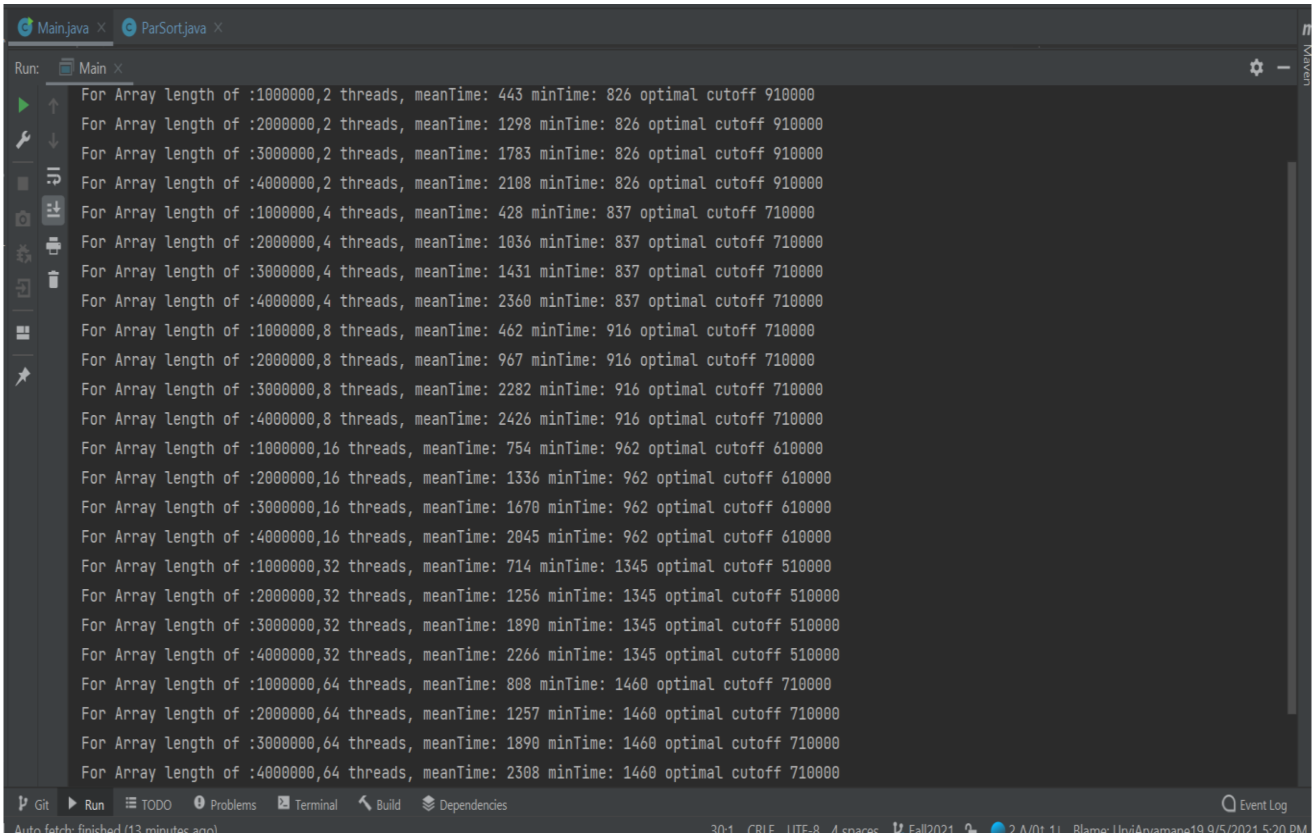
```
C:\Users\hiral\Downloads\openjdk-14+36_windows-x64_bin\jdk-14\bin\java.exe ...
Degree of parallelism: 7
Available processor on the system: 8
Using java.util.concurrent.ForkJoinPool@7530d0a[Running, parallelism = 2, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions =
For 2 threads, meanTime: 2102 minTime: 3227 optimal cutoff 810000
Using java.util.concurrent.ForkJoinPool@2dda6444[Running, parallelism = 4, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions =
For 4 threads, meanTime: 2841 minTime: 5420 optimal cutoff 860000
Using java.util.concurrent.ForkJoinPool@5e9f23b4[Running, parallelism = 8, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions =
For 8 threads, meanTime: 2100 minTime: 3958 optimal cutoff 510000
Using java.util.concurrent.ForkJoinPool@4783da3f[Running, parallelism = 16, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions
For 16 threads, meanTime: 1123 minTime: 1994 optimal cutoff 960000
Using java.util.concurrent.ForkJoinPool@378fd1ac[Running, parallelism = 32, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions
For 32 threads, meanTime: 1041 minTime: 1945 optimal cutoff 660000
Using java.util.concurrent.ForkJoinPool@49097b5d[Running, parallelism = 64, size = 0, active = 0, running = 0, steals = 0, tasks = 0, submissions
For 64 threads, meanTime: 1019 minTime: 1906 optimal cutoff 960000

Process finished with exit code 0
```

From above snapshot we can conclude that for 8 threads the cut off is most optimal of 510000 for a fixed size array of 2000000

MSSES INFO6205 13718 PROGRAM STRUCTURES & ALGORITHMS SEC 01- FALL2021 ASSIGNMENT 5: HIRAL RAJESH NAGDA (001560027)

The snapshot below is for different threads vs cutoff values for a varying array sizes



```
For Array length of :1000000,2 threads, meanTime: 443 minTime: 826 optimal cutoff 910000
For Array length of :2000000,2 threads, meanTime: 1298 minTime: 826 optimal cutoff 910000
For Array length of :3000000,2 threads, meanTime: 1783 minTime: 826 optimal cutoff 910000
For Array length of :4000000,2 threads, meanTime: 2108 minTime: 826 optimal cutoff 910000
For Array length of :1000000,4 threads, meanTime: 428 minTime: 837 optimal cutoff 710000
For Array length of :2000000,4 threads, meanTime: 1036 minTime: 837 optimal cutoff 710000
For Array length of :3000000,4 threads, meanTime: 1431 minTime: 837 optimal cutoff 710000
For Array length of :4000000,4 threads, meanTime: 2360 minTime: 837 optimal cutoff 710000
For Array length of :1000000,8 threads, meanTime: 462 minTime: 916 optimal cutoff 710000
For Array length of :2000000,8 threads, meanTime: 967 minTime: 916 optimal cutoff 710000
For Array length of :3000000,8 threads, meanTime: 2282 minTime: 916 optimal cutoff 710000
For Array length of :4000000,8 threads, meanTime: 2426 minTime: 916 optimal cutoff 710000
For Array length of :1000000,16 threads, meanTime: 754 minTime: 962 optimal cutoff 610000
For Array length of :2000000,16 threads, meanTime: 1336 minTime: 962 optimal cutoff 610000
For Array length of :3000000,16 threads, meanTime: 1670 minTime: 962 optimal cutoff 610000
For Array length of :4000000,16 threads, meanTime: 2045 minTime: 962 optimal cutoff 610000
For Array length of :1000000,32 threads, meanTime: 714 minTime: 1345 optimal cutoff 510000
For Array length of :2000000,32 threads, meanTime: 1256 minTime: 1345 optimal cutoff 510000
For Array length of :3000000,32 threads, meanTime: 1890 minTime: 1345 optimal cutoff 510000
For Array length of :4000000,32 threads, meanTime: 2266 minTime: 1345 optimal cutoff 510000
For Array length of :1000000,64 threads, meanTime: 808 minTime: 1460 optimal cutoff 710000
For Array length of :2000000,64 threads, meanTime: 1257 minTime: 1460 optimal cutoff 710000
For Array length of :3000000,64 threads, meanTime: 1890 minTime: 1460 optimal cutoff 710000
For Array length of :4000000,64 threads, meanTime: 2308 minTime: 1460 optimal cutoff 710000
```

If thread count matches the number of processors then it takes minimum time as well as gives the optimal value of cutoff. That means there is most processor engagement and least context switching gives the best result of cutoff value at different array sizes as well.

Its not optimal in all cases but most optimal at array size of 1000000 thread count: 8 and mean time is 462ms and cutoff is 710000

Hence performance improvements can be seen when cutoff value results in number of subtasks nearing the processor count. Ratio of cutoff and arraysize 710000/1000000 approx to 8,

- The observations are in the excel file named ‘assignment_console_ouput_1’ in the project directory