Program Structures & Algorithms

Spring 2022

Assignment No. 4 – Parallel Sorting

Name: Tanya Shah
(NUID): 002988713

- **Task**

Implement Parallel Sorting such that each partition is sorted in parallel based on following -

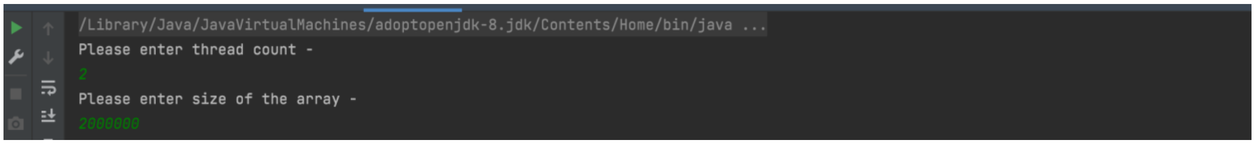
1. Find the optimal cutoff value to identify when to switch to system sort.
2. Update the cutoff values while sorting array partitions.
3. Determine recursion depth or the number of available threads to identify number of partitions to be parallelized.
4. Finding the optimal value of threads.

Default Values for the experiment performed -

Number of threads - 2

Size of the array - 2000000

These values double in each iteration while examining the cutoff values and parallel sorting performance.



```
/Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/Contents/Home/bin/java ...
Please enter thread count -
2
Please enter size of the array -
2000000
```

Part 1 Modify *Main.java*

1. Updated main() method to get initial thread count size and initial array size.
2. Doubling the thread count and array size in each iteration to get results.
3. Used ParSort to get degree of parallelism.
4. Generated csv file for each degree of parallelism.

Part 2 Modify *ParSort.java*

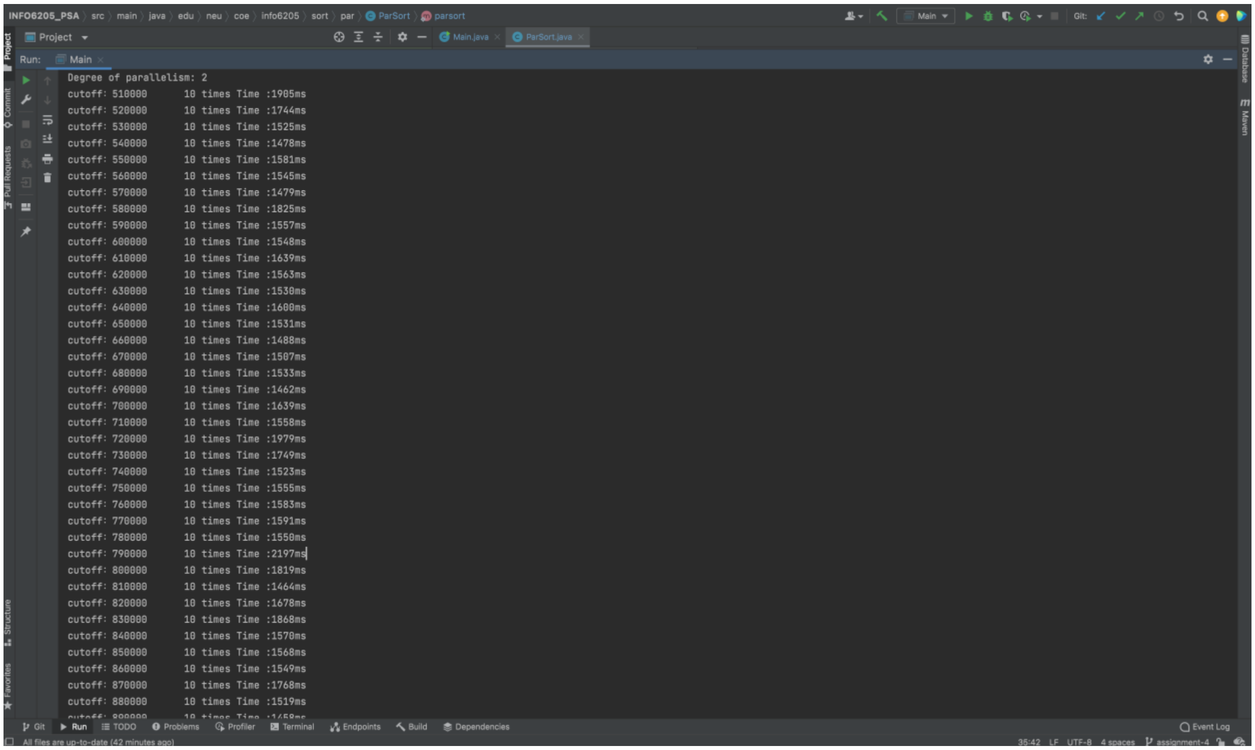
Defined a ForkJoinPool and asynchronously used it in CompletableFuture.

- **Relationship Conclusion**

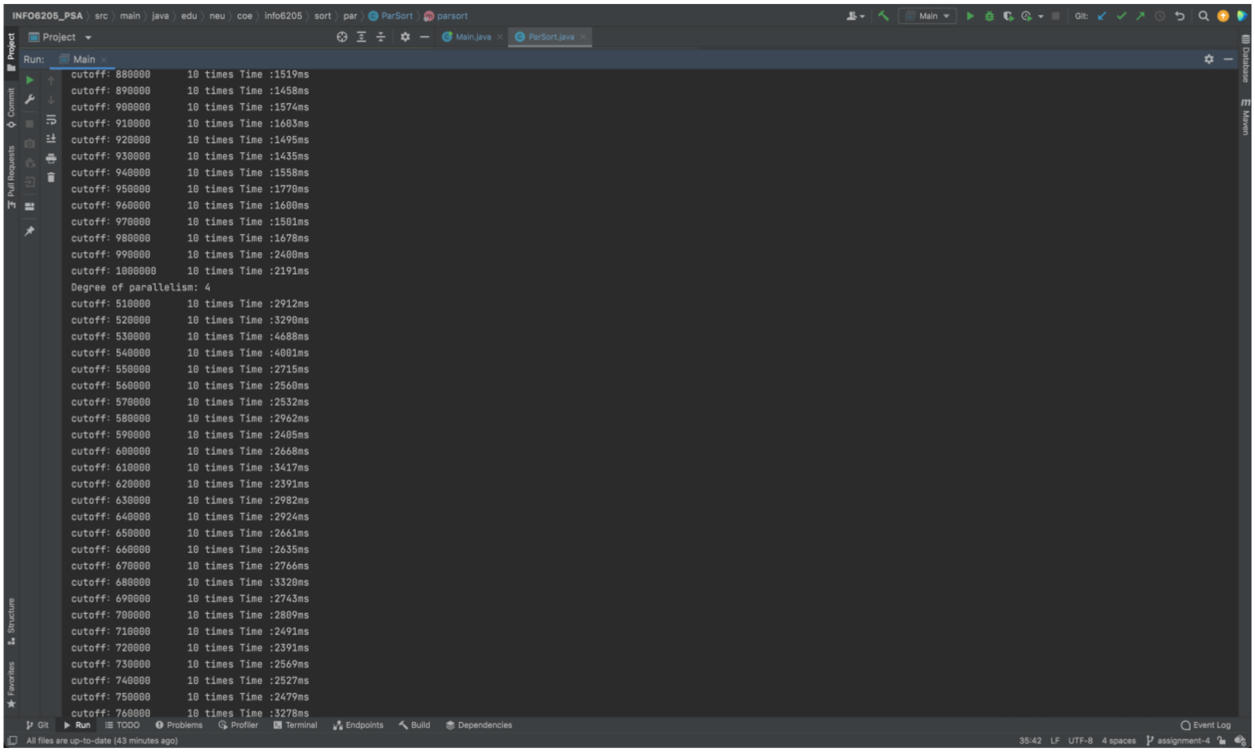
Parallel Sorting uses a balancing mechanism to assign sorting partitions to various threads. It uses cutoff values to identify when to switch to system sort based on the size of the array, as we move towards higher thread count with higher cutoff values, we switch to system sort. Hence, we can say that when the cutoff values are lower and thread count is higher the sorting time increases, lower performance and when cutoff values are lower and thread count is lower the sorting time decreases, higher performance.

- **Output screenshot**

Main.java Output

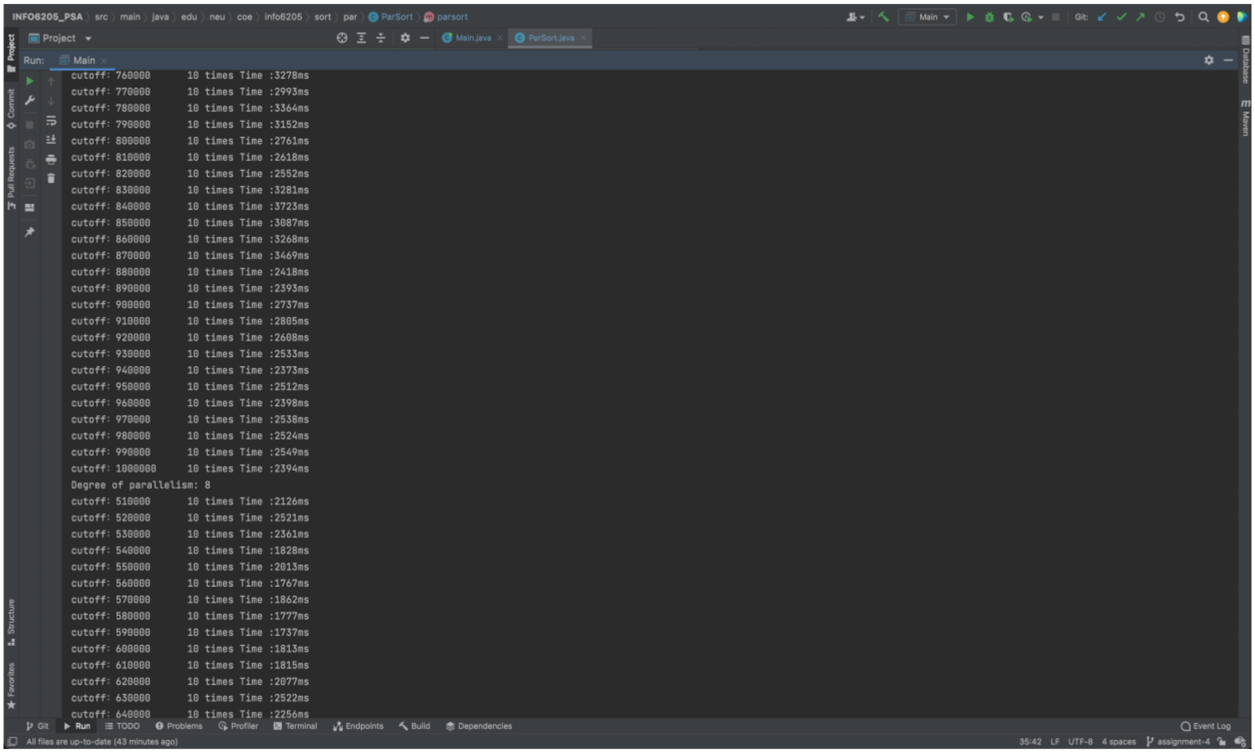

INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run:
Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (42 minutes ago)

```
Degree of parallelism: 2
cutoff: 510000      10 times Time :1905ms
cutoff: 520000      10 times Time :1744ms
cutoff: 530000      10 times Time :1525ms
cutoff: 540000      10 times Time :1478ms
cutoff: 550000      10 times Time :1581ms
cutoff: 560000      10 times Time :1545ms
cutoff: 570000      10 times Time :1479ms
cutoff: 580000      10 times Time :1825ms
cutoff: 590000      10 times Time :1557ms
cutoff: 600000      10 times Time :1548ms
cutoff: 610000      10 times Time :1639ms
cutoff: 620000      10 times Time :1563ms
cutoff: 630000      10 times Time :1530ms
cutoff: 640000      10 times Time :1600ms
cutoff: 650000      10 times Time :1531ms
cutoff: 660000      10 times Time :1488ms
cutoff: 670000      10 times Time :1507ms
cutoff: 680000      10 times Time :1533ms
cutoff: 690000      10 times Time :1462ms
cutoff: 700000      10 times Time :1639ms
cutoff: 710000      10 times Time :1558ms
cutoff: 720000      10 times Time :1979ms
cutoff: 730000      10 times Time :1749ms
cutoff: 740000      10 times Time :1523ms
cutoff: 750000      10 times Time :1555ms
cutoff: 760000      10 times Time :1583ms
cutoff: 770000      10 times Time :1591ms
cutoff: 780000      10 times Time :1550ms
cutoff: 790000      10 times Time :2197ms
cutoff: 800000      10 times Time :1819ms
cutoff: 810000      10 times Time :1464ms
cutoff: 820000      10 times Time :1678ms
cutoff: 830000      10 times Time :1868ms
cutoff: 840000      10 times Time :1570ms
cutoff: 850000      10 times Time :1568ms
cutoff: 860000      10 times Time :1549ms
cutoff: 870000      10 times Time :1768ms
cutoff: 880000      10 times Time :1519ms
cutoff: 890000      10 times Time :1458ms
```



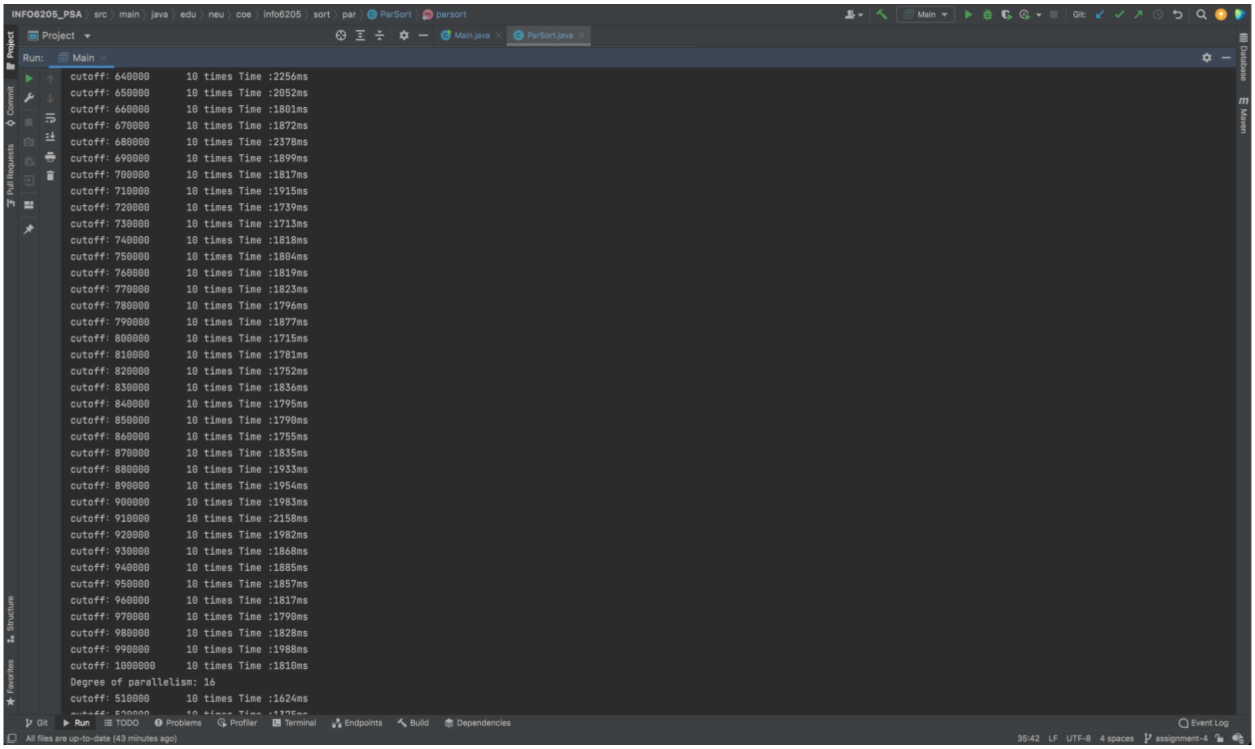
INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (43 minutes ago)
35:42 LF UTF-8 4 spaces assignment-4

```
cutoff: 880000      10 times Time :1519ms
cutoff: 890000      10 times Time :1458ms
cutoff: 900000      10 times Time :1574ms
cutoff: 910000      10 times Time :1603ms
cutoff: 920000      10 times Time :1495ms
cutoff: 930000      10 times Time :1435ms
cutoff: 940000      10 times Time :1558ms
cutoff: 950000      10 times Time :1770ms
cutoff: 960000      10 times Time :1600ms
cutoff: 970000      10 times Time :1501ms
cutoff: 980000      10 times Time :1678ms
cutoff: 990000      10 times Time :2400ms
cutoff: 1000000     10 times Time :2191ms
Degree of parallelism: 4
cutoff: 510000      10 times Time :2912ms
cutoff: 520000      10 times Time :3290ms
cutoff: 530000      10 times Time :4688ms
cutoff: 540000      10 times Time :4001ms
cutoff: 550000      10 times Time :2715ms
cutoff: 560000      10 times Time :2560ms
cutoff: 570000      10 times Time :2532ms
cutoff: 580000      10 times Time :2962ms
cutoff: 590000      10 times Time :2405ms
cutoff: 600000      10 times Time :2668ms
cutoff: 610000      10 times Time :3417ms
cutoff: 620000      10 times Time :2391ms
cutoff: 630000      10 times Time :2982ms
cutoff: 640000      10 times Time :2924ms
cutoff: 650000      10 times Time :2661ms
cutoff: 660000      10 times Time :2635ms
cutoff: 670000      10 times Time :2766ms
cutoff: 680000      10 times Time :3320ms
cutoff: 690000      10 times Time :2743ms
cutoff: 700000      10 times Time :2809ms
cutoff: 710000      10 times Time :2491ms
cutoff: 720000      10 times Time :2391ms
cutoff: 730000      10 times Time :2569ms
cutoff: 740000      10 times Time :2527ms
cutoff: 750000      10 times Time :2479ms
cutoff: 760000      10 times Time :3278ms
```

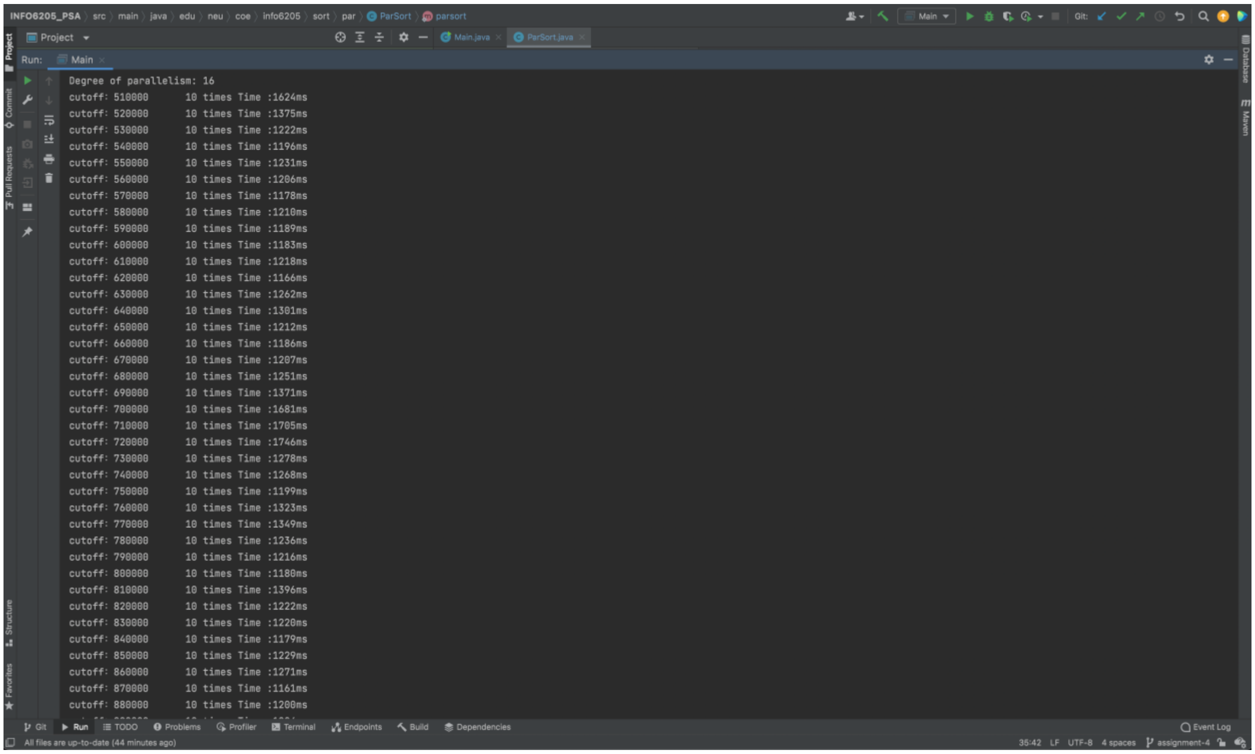


INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (43 minutes ago)
35:42 LF UTF-8 4 spaces assignment-4

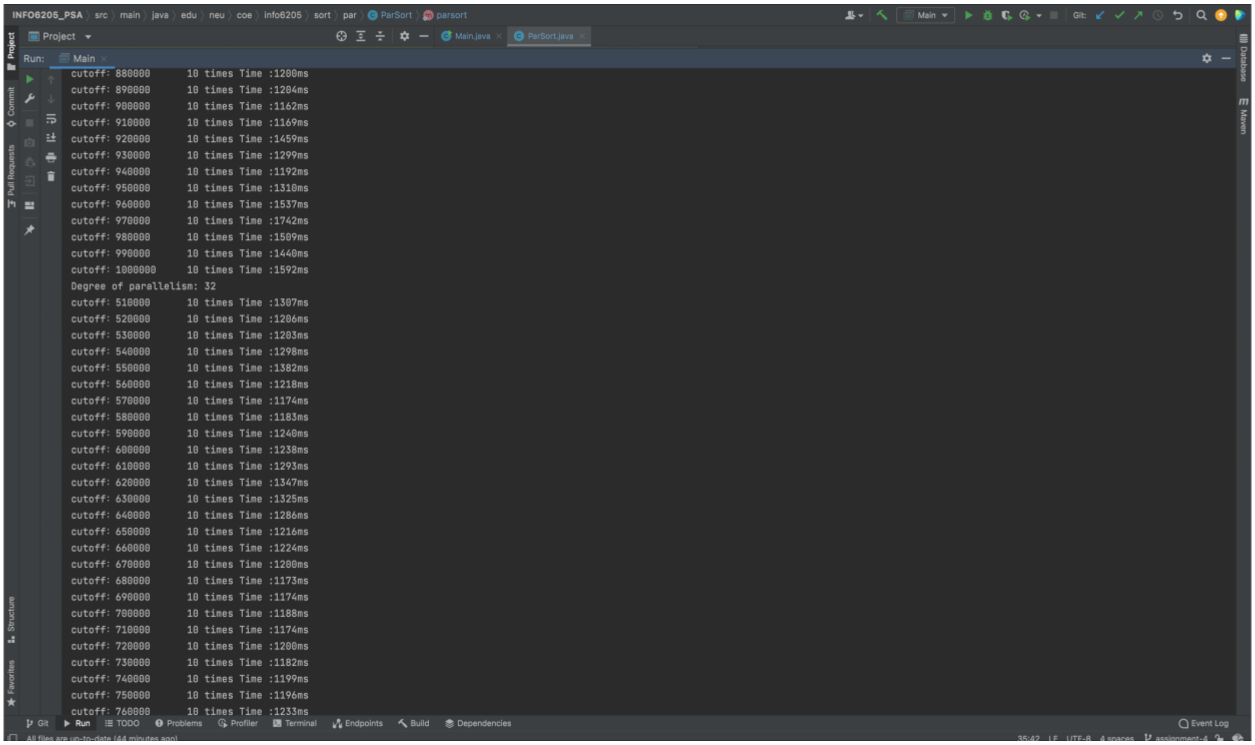
```
cutoff: 760000      10 times Time :3278ms
cutoff: 770000      10 times Time :2993ms
cutoff: 780000      10 times Time :3364ms
cutoff: 790000      10 times Time :3152ms
cutoff: 800000      10 times Time :2761ms
cutoff: 810000      10 times Time :2618ms
cutoff: 820000      10 times Time :2552ms
cutoff: 830000      10 times Time :3281ms
cutoff: 840000      10 times Time :3723ms
cutoff: 850000      10 times Time :3087ms
cutoff: 860000      10 times Time :3268ms
cutoff: 870000      10 times Time :3469ms
cutoff: 880000      10 times Time :2418ms
cutoff: 890000      10 times Time :2393ms
cutoff: 900000      10 times Time :2737ms
cutoff: 910000      10 times Time :2805ms
cutoff: 920000      10 times Time :2608ms
cutoff: 930000      10 times Time :2533ms
cutoff: 940000      10 times Time :2373ms
cutoff: 950000      10 times Time :2512ms
cutoff: 960000      10 times Time :2398ms
cutoff: 970000      10 times Time :2538ms
cutoff: 980000      10 times Time :2524ms
cutoff: 990000      10 times Time :2549ms
cutoff: 1000000     10 times Time :2394ms
Degree of parallelism: 8
cutoff: 510000      10 times Time :2126ms
cutoff: 520000      10 times Time :2521ms
cutoff: 530000      10 times Time :2361ms
cutoff: 540000      10 times Time :1828ms
cutoff: 550000      10 times Time :2013ms
cutoff: 560000      10 times Time :1767ms
cutoff: 570000      10 times Time :1862ms
cutoff: 580000      10 times Time :1777ms
cutoff: 590000      10 times Time :1737ms
cutoff: 600000      10 times Time :1813ms
cutoff: 610000      10 times Time :1815ms
cutoff: 620000      10 times Time :2077ms
cutoff: 630000      10 times Time :2522ms
cutoff: 640000      10 times Time :2256ms
```


INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (43 minutes ago)
35:42 LF UTF-8 4 spaces assignment-4

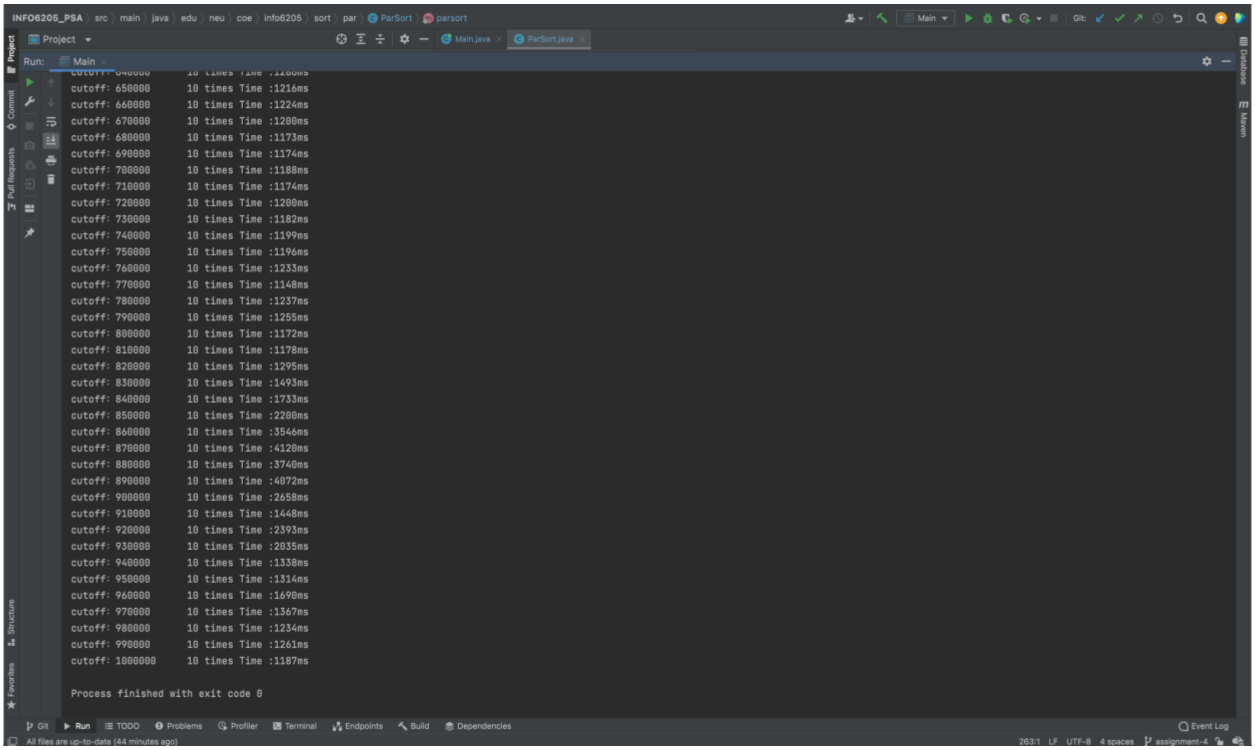
```
cutoff: 640000      10 times Time :2256ms
cutoff: 650000      10 times Time :2052ms
cutoff: 660000      10 times Time :1801ms
cutoff: 670000      10 times Time :1872ms
cutoff: 680000      10 times Time :2378ms
cutoff: 690000      10 times Time :1899ms
cutoff: 700000      10 times Time :1817ms
cutoff: 710000      10 times Time :1915ms
cutoff: 720000      10 times Time :1739ms
cutoff: 730000      10 times Time :1713ms
cutoff: 740000      10 times Time :1818ms
cutoff: 750000      10 times Time :1804ms
cutoff: 760000      10 times Time :1819ms
cutoff: 770000      10 times Time :1823ms
cutoff: 780000      10 times Time :1796ms
cutoff: 790000      10 times Time :1877ms
cutoff: 800000      10 times Time :1715ms
cutoff: 810000      10 times Time :1781ms
cutoff: 820000      10 times Time :1752ms
cutoff: 830000      10 times Time :1836ms
cutoff: 840000      10 times Time :1795ms
cutoff: 850000      10 times Time :1790ms
cutoff: 860000      10 times Time :1755ms
cutoff: 870000      10 times Time :1835ms
cutoff: 880000      10 times Time :1933ms
cutoff: 890000      10 times Time :1954ms
cutoff: 900000      10 times Time :1983ms
cutoff: 910000      10 times Time :2158ms
cutoff: 920000      10 times Time :1982ms
cutoff: 930000      10 times Time :1868ms
cutoff: 940000      10 times Time :1885ms
cutoff: 950000      10 times Time :1857ms
cutoff: 960000      10 times Time :1817ms
cutoff: 970000      10 times Time :1790ms
cutoff: 980000      10 times Time :1828ms
cutoff: 990000      10 times Time :1988ms
cutoff: 1000000     10 times Time :1810ms
Degree of parallelism: 16
cutoff: 510000      10 times Time :1624ms
```


INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (44 minutes ago)
35:42 LF UTF-8 4 spaces assignment-4

```
Degree of parallelism: 16
cutoff: 510000      10 times Time :1624ms
cutoff: 520000      10 times Time :1375ms
cutoff: 530000      10 times Time :1222ms
cutoff: 540000      10 times Time :1196ms
cutoff: 550000      10 times Time :1231ms
cutoff: 560000      10 times Time :1206ms
cutoff: 570000      10 times Time :1178ms
cutoff: 580000      10 times Time :1210ms
cutoff: 590000      10 times Time :1189ms
cutoff: 600000      10 times Time :1183ms
cutoff: 610000      10 times Time :1218ms
cutoff: 620000      10 times Time :1166ms
cutoff: 630000      10 times Time :1262ms
cutoff: 640000      10 times Time :1301ms
cutoff: 650000      10 times Time :1212ms
cutoff: 660000      10 times Time :1186ms
cutoff: 670000      10 times Time :1207ms
cutoff: 680000      10 times Time :1251ms
cutoff: 690000      10 times Time :1371ms
cutoff: 700000      10 times Time :1681ms
cutoff: 710000      10 times Time :1705ms
cutoff: 720000      10 times Time :1746ms
cutoff: 730000      10 times Time :1278ms
cutoff: 740000      10 times Time :1268ms
cutoff: 750000      10 times Time :1199ms
cutoff: 760000      10 times Time :1323ms
cutoff: 770000      10 times Time :1349ms
cutoff: 780000      10 times Time :1236ms
cutoff: 790000      10 times Time :1216ms
cutoff: 800000      10 times Time :1180ms
cutoff: 810000      10 times Time :1396ms
cutoff: 820000      10 times Time :1222ms
cutoff: 830000      10 times Time :1220ms
cutoff: 840000      10 times Time :1179ms
cutoff: 850000      10 times Time :1229ms
cutoff: 860000      10 times Time :1271ms
cutoff: 870000      10 times Time :1161ms
cutoff: 880000      10 times Time :1200ms
```


INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (44 minutes ago)

```
cutoff: 880000      10 times Time :1200ms
cutoff: 890000      10 times Time :1204ms
cutoff: 900000      10 times Time :1162ms
cutoff: 910000      10 times Time :1169ms
cutoff: 920000      10 times Time :1459ms
cutoff: 930000      10 times Time :1299ms
cutoff: 940000      10 times Time :1192ms
cutoff: 950000      10 times Time :1310ms
cutoff: 960000      10 times Time :1537ms
cutoff: 970000      10 times Time :1742ms
cutoff: 980000      10 times Time :1509ms
cutoff: 990000      10 times Time :1440ms
cutoff: 1000000     10 times Time :1592ms
Degree of parallelism: 32
cutoff: 510000      10 times Time :1307ms
cutoff: 520000      10 times Time :1206ms
cutoff: 530000      10 times Time :1203ms
cutoff: 540000      10 times Time :1298ms
cutoff: 550000      10 times Time :1382ms
cutoff: 560000      10 times Time :1218ms
cutoff: 570000      10 times Time :1174ms
cutoff: 580000      10 times Time :1183ms
cutoff: 590000      10 times Time :1240ms
cutoff: 600000      10 times Time :1238ms
cutoff: 610000      10 times Time :1293ms
cutoff: 620000      10 times Time :1347ms
cutoff: 630000      10 times Time :1325ms
cutoff: 640000      10 times Time :1286ms
cutoff: 650000      10 times Time :1216ms
cutoff: 660000      10 times Time :1224ms
cutoff: 670000      10 times Time :1200ms
cutoff: 680000      10 times Time :1173ms
cutoff: 690000      10 times Time :1174ms
cutoff: 700000      10 times Time :1188ms
cutoff: 710000      10 times Time :1174ms
cutoff: 720000      10 times Time :1200ms
cutoff: 730000      10 times Time :1182ms
cutoff: 740000      10 times Time :1199ms
cutoff: 750000      10 times Time :1196ms
cutoff: 760000      10 times Time :1233ms
```


INFO6205_PSA 〉 src 〉 main 〉 java 〉 edu 〉 neu 〉 coe 〉 info6205 〉 sort 〉 par 〉 ParSort 〉 parsort
Project
Main.java
ParSort.java
Run: Main
Git
Run
TODO
Problems
Profiler
Terminal
Endpoints
Build
Dependencies
Event Log
All files are up-to-date (44 minutes ago)

```
cutoff: 640000      10 times Time :1286ms
cutoff: 650000      10 times Time :1216ms
cutoff: 660000      10 times Time :1224ms
cutoff: 670000      10 times Time :1200ms
cutoff: 680000      10 times Time :1173ms
cutoff: 690000      10 times Time :1174ms
cutoff: 700000      10 times Time :1188ms
cutoff: 710000      10 times Time :1174ms
cutoff: 720000      10 times Time :1200ms
cutoff: 730000      10 times Time :1182ms
cutoff: 740000      10 times Time :1199ms
cutoff: 750000      10 times Time :1196ms
cutoff: 760000      10 times Time :1233ms
cutoff: 770000      10 times Time :1148ms
cutoff: 780000      10 times Time :1237ms
cutoff: 790000      10 times Time :1255ms
cutoff: 800000      10 times Time :1172ms
cutoff: 810000      10 times Time :1178ms
cutoff: 820000      10 times Time :1295ms
cutoff: 830000      10 times Time :1493ms
cutoff: 840000      10 times Time :1733ms
cutoff: 850000      10 times Time :2200ms
cutoff: 860000      10 times Time :3546ms
cutoff: 870000      10 times Time :4120ms
cutoff: 880000      10 times Time :3740ms
cutoff: 890000      10 times Time :4072ms
cutoff: 900000      10 times Time :2658ms
cutoff: 910000      10 times Time :1448ms
cutoff: 920000      10 times Time :2393ms
cutoff: 930000      10 times Time :2035ms
cutoff: 940000      10 times Time :1338ms
cutoff: 950000      10 times Time :1314ms
cutoff: 960000      10 times Time :1690ms
cutoff: 970000      10 times Time :1367ms
cutoff: 980000      10 times Time :1234ms
cutoff: 990000      10 times Time :1261ms
cutoff: 1000000     10 times Time :1187ms

Process finished with exit code 0
```

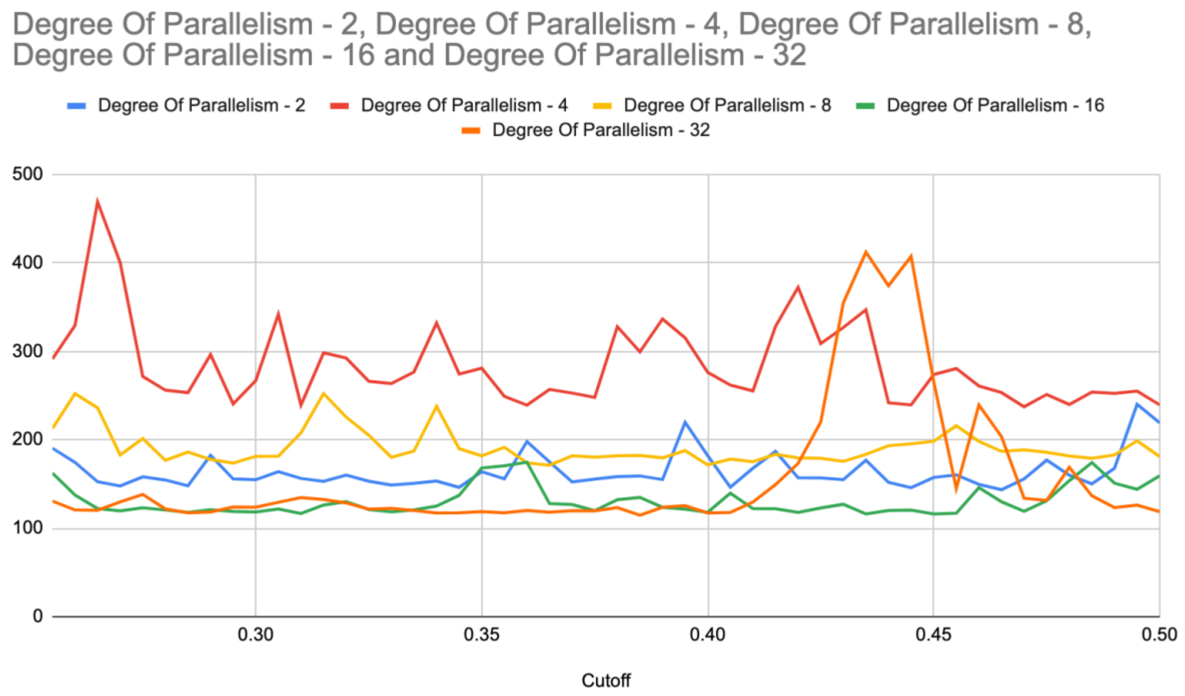
CSV DATA

Generated CSVs are pushed on github in src/parSortResults folder.

Cutoff	Degree Of Parallelism - 2	Degree Of Parallelism - 4	Degree Of Parallelism - 8	Degree Of Parallelism - 16	Degree Of Parallelism - 32
0.255	**190.5**	**291.2**	**212.6**	**162.4**	**130.7**
0.26	174.4	329	252.1	137.5	120.6
0.265	152.5	468.8	236.1	122.2	120.3
0.27	147.8	400.1	182.8	119.6	129.8
0.275	158.1	271.5	201.3	123.1	138.2
0.28	154.5	256	176.7	120.6	121.8
0.285	147.9	253.2	186.2	117.8	117.4
0.29	182.5	296.2	177.7	121	118.3
0.295	155.7	240.5	173.7	118.9	124
0.3	154.8	266.8	181.3	118.3	123.8
0.305	163.9	341.7	181.5	121.8	129.3
0.31	156.3	239.1	207.7	116.6	134.7
0.315	153	298.2	252.2	126.2	132.5
0.32	160	292.4	225.6	130.1	128.6
0.325	153.1	266.1	205.2	121.2	121.6
0.33	148.8	263.5	180.1	118.6	122.4
0.335	150.7	276.6	187.2	120.7	120
0.34	153.3	332	237.8	125.1	117.3
0.345	146.2	274.3	189.9	137.1	117.4
0.35	163.9	280.9	181.7	168.1	118.8
0.355	155.8	249.1	191.5	170.5	117.4
0.36	197.9	239.1	173.9	174.6	120
0.365	174.9	256.9	171.3	127.8	118.2
0.37	152.3	252.7	181.8	126.8	119.9
0.375	155.5	247.9	180.4	119.9	119.6
0.38	158.3	327.8	181.9	132.3	123.3
0.385	159.1	299.3	182.3	134.9	114.8
0.39	155	336.4	179.6	123.6	123.7
0.395	219.7	315.2	187.7	121.6	125.5

0.4	181.9	276.1	171.5	118	117.2
0.405	146.4	261.8	178.1	139.6	117.8
0.41	167.8	255.2	175.2	122.2	129.5
0.415	186.8	328.1	183.6	122	149.3
0.42	157	372.3	179.5	117.9	173.3
0.425	156.8	308.7	179	122.9	220
0.43	154.9	326.8	175.5	127.1	354.6
0.435	176.8	346.9	183.5	116.1	412
0.44	151.9	241.8	193.3	120	374
0.445	145.8	239.3	195.4	120.4	407.2
0.45	157.4	273.7	198.3	116.2	265.8
0.455	160.3	280.5	215.8	116.9	144.8
0.46	149.5	260.8	198.2	145.9	239.3
0.465	143.5	253.3	186.8	129.9	203.5
0.47	155.8	237.3	188.5	119.2	133.8
0.475	177	251.2	185.7	131	131.4
0.48	160	239.8	181.7	153.7	169
0.485	150.1	253.8	179	174.2	136.7
0.49	167.8	252.4	182.8	150.9	123.4
0.495	240	254.9	198.8	144	126.1
0.5	219.1	239.4	181	159.2	118.7

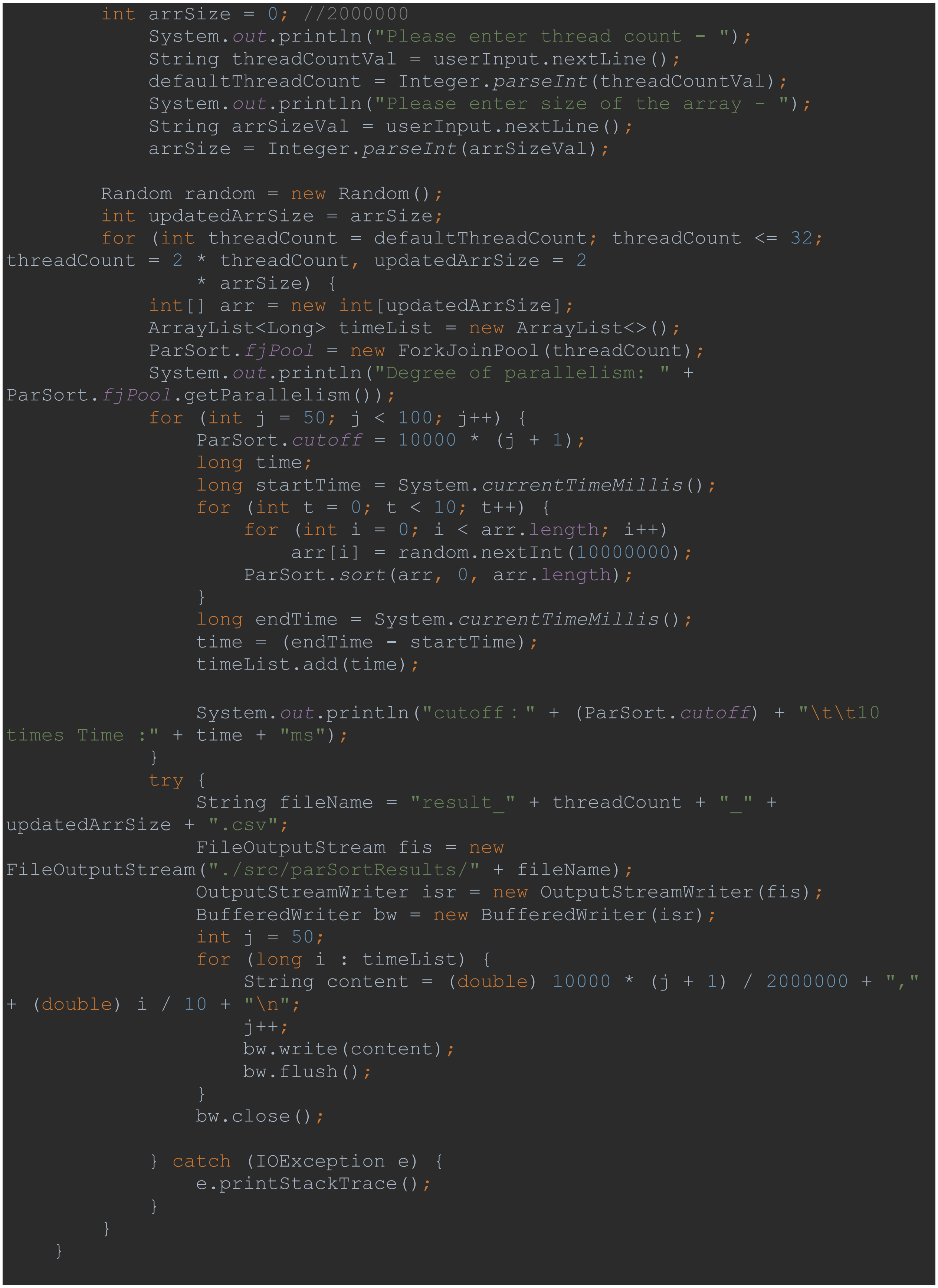
- **Evidence / Graph**



- **Code**

Main.java

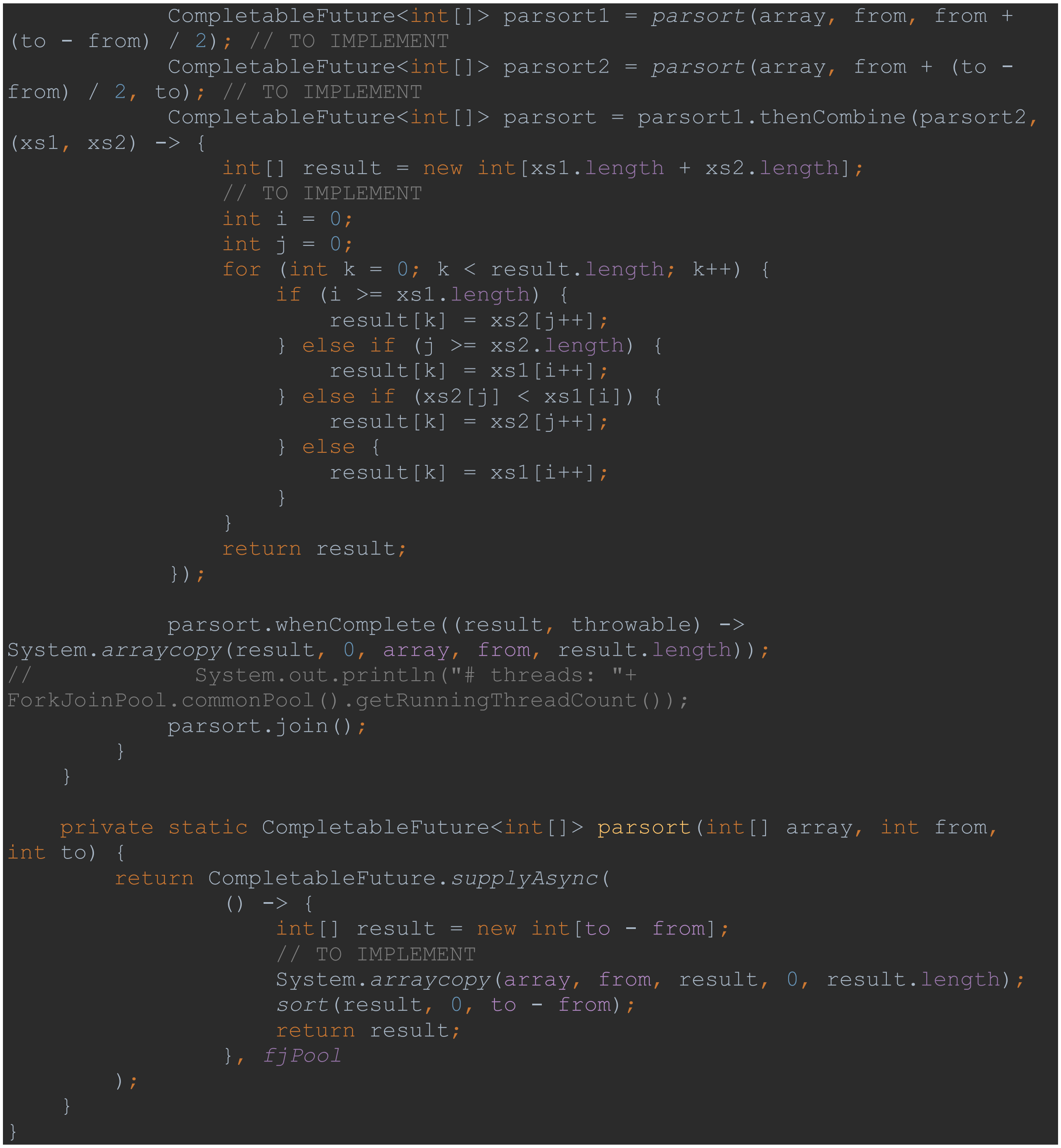
```
package edu.neu.coe.info6205.sort.par;

import java.io.BufferedWriter;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.util.*;
import java.util.concurrent.ForkJoinPool;

/**
 * This code has been fleshed out by Ziyao Qiao. Thanks very much. TODO tidy
it
 * up a bit.
 */
public class Main {

    public static void main(String[] args) {
        processArgs(args);
        Scanner userInput = new Scanner(System.in);
        int defaultThreadCount = 0;
```



```
        int arrSize = 0; //2000000
            System.out.println("Please enter thread count - ");
            String threadCountVal = userInput.nextLine();
            defaultThreadCount = Integer.parseInt(threadCountVal);
            System.out.println("Please enter size of the array - ");
            String arrSizeVal = userInput.nextLine();
            arrSize = Integer.parseInt(arrSizeVal);

        Random random = new Random();
        int updatedArrSize = arrSize;
        for (int threadCount = defaultThreadCount; threadCount <= 32;
threadCount = 2 * threadCount, updatedArrSize = 2
                * arrSize) {
            int[] arr = new int[updatedArrSize];
            ArrayList<Long> timeList = new ArrayList<>();
            ParSort.fjPool = new ForkJoinPool(threadCount);
            System.out.println("Degree of parallelism: " +
ParSort.fjPool.getParallelism());
            for (int j = 50; j < 100; j++) {
                ParSort.cutoff = 10000 * (j + 1);
                long time;
                long startTime = System.currentTimeMillis();
                for (int t = 0; t < 10; t++) {
                    for (int i = 0; i < arr.length; i++)
                        arr[i] = random.nextInt(10000000);
                    ParSort.sort(arr, 0, arr.length);
                }
                long endTime = System.currentTimeMillis();
                time = (endTime - startTime);
                timeList.add(time);

                System.out.println("cutoff：" + (ParSort.cutoff) + "\t\t10
times Time :" + time + "ms");
            }
            try {
                String fileName = "result_" + threadCount + "_" +
updatedArrSize + ".csv";
                FileOutputStream fis = new
FileOutputStream("./src/parSortResults/" + fileName);
                OutputStreamWriter isr = new OutputStreamWriter(fis);
                BufferedWriter bw = new BufferedWriter(isr);
                int j = 50;
                for (long i : timeList) {
                    String content = (double) 10000 * (j + 1) / 2000000 + ","
+ (double) i / 10 + "\n";
                    j++;
                    bw.write(content);
                    bw.flush();
                }
                bw.close();

            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
    private static void processArgs(String[] args) {
        String[] xs = args;
        while (xs.length > 0)
            if (xs[0].startsWith("-"))
                xs = processArg(xs);
    }

    private static String[] processArg(String[] xs) {
        String[] result = new String[0];
        System.arraycopy(xs, 2, result, 0, xs.length - 2);
        processCommand(xs[0], xs[1]);
        return result;
    }

    private static void processCommand(String x, String y) {
        if (x.equalsIgnoreCase("N"))
            setConfig(x, Integer.parseInt(y));
        else
            // TODO sort this out
            if (x.equalsIgnoreCase("P")) // noinspection
ResultOfMethodCallIgnored
                ForkJoinPool.getCommonPoolParallelism();
    }

    private static void setConfig(String x, int i) {
        configuration.put(x, i);
    }

    @SuppressWarnings("MismatchedQueryAndUpdateOfCollection")
    private static final Map<String, Integer> configuration = new
HashMap<>();

}
```

ParSort.java

```
package edu.neu.coe.info6205.sort.par;

import java.util.Arrays;
import java.util.concurrent.*;

/**
 * This code has been fleshed out by Ziyao Qiao. Thanks very much.
 * TODO tidy it up a bit.
 */
class ParSort {

    public static int cutoff = 1000;

    public static ForkJoinPool fjPool;


    public static void sort(int[] array, int from, int to) {
        if (to - from < cutoff) Arrays.sort(array, from, to);
        else {
            // FIXME next few lines should be removed from public repo.
```



```
            CompletableFuture<int[]> parsort1 = parsort(array, from, from +
(to - from) / 2); // TO IMPLEMENT
            CompletableFuture<int[]> parsort2 = parsort(array, from + (to -
from) / 2, to); // TO IMPLEMENT
            CompletableFuture<int[]> parsort = parsort1.thenCombine(parsort2,
(xs1, xs2) -> {
                int[] result = new int[xs1.length + xs2.length];
                // TO IMPLEMENT
                int i = 0;
                int j = 0;
                for (int k = 0; k < result.length; k++) {
                    if (i >= xs1.length) {
                        result[k] = xs2[j++];
                    } else if (j >= xs2.length) {
                        result[k] = xs1[i++];
                    } else if (xs2[j] < xs1[i]) {
                        result[k] = xs2[j++];
                    } else {
                        result[k] = xs1[i++];
                    }
                }
                return result;
            });

            parsort.whenComplete((result, throwable) ->
System.arraycopy(result, 0, array, from, result.length));
//              System.out.println("# threads: "+
ForkJoinPool.commonPool().getRunningThreadCount());
            parsort.join();
        }
    }

    private static CompletableFuture<int[]> parsort(int[] array, int from,
int to) {
        return CompletableFuture.supplyAsync(
                () -> {
                    int[] result = new int[to - from];
                    // TO IMPLEMENT
                    System.arraycopy(array, from, result, 0, result.length);
                    sort(result, 0, to - from);
                    return result;
                }, fjPool
        );
    }
}
```