## **PSA ASSIGNMENT 5**

#### YASH KHOPKAR

#### **CONCLUSION:**

We can conclude that as with the increase in the array size performance is increases. Initially when the array size is small system took more time to sort but if be continuously increase the size of the array after certain size we can find that eventually, it will take less time to sort the array and on further increase in the size of the array the time taken to sort the array increases by very small amount.

ARRAY SIZE	CUTOFF	SORTING TIME (MIL
		SEC)
2000	1000	68.0
4000	500	55.0
4000	1000	63.0
8000	500	68.0
8000	1000	63.0

#### **SCREENSHOTS:**

```
🛭 Main.java 🖾 🚨 ParSort.java

    Problems @ Javadoc    Declaration    □ Console    □

                                                                                                      7 public class Main {
                                                                            <terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_17
                                                                            cut-off: 500
       public static void main(String[] args) {
 9⊜
                                                                            cut-off: 500
10
                                                                            cut-off: 500
11 //
            int avgTime=0, avgThreadCount=0;
                                                                            cut-off: 500
12
                                                                            sorting time: 68.0 cut-off: 1000 Thread count: 6
13
14 //
             if (args.length>0) ParSort.cutoff = Integer.parseInt(args[0])
15
16 //
             for(ParSort.cutoff=500000; ParSort.cutoff<5000000; ParSort.cu
17
18 //
19 //
              for(int k=0:k<10:k++) {
                      ParSort.cutoff=1000;
20
21
22
23
            Random random = new Random(0L);
24
25
            int[] array = new int[2000];
26
            for (int i = 0; i < array.length; i++) array[i] = random.nextInt(20000000);</pre>
27
28 //
              ParSort.cutoff= array.length *30/100;
29
30 //
              ParSort.counThread = 1;
            double start = System.currentTimeMillis();
32
33
            ParSort.sort(array, 0, array.length-1);
34
35
            double end = System.currentTimeMillis();
37 //
             for (int i : array) System.out.println(i);
39
            System.out.println("sorting time: "+ (end-start)+ " cut-off: "+ ParSort.cutoff+ " Thread count: "+ ParSort.coun
```

```
    Main.java 
    □ ParSort.java

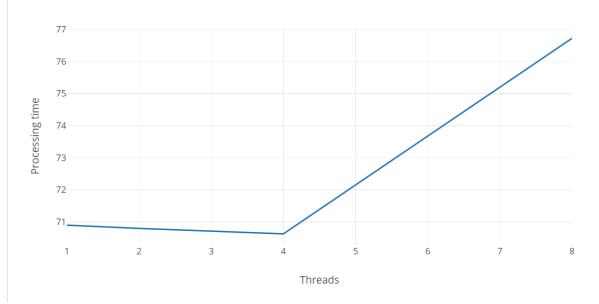
                                                                             Problems @ Javadoc  □ Declaration □ Console  □
                                                                                                       ■ X ¾ 🗎 🔐 🗗 🗗 🖶 🔻
   public class Main {
                                                                              <terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_17
                                                                             cut-off: 250
 9⊜
        public static void main(String[] args) {
                                                                             cut-off: 250
10
                                                                             cut-off: 250
11 //
             int avgTime=0, avgThreadCount=0;
                                                                             cut-off: 250
 12
                                                                              cut-off: 250
 13
                                                                             cut-off: 250
              if (args.length>0) ParSort.cutoff = Integer.parseInt(args[0] cut-off: 250
 14 //
 15
                                                                             cut-off: 250
              for(ParSort.cutoff=500000; ParSort.cutoff<5000000; ParSort.c cut-off: 250
 16 //
 17
                                                                             cut-off: 500
18 //
                                                                              sorting time: 55.0 cut-off: 500 Thread count: 28
 19 //
              for(int k=0;k<10;k++) {
20
21
                        ParSort.cutoff=500;
 22
 23
            Random random = new Random(0L):
 24
 25
            int[] array = new int[4000];
 26
            for (int i = 0; i < array.length; i++) array[i] = random.nextInt(20000000);</pre>
 27
 28 //
              ParSort.cutoff= array.length *30/100;
 29
 30 //
              ParSort.counThread = 1;
 31
            double start = System.currentTimeMillis();
 32
 33
            ParSort.sort(array, 0, array.length-1);
 35
            double end = System.currentTimeMillis();
 36
 37 //
              for (int i : array) System.out.println(i);
 38
 39
             System.out.println("sorting time: "+ (end-start)+ " cut-off: "+ ParSort.cutoff+ " Thread count: "+ ParSort.coun
 40 //
               if (array[0]==0) System.out.println("Success!");
              System.out.println("Thread count: "+ ParSort.counThread);
41 //
```

```
🔑 Main.java 🖾 🔑 ParSort.java
                                                                            Problems @ Javadoc  □ Declaration □ Console ⋈
                                                                                                     7 public class Main {
                                                                            <terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_
                                                                            cut-off: 500
        public static void main(String[] args) {
 9⊜
                                                                            cut-off: 500
10
                                                                            cut-off: 500
11 //
            int avgTime=0, avgThreadCount=0;
                                                                            cut-off: 500
12
                                                                            cut-off: 500
13
                                                                            cut-off: 500
             if (args.length>0) ParSort.cutoff = Integer.parseInt(args[0 cut-off: 500
14 //
15
                                                                            cut-off: 500
              for(ParSort.cutoff=500000); ParSort.cutoff<50000000; ParSort.cutoff sorting time: 63.0 cut-off: 1000 Thread count: 14
16 //
17
18 //
              for(int k=0; k<10; k++) {
19 //
20
                       ParSort.cutoff=1000:
21
22
23
            Random random = new Random(0L):
24
25
            int[] array = new int[4000];
26
            for (int i = 0; i < array.length; i++) array[i] = random.nextInt(20000000);</pre>
27
28 //
              ParSort.cutoff= arrav.length *30/100:
29
30 //
              ParSort.counThread = 1:
31
            double start = System.currentTimeMillis();
32
33
            ParSort.sort(array, 0, array.length-1);
34
35
            double end = System.currentTimeMillis();
36
37 //
             for (int i : array) System.out.println(i);
38
            System.out.println("sorting time: "+ (end-start)+ " cut-off: "+ ParSort.cutoff+ " Thread count: "+ ParSort.cour
39
40 //
              if (array[0]==0) System.out.println("Success!");
              System.out.println("Thread count: "+ ParSort.counThread);
41 //
```

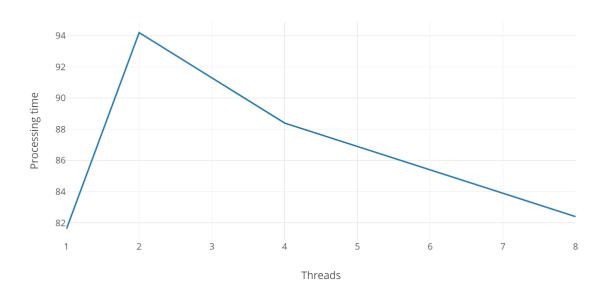
```
🚨 Main.java 🖾 🚨 ParSort.java
                                                                             🖫 Problems 🎯 Javadoc 🚇 Declaration 📮 Console 🛚
                                                                                                      ■ X ¾ 🕞 📑 🗗 🗗 🗗 🔻 📑
            int avgTime=0, avgThreadCount=0;
 11 //
                                                                             <terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_171
12
                                                                             cut-off: 250
13
14 //
                                                                             cut-off: 500
              if (args.length>0) ParSort.cutoff = Integer.parseInt(args[0])
                                                                             cut-off: 500
 15
                                                                             cut-off: 500
 16 //
             for(ParSort.cutoff=500000: ParSort.cutoff<5000000:ParSort.cu
                                                                             cut-off: 250
 17
                                                                             cut-off: 250
 18 //
                                                                             cut-off: 1000
 19 //
              for(int k=0;k<10;k++) {
                                                                             cut-off: 1000
 20
                      ParSort.cutoff=500;
                                                                             cut-off: 250
 21
                                                                             cut-off: 250
 22
                                                                             sorting time: 68.0 cut-off: 500 Thread count: 36
 23
            Random random = new Random(0L);
 24
 25
            int[] array = new int[8000];
 26
            for (int i = 0; i < array.length; i++) array[i] = random.nextInt(20000000);</pre>
 27
 28 //
              ParSort.cutoff= array.length *30/100;
 29
 30 //
             ParSort.counThread = 1;
 31
            double start = System.currentTimeMillis();
 32
 33
            ParSort.sort(array, 0, array.length-1);
 34
 35
            double end = System.currentTimeMillis();
 36
 37 //
             for (int i : array) System.out.println(i);
 38
            System.out.println("sorting time: "+ (end-start)+ " cut-off: "+ ParSort.cutoff+ " Thread count: "+ ParSort.coun
39
🚨 Main.java 🛭 🚨 ParSort.java
                                                                            Problems @ Javadoc 	☐ Declaration ☐ Console ※
10
                                                                                                      ■ X ¾ 🖟 🖟 🗗 🗗 🗗 🗗 🔻 📑 🔻
11 //
            int avgTime=0, avgThreadCount=0;
                                                                            <terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_171
12
                                                                            cut-off: 500
13
                                                                            cut-off: 500
14 //
             if (args.length>0) ParSort.cutoff = Integer.parseInt(args[0])
                                                                            cut-off: 500
15
                                                                            cut-off: 500
             for(ParSort.cutoff=500000; ParSort.cutoff<5000000; ParSort.cu
16 //
                                                                            cut-off: 500
 17
                                                                            cut-off: 500
18 //
                                                                            cut-off: 500
              for(int k=0; k<10; k++) {
19 //
                                                                            cut-off: 1000
                      ParSort.cutoff=1000;
 20
                                                                            cut-off: 500
21
                                                                            cut-off: 500
 22
                                                                            sorting time: 63.0 cut-off: 1000 Thread count: 28
 23
            Random random = new Random(0L);
 24
25
            int[] array = new int[8000];
 26
            for (int i = 0; i < array.length; i++) array[i] = random.nextInt(20000000);</pre>
27
28 //
              ParSort.cutoff= array.length *30/100;
29
30 //
             ParSort.counThread = 1:
            double start = System.currentTimeMillis();
 31
 33
            ParSort.sort(array, 0, array.length-1);
 34
 35
            double end = System.currentTimeMillis():
36
 37 //
             for (int i : array) System.out.println(i):
38
 39
            System.out.println("sorting time: "+ (end-start)+ " cut-off: "+ ParSort.cutoff+ " Thread count: "+ ParSort.coun
 40 //
              if (array[0]==0) System.out.println("Success!");
```



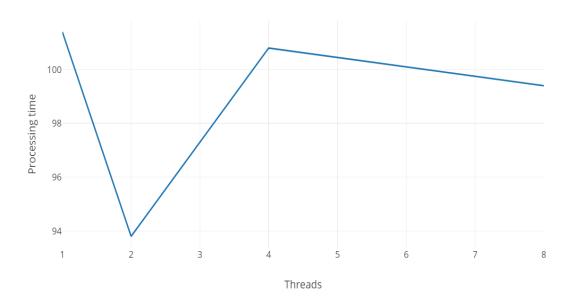
### Array[2000], Cutoff-1000



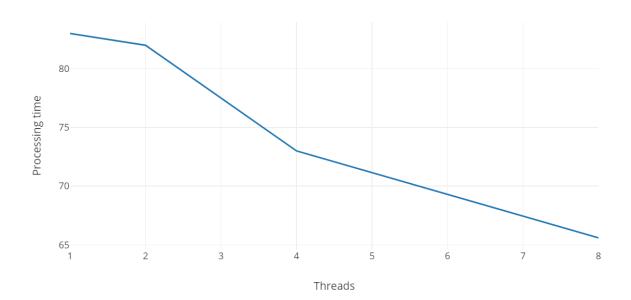
### Array[8000], Cutoff-1000



### Array[8000], Cutoff-500



# Array[4000], Cutoff-1000



#### Array[4000], Cutoff-500

