PSA Assignment 4 Parallel Sorting

Name Siddharth Kushal

NUID 002104288

Tasks:

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.

Below were the applicable code changes for this assignment:

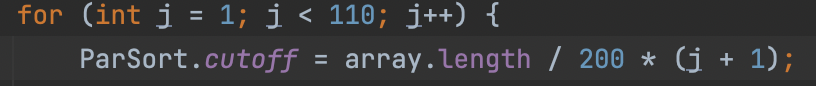
1. Array size of different size (more than the cutoff) – 1 mil, 2 mil and 3 mil

2. Threads increased from range 2 to 32 using Math.pow function

Text

Description automatically generated

Increase the cutoff



Degree of Parallelism: 7

I experimented with several cutoff values, numerous threads, and three distinct array sizes.

The output is summarized below.

Text, calendar

Description automatically generated with medium confidence

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

Array size – 1000000

Chart, line chart

Description automatically generated

Array Size: 2000000

Chart, line chart

Description automatically generated

Array size : 3000000

Chart, line chart

Description automatically generated

CONCLUSION:

Part 1:

We can infer from above graphs that:

The best is for the array size of 100000.

The cutoff is at 12000 and the time is 308ms.

Part 2:

It takes 12% of the array size for the thread 16.

Best cutoff = array size/ no. of threads.

Part 3:

From the graphs plotted the lowest is at thread 16

The graph shows that time consumed on sorting where array size is 100000 and cutoff is 12000

The lowest is at thread 16

Chart, line chart

Description automatically generated