**ASSIGNMENT 4 (Parallel Sorting)**

**Tejasvi Bhadrinath- 2952291**

Output Screen samples:

For 200000 elements:

A screenshot of a computer

Description automatically generated

Text

Description automatically generated

For 500000 elements:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

For 1000000:

Text

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Observations:



Conclusion:

* For a fixed-array size, as the cut-off increases the efficiency of the sorting algorithm drops. With the least performance seen where there is no parallelism, i.e when the cut-off is the same as the array size.
* As the depth is increased, i.e more recursive sorting is done – the sorting algorithm performs better. The most effective depth seems to be 3 where there a total of 8 parallel threads sorting parts of the unsorted array. Increasing or decreasing the depth seems to result in a less efficient sorting algorithm. This is expected as the experiment was run on an eight-thread machine.
* Across all observations the most efficient combination for cut-off and threads is when the number of threads was 8 and the cut-off was a small percentage of the array size (~10%)