**Program Structures & Algorithms**

Assignment No. 4 - Parallel Sorting

Mayannk Kumaar - 001537115

**Task:**

* Step 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.
* Step 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).

**Output Values:Graphical user interface, text

Description automatically generated**

**Graphical user interface, text

Description automatically generated**

**Graphical user interface, text

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Code:**

Main.java, ParSort.java files attached.

**Observations:**

As per observation, **32** threads are performing sorting better as the array size increases. So, performing sorting on various high array size by using only 32 threads we found a cutoff value to be approximately **300000**