Exercise 3: Task Parallelism

test environment:
Used PC had 12 Cores

SpeedUP:

Test Data: We used the same array in every test. The array has a size of 10 million integers to ensure a meaningful result, and the integers are evenly (randomly) distributed to create a fair sample. We used for the **Threshold the length of the current array**.

The runtime of both the Quicksort and the Mergesort are drawn in the diagram below

Tested Algorithms:
1. Quicksort Single threaded
2. Quicksort Parallel with Thresholds between 0 and 5.000.000

The **best speedup was 4,66** with a threshold of 540.000.
Like we can see the **optimised number of threshold** is **between: 360.000 and 610.000**
When the array size was reduced to **5%** of the original array, sequential Sorting seems to outperform further parallel subdividing.
The naive algorithm is much slower than the single threaded approach, because of the huge overhead, which is why it makes sense to resort back to the single threaded algorithm for smaller array sizes. The exact data will be appended to the submission as an excel sheet.

3. Mergesort Single threaded
4. Mergesort Parallel with Threshold between 0 and 5.000.000

The **best speedup was 3,26** with a threshold of 1.57 million.
The **optimised number of threshold** lies between: **1.000.000 and 2.000.000**
For the mergesort, the best threshold for us is between 10% and 20% of the original array size.

As a side note: with the naive parallel approach we had a speedup of 0.004 (Quicksort) and 0.002 (Mergesort), which means that the naive approach takes 200 - 400 times as long as the sequential algorithm.

Threshold	Quicksort	Mergesort
0	374912	1002245
10000	1423	3068
20000	899	2426
30000	692	2758
40000	592	2360
50000	559	2268
60000	544	2092
70000	516	2037
80000	452	1920
90000	475	1893
100000	449	1676
110000	458	2217
120000	411	1922
130000	398	2743
140000	437	1783
150000	408	2310
160000	408	2452
170000	379	2729
180000	401	1515
190000	405	1851
200000	383	1723
210000	380	2020
220000	366	1497
230000	376	1524
240000	368	2142
250000	355	1493
260000	371	1596
270000	375	1360
280000	364	1299
290000	367	1239
300000	362	1323
310000	377	1369
320000	372	1233
330000	367	1179
340000	358	1112
350000	372	1134
360000	361	1240
370000	392	1235