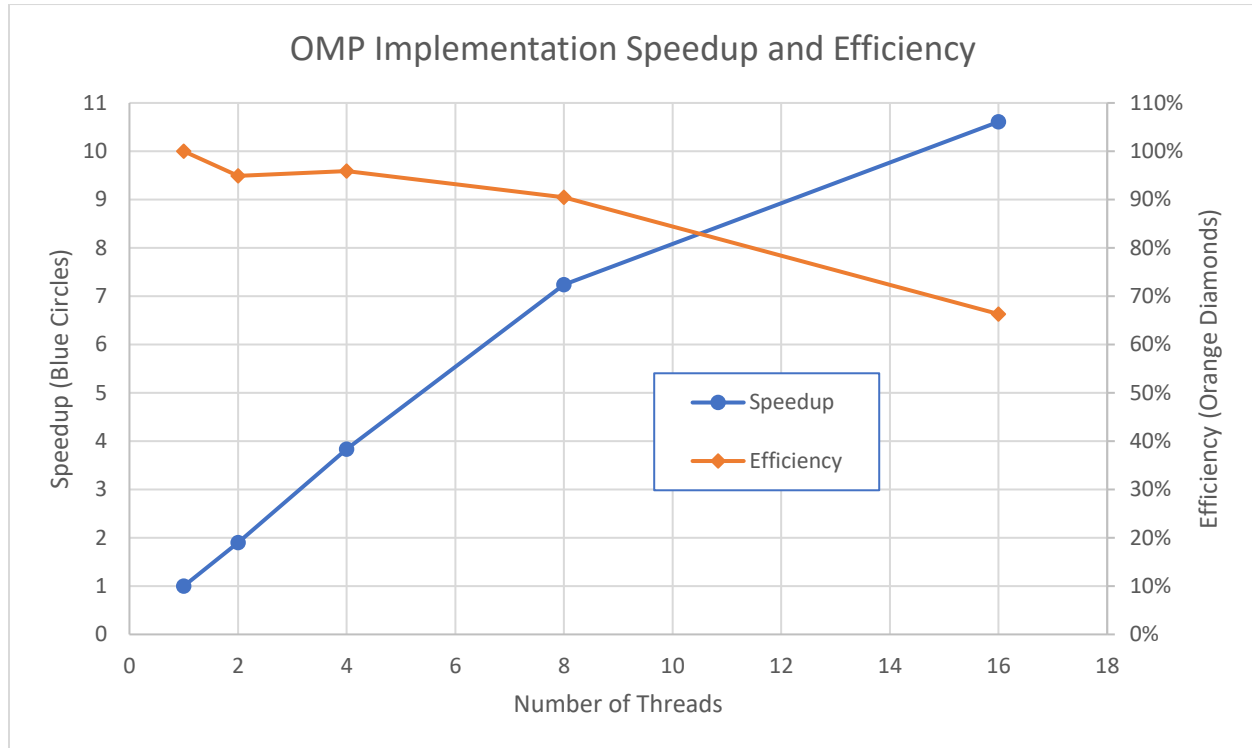Group 1: William Beck, and Robert Lucio

Homework 10 Results Summary

A plot of the speedup and efficiency of the Open MP parallelization of the linear sort algorithm is shown below.

The speedup was surprisingly linear for the first 8 threads. This may be because of the shared memory parallelization, as opposed to the distributed memory parallelization seen previously in the course. Additionally, the efficiency stays above or near 90% for up to eight cores. However, afterward, the performance begins to diminish appreciably. This is likely due to the increasing overhead of the parallelization. Additionally, the efficiency Even with the parallelization, the highest speed was seen with the binary sort. As stated in the Lab 9 report, the binary search takes a max number of $N = \text{Floor}(\log_2(array\ size)) + 1$ iterations with some overhead. In contrast, increasing the number of parallel workers can only at best decrease the search time by the number of active workers, assuming perfect efficiency.