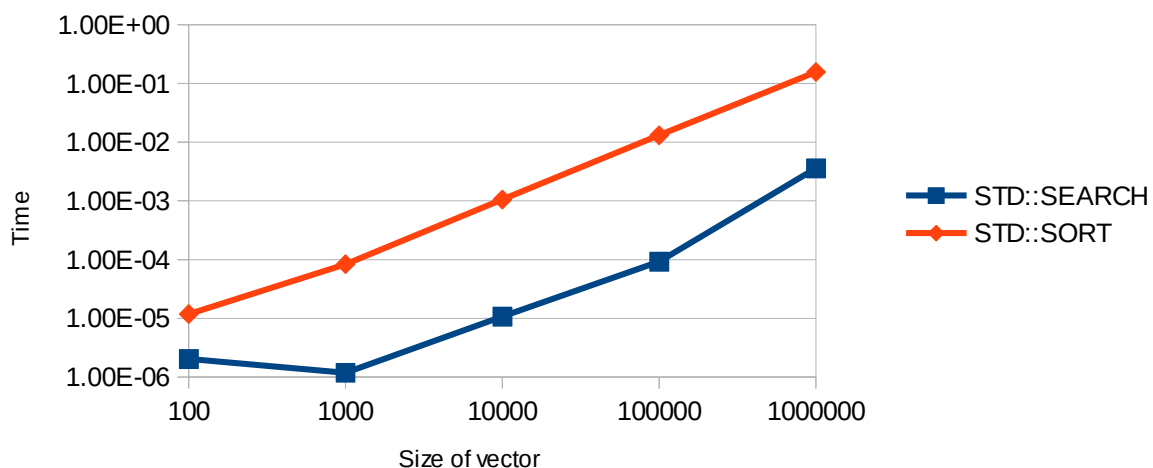


Parallel Search and Sort

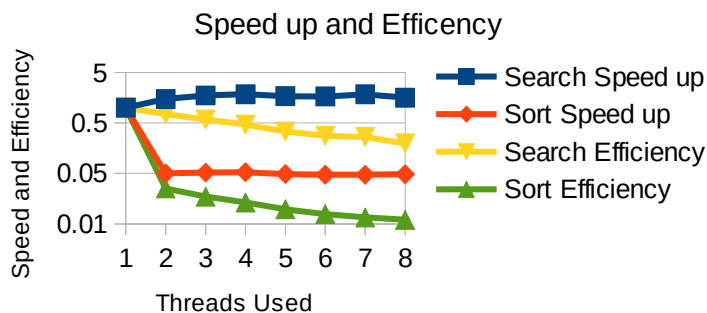
This project was very informative. It showed that while you can search and sort in parallel, the gains are minimal if anything. Most of the time my parallel search algorithm was slower than `std::search`, there were a few times I could beat the time, but it was so close that it really didn't matter. For the `std::sort` however that was a different story. There was no way I could beat it. It is just very fast and efficient. The main take away lesson is to use `std::search` and `std::sort`, as they are both very fast, and consume minimal resources.

Enough of that, now its time for pretty graphs. When applicable the Graphs show data for the 1,000,000 size. Just so that I could keep this report to one page. Also all the vectors used were seeded with random values, using `rand()`;

STD::SEARCH and STD::SORT



Size = 1000000



Size = 1000000

