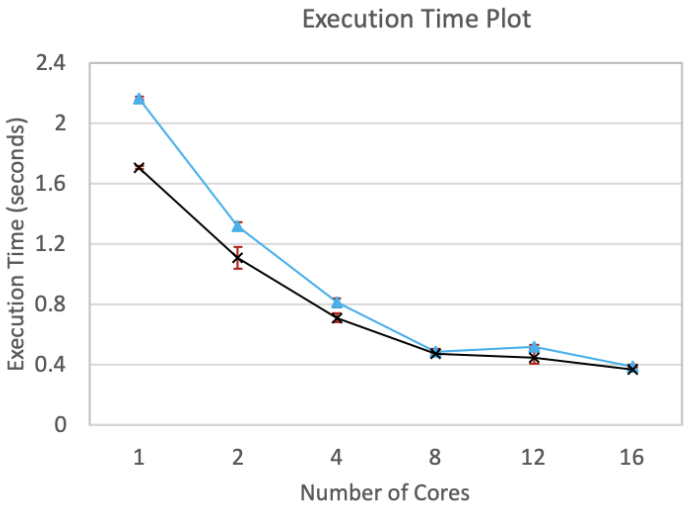
CSE 539S Project 1 Report

Amanda Hua, Funda Atik

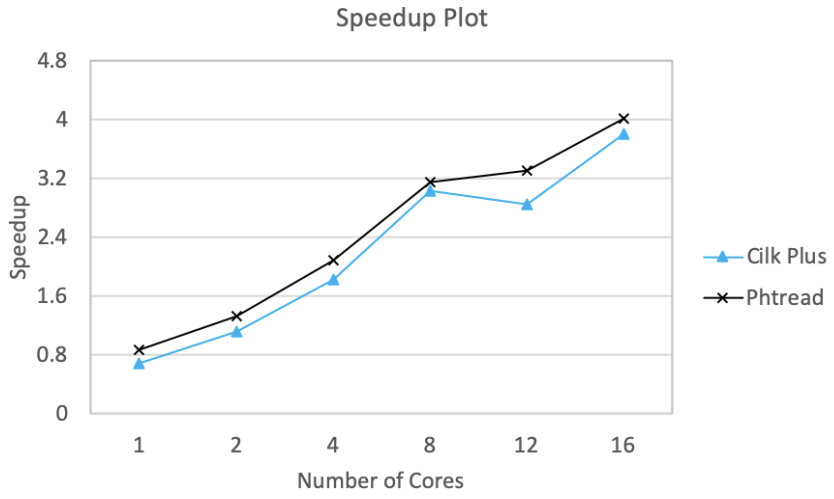
1. We didn't have any changes in our algorithm from the one presented in class. The essence of the parallelization came from providing a parallel merge in cilk_sort - a recursive algorithm that used binary search for optimized replacement of items in a sorted fashion.
2. The parallelization strategy for pthread was different in that we had to take into account the method of spawning a pthread. It was necessary to create separate structures to pass parameters as a single parameter to a method every time to ensure proper joins and track the number of threads created.
3. Execution Time Plot

   Each test is run with an array of size 10 million.

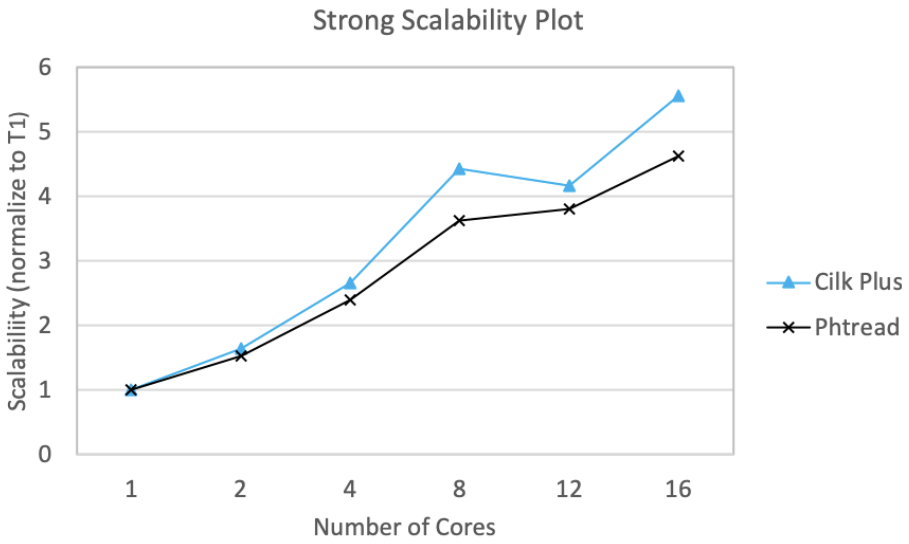


# Cores	cilk Std. dev (%)	pthread Std. dev (%)
1	0.52%	0.58%
2	2.06%	6.41%
4	3.58%	4.41%
8	3.28%	3.08%
12	2.14%	9.69%
16	2.92%	3.36%

4. Speedup Plot



5. Scalability Plot



6. We first came together to discuss how to implement a parallel merge sort algorithm by using Cilk Plus and pthreads, how they are different from each other, what helper functions we need. After we decided on the algorithm for both versions, we wrote the serial version of the merge sort together. Amanda wrote the base of cilk implementation, and Funda wrote the base of pthreads implementation. Then, we met again to revise these two versions and fix any bugs that we have.