# Parallel Computing Homework 2

Tyler Scotti

Feburary 9th, 2026

## Question 1: Revising the code to implement a thread-based parallel merge sort.

```
[tscotti24@grace3 HW2-735]$ ./sort_list.exe 4 1
List Size = 16, Threads = 2, error = 0, time (sec) =   0.0001, qsort_time =   0.0000
[tscotti24@grace3 HW2-735]$ ./sort_list.exe 4 2
List Size = 16, Threads = 4, error = 0, time (sec) =   0.0002, qsort_time =   0.0000
[tscotti24@grace3 HW2-735]$ ./sort_list.exe 4 3
List Size = 16, Threads = 8, error = 0, time (sec) =   0.0004, qsort_time =   0.0000
[tscotti24@grace3 HW2-735]$ ./sort_list.exe 20 4
List Size = 1048576, Threads = 16, error = 0, time (sec) =   0.0116, qsort_time =   0.1115
[tscotti24@grace3 HW2-735]$ ./sort_list.exe 24 8
List Size = 16777216, Threads = 256, error = 0, time (sec) =   0.1245, qsort_time =   2.1138
[tscotti24@grace3 HW2-735]$
```

Figure 1: Execution Time and Error Proof

## Question 2: Plot Speed up and Efficency

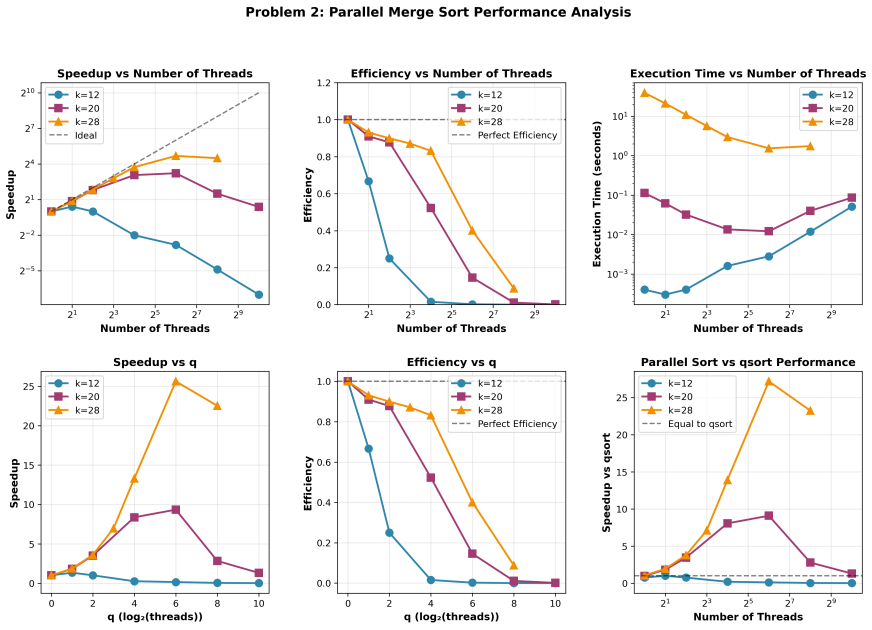


Figure 2: Speed Up and Efficency

# Question 3: Speed up

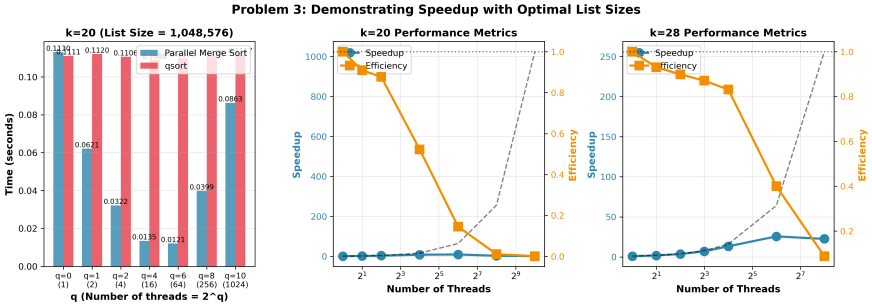


Figure 3: Speed up