Computer Science 385
Design and Analysis of Algorithms
Siena University
Spring 2026

Emprical Study 2

Group Formation: previously formed
Due: 4:00 PM, Friday, April 10, 2026 (code)
and 4:00 PM, Friday, April 17, 2026 (writeup)

This is an expansion on the first empirical study you completed earlier to include more sorting algorithms. The expectation is that you will work in the same groups and use the same repository, but changes in group membership may be requested.

Like with the earlier study, you are permitted to use AI assistance, with appropriate documentation, to implement the sorting algorithms.

Extended Empirical Analysis of Sorting Algorithms

Extend your code and expand your analysis to include the following sorting algorithms:

- mergesort
- quicksort with the pivot chosen as the first element in the (sub)array
- quicksort with the pivot chosen as a random element in the (sub)array
- quicksort with the pivot chosen by the "median of three" method
- heapsort

Notes on recursive implementations:

- You can probably get away with a recursive implementation for mergesort and not run into trouble, but an iterative implementation is likely to be a bit more efficient.
- Heapsort is just as easily (possibly more easily) implemented iteratively, but a recursive version should work if you prefer.
- A recursive quicksort is likely to run into call stack limitations when the worse-case behavior results in unbalanced splits.

Grading

This assignment will be graded out of 70 points.

Feature	Value	Score
Code	40	
Writeup	30	
Total	70	