**Cross-Review Summary:**

This summary compares **Heap Sort** and **Shell Sort** based on their **theoretical complexity, empirical performance, code quality, and optimizations.** The aim is to identify strengths, weaknesses, and suitable scenarios for each algorithm.

Comparison:

- Heap Sort guarantees O (n log n) performance across all cases.

- Shell Sort can be faster on medium arrays, particularly with Sedgewick or Knuth sequences.

**-**Both are in-place, space efficient, O(1).

-Heap Sort scales predictably as O(n log n) with consistent comparisons and swaps.

-Shell Sort empirical runtime depends on gap sequence: Sedgewick < Knuth < Shell.

Code quality:

Both are well-structured and readable, allow detailed benchmarking.

Optimization: the analysis proposed csv adding, added. Improved readability.

Overall:

Heap sort provides deterministic, reliable performance.

Shell sort provides faster runtime for practical input sizes.