

Project 1

<p>30% Project 1 (CLO/Synth) Develop practical skills with standard methods and tools used to analyze algorithms</p>
--

Tasks

1. Code and generate run logs.
 - Natural recursive mergesort
 - Non-recursive version (give a pseudocode)
2. Study sequence of instances.
3. Study efficiency sequence (generate the sequence explicitly).
4. Investigate efficiency variability; describe typical instances in each case, if any.
5. Determine if the sorting is stable.
6. Collect run-time data.
7. Plot both the efficiency sequence and the runtime and compare.
8. Use Excel to investigate the order of growth for both and compare; determine the efficiency class.

Report

1. Code and pseudocode (pick a professional tool and a style).
2. Run logs (show instances, counts, sorting stability, test cases and typical instances).
3. Plots from Task 7.
4. Excel-based order of growth investigation.
5. Write up (1–2 pages max): commentary on basic operation choices, discovered behaviors, plots, and efficiency.
6. References, if any (for course slides, use the format: Slide 5-1 for slide 1 in slide set 5).

Submission – Due TBA

- Report items 1 – 2: 30-minute in-class presentation (including demo runs)
- Report items 3 – 6: upload PDF from ASSIGNMENT page in course website